

# «Единый клиент»

Руководство по эксплуатации

# СОДЕРЖАНИЕ

<b>РУКОВОДСТВО ПО ИНТЕГРАЦИИ</b> .....	<b>6</b>
Сценарии интеграции .....	7
Загрузить данные из исходных систем в ЕК .....	8
Выгрузить данные из ЕК в хранилище и BI-системы .....	9
Найти клиента в ЕК из CRM или колл-центра .....	10
Исправить клиентские данные в онлайн, при вводе в фронтенд-системах .....	11
Найти потенциальные дубли в онлайн, при создании клиента в фронтенд-системе .....	12
Передать эталонные карточки из ЕК обратно в исходные системы .....	13
Идентифицировать клиента .....	16
Загрузить стоп-листы и сверить с клиентской базой .....	22
Механизмы интеграции .....	25
Полный цикл обработки данных через JMS .....	27
<b>РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ</b> .....	<b>30</b>
Процесс работы дата-стюардов и типовые вопросы .....	34
Процесс работы дата-стюардов .....	48
Типовые вопросы при работе дата стюардов .....	55
Требования к менеджеру данных .....	62
Общие сценарии .....	63
Начало работы с системой .....	64
Работа со списком контрагентов .....	65
Просмотр карточки контрагента .....	84
Работа со списком групп дубликатов .....	138
Просмотр группы дубликатов .....	156
Работа с черными списками .....	158
Выход в АРМ Администратора .....	160
Подтверждение и отклонение связей .....	161
Сценарии работы менеджера данных .....	163
Проверка информации о контрагентах .....	164
Работа с группами дубликатов .....	174
Работа с конфликтующими изменениями .....	180
Разъединение дубликатов контрагентов .....	184
Сценарии работы оператора .....	187
Расширенный поиск .....	188
Поиск по реквизитам контрагента .....	193
Поиск по реквизитам юрлица .....	195
Поиск по адресам .....	196
Поиск по телефонам .....	198
Поиск по email .....	202
Поиск по документам .....	204
Поиск по дубликатам .....	210
Поиск по группам дубликатов .....	211
Поиск по аккаунтам .....	212
Поиск по реквизитам любого контрагента .....	213
Описание кодов качества .....	214
Коды качества полей документа ФЛ (ДУЛ) .....	216
Коды качества email .....	248
Коды качества адреса .....	258
Коды качества телефона .....	282
Коды качества ФИО и пола .....	290

Прочие коды качества .....	320
Горячие клавиши .....	345
<b>РУКОВОДСТВО ОФИЦЕРА БЕЗОПАСНОСТИ .....</b>	<b>347</b>
Защита системы «Единый клиент» от воздействия вредоносного кода .....	350
Настройка SSL (https) .....	351
Режимы функционирования .....	352
<b>РУКОВОДСТВО ПО СОПРОВОЖДЕНИЮ .....</b>	<b>353</b>
Релизы «Единого клиента» .....	354
Диагностика проблем .....	355
Утилита диагностики .....	356
Read-only пользователь СУБД для диагностики .....	365
Диагностические SQL-запросы .....	366
Трассировка на уровне СУБД .....	376
Lucene. Медленно открываются страницы в UI .....	378
Проблемы с жестким диском на Windows — сбор информации .....	379
Автоматический мониторинг логов .....	380
Диагностика «зависаний» JVM с помощью jHiccup .....	381
Как снять диагностику выполнения web-запросов (access.log) .....	382
Профилирование удаленного стенда .....	384
Мониторинг производительности систем на Linux .....	396
Снятие полной профилировки на стенде .....	401
Диагностика запросов из браузера .....	408
Типовые проблемы .....	409
АРМ Менеджера данных долго загружает список контрагентов .....	410
Недоступен CleanService при отсутствии доступа к сервисам ФМС .....	412
Ошибка автоматической миграции .....	413
Стандартные характеристики сервера приложения .....	414
Как создать запрос в поддержку и управлять им .....	417
Инструкция по работе с JIRA .....	418
Собрать диагностику .....	420
<b>МОНИТОРИНГ РАБОТЫ СИСТЕМЫ .....</b>	<b>422</b>
Сетевая доступность .....	423
Доступность БД .....	424
Доступность «Фактора» .....	429
Доступность «Единого клиента» .....	433
Работоспособность задач пакетной обработки данных .....	438
Работоспособность интерфейса пользователя .....	440
Мониторинг журналов .....	441
Метрики для мониторинга .....	443
<b>РУКОВОДСТВО ПО РЕЗЕРВНОМУ КОПИРОВАНИЮ .....</b>	<b>447</b>
<b>РУКОВОДСТВО ПО ВОССТАНОВЛЕНИЮ ПОСЛЕ СБОЯ .....</b>	<b>449</b>
Восстановление БД .....	450
Восстановление ПО на сервере приложений .....	451
Запуск системы Фактор .....	453

Запуск системы Единый клиент .....	454
Восстановление поискового индекса .....	455
<b>SOAP И REST (ОПИСАНИЕ ОНЛАЙН-ИНТЕРФЕЙСОВ) .....</b>	<b>457</b>
Объектная модель онлайн-интерфейсов .....	459
Примеры — объектная модель (онлайн) .....	461
Элементы — объектная модель (онлайн) .....	464
Общее описание объектной модели .....	506
PartyWS и PartyRA .....	507
Party.save .....	509
Party.saveAndMerge .....	536
Party.getByRawID .....	555
Party.getByHID .....	569
Party.search .....	583
Party.fuzzySearch .....	608
Party.findInFns .....	638
Party.searchRelatedParties .....	643
Party.checkBlacklist .....	654
Party.checkBlacklistByRawID .....	670
Party.checkBlackListByHid .....	676
Party.close .....	681
Party.closeAttribute .....	684
Party.closeAttributeByRawId .....	686
Party.getReference .....	689
Party.saveRelations .....	695
Party.getRelations .....	700
Party.getSourceAttributes .....	705
Party.mergeSource .....	709
PartyWS и PartyRA (WSDL, WADL и схемы) .....	712
TaskWS и TaskRA .....	713
Task.execute .....	714
Task.status .....	717
TaskWS и TaskRA (WSDL и схемы) .....	720
CleanService .....	721
CleanService.address .....	724
CleanService.email .....	739
CleanService.legal .....	742
CleanService.passport .....	747
CleanService.phone .....	755
CleanService.physical .....	760
CleanService.soap-name-declension .....	764
Безопасность .....	768
Особенности сервисов .....	769
<b>ОПИСАНИЕ JMS-ИНТЕРФЕЙСА .....</b>	<b>770</b>
Расширения JMS-сообщения .....	771
Ограничения сервиса .....	781
Примеры использования .....	782
JMS. Создание контрагента .....	783
JMS. Изменение контрагента .....	786
JMS. Слияние контрагентов .....	830
JMS. Закрытие контрагента .....	850
JMS. Изменение значения поля .....	856

JMS. Разлиание контрагентов .....	858
JMS. Создание связей .....	865
JMS. Закрытие связей .....	869
JMS. Восстановление на версию .....	870
JMS. Проверка по черным спискам .....	897
JMS. Настройка отправки прошлых значений ФИО и ДУЛ .....	903
Структура JMS сообщения .....	906
Заголовки выходных JMS-сообщений .....	907
Тело выходных JMS-сообщений .....	908
JMS.FAQ .....	913

# РУКОВОДСТВО ПО ИНТЕГРАЦИИ

Типичные задачи интеграции:

- Загрузить данные из исходных систем в ЕК
- Выгрузить данные из ЕК в хранилище и BI-системы
- Найти клиента в ЕК из CRM или колл-центра
- Исправить клиентские данные в онлайн, при вводе в фронтенд-системах
- Найти потенциальные дубли в онлайн, при создании клиента в фронтенд-системе
- Передать эталонные карточки из ЕК обратно в исходные системы

Механизмы интеграции:

- Пакетная загрузка (ETL, файлы)
- Пакетная выгрузка (представления СУБД)
- SOAP и REST-сервисы (онлайн)
- JMS-интерфейс (очереди сообщений)

## СЦЕНАРИИ ИНТЕГРАЦИИ

- Загрузить данные из исходных систем в ЕК
- Выгрузить данные из ЕК в хранилище и BI-системы
- Найти клиента в ЕК из CRM или колл-центра
- Исправить клиентские данные в онлайн, при вводе в фронтенд-системах
- Найти потенциальные дубли в онлайн, при создании клиента в фронтенд-системе
- Передать эталонные карточки из ЕК обратно в исходные системы
- Идентифицировать клиента
- Загрузить стоп-листы и сверить с клиентской базой

# ЗАГРУЗИТЬ ДАННЫЕ ИЗ ИСХОДНЫХ СИСТЕМ В ЕК

Пользователи создают и изменяют клиентские карточки в исходных системах. ETL/ESB-система забирает их и передает в ЕК:

- **пакетно** (инкрементами через ETL-процедуры, два-три раза в сутки),
- либо в онлайн-режиме (через SOAP/REST-сервисы или JMS-очереди).

Используйте пакетный режим, если:

- Данные в исходной системе изменяются редко.
- Данные в исходной системе изменяются часто, но для внешних систем становятся «видны» раз в сутки или реже.
- У исходной системы нет онлайн-интерфейса (есть доступ только на уровне БД).
- Вы предпочитаете ETL-инструментарий для интеграции систем (вместо ESB).
- Не обязательно передавать изменения в ЕК моментально, можно несколько раз в сутки.

Используйте онлайн-режим, если:

- Изменения клиентских данных нужно передавать в ЕК моментально.
- Вы предпочитаете ESB (интеграционную шину) с поддержкой SOAP или JMS (вместо ETL).



# ВЫГРУЗИТЬ ДАННЫЕ ИЗ ЕК В ХРАНИЛИЩЕ И BI-СИСТЕМЫ

ETL/ESB-система забирает эталонные клиентские карточки из ЕК и передает их в хранилище (Data Warehouse) или Business Intelligence систему.

Если предпочитаете ETL-инструментарий, используйте представления СУБД (db view).

Если предпочитаете ESB (интеграционную шину), забирайте данные через JMS-очередь сообщений.

## НАЙТИ КЛИЕНТА В ЕК ИЗ CRM ИЛИ КОЛЛ-ЦЕНТРА

1. Пользователь в CRM указывает любую известную информацию о клиенте (ФИО, дата рождения, ИНН, адрес, телефон, номер паспорта, номер договора и т.п.)
2. CRM вызывает сервис [полнотекстового поиска ЕК](#).
3. ЕК возвращает список клиентов, подходящих под указанные критерии.
4. CRM показывает список пользователю.
5. Пользователь выбирает клиента из списка.
6. CRM открывает карточку клиента по его идентификатору. Полную информацию по клиенту CRM получает из ЕК [отдельным запросом](#).

## ИСПРАВИТЬ КЛИЕНТСКИЕ ДАННЫЕ В ОНЛАЙН, ПРИ ВВОДЕ В ФРОНТЕНД- СИСТЕМАХ

1. Пользователь вводит контактные данные клиента в фронтенд-системе.
2. Она вызывает [онлайн-интерфейс стандартизации ЕК](#).
3. ЕК возвращает исправленные данные обратно в фронтенд-систему.
4. Система показывает исправленные данные и коды качества пользователю.
5. Пользователь проверяет и подтверждает клиентские данные.
6. Фронтенд-система сохраняет клиентскую карточку.

Через ЕК можно проверить клиентскую карточку целиком или отдельные ее компоненты (ФИО, ИНН, адрес, телефон, документ, email).

## НАЙТИ ПОТЕНЦИАЛЬНЫЕ ДУБЛИ В ОНЛАЙН, ПРИ СОЗДАНИИ КЛИЕНТА В ФРОНТЕНД- СИСТЕМЕ

1. Пользователь в фронтенд-системе указывает контактные данные клиента.
2. Фронтенд-система вызывает сервис поиска дублей в ЕК.
3. ЕК возвращает клиентские карточки, похожие на контактные данные из фронтенд-системы.  
Для каждой карточки ЕК указывает, из какой системы она пришла.
4. Фронтенд-система показывает список пользователю.
5. Пользователь выбирает подходящий вариант из списка.
6. Фронтенд-система не создает новую карточку, а использует выбранную пользователем.

На шаге 5 пользователь может выбрать карточку, для которой нет соответствия в фронтенд-системе. Тогда фронтенд-система создает новую карточку, но сохраняет в ней идентификатор карточки-дубликата. При очередной загрузке данных из фронтенд-системы ЕК по этому идентификатору определяет карточки как 100% дубликаты и объединяет их.

# ПЕРЕДАТЬ ЭТАЛОННЫЕ КАРТОЧКИ ИЗ ЕК ОБРАТНО В ИСХОДНЫЕ СИСТЕМЫ

ETL/ESB-система забирает эталонные клиентские карточки из ЕК и передает их обратно в исходные системы. ЕК не навязывает конкретный алгоритм и правила обновления. Их реализует ETL/ESB-система.

Прежде чем начинать проект по обратному распространению, ответьте на вопросы:

1. Как сопоставить эталонную клиентскую карточку с ее прообразами в исходной системе?  
ЕК сопоставляет идентификаторы исходных и эталонных клиентов. ETL/ESB-система должна корректно их обрабатывать для всех сценариев: создание, редактирование, обновление, слияние, разъединение и переслияние клиентов.
2. Как сопоставить атрибуты эталонной карточки (адреса, телефоны, документы) с атрибутами карточки в исходной системе?  
Если у клиента в исходной системе 2 мобильных телефона, а в эталонной карточке 3 телефона, как понять соответствие между ними? Что делать с «лишними» телефонами? Что делать, если для исходного телефона не нашлось эталонного?
3. Какую атрибутику в исходной системе можно менять, а какую нельзя?  
Можно ли менять паспортные данные, даже если они очевидно неправильные? Если их изменять, то какие данные указывать в отчетности для регулятора?
4. Как трактовать коды качества при обратном распространении?  
Распространять ли исправленные ФИО с кодом качества «Корректный», или лучше сохранять исходную информацию «как есть»?
5. Как устранять дубликаты и перепривязывать связанные сущности (договоры, историю обращений)?  
Удалять ли дубликаты в исходной системе? Если да, то как решить для каждой группы дубликатов: кого удалить, а кого оставить? Как быть со ссылками на дубликаты во внешних системах? Должны ли удаленные в исходной системе клиенты удаляться в ЕК?
6. Как избежать «наведенных» обновлений из исходной системы (обновлений, которые возникли в результате обратного распространения из ЕК)?  
Обратный поток с точки зрения исходной системы выглядит как обычное обновление. Это обновление передается «обратно» в ЕК. Единый клиент умеет отсеивать такие изменения (чтобы они не «зацикливались»), но его нужно настраивать под специфику данных и исходных систем.

Обратный поток из ЕК в исходные системы — сложный и чреватый ошибками процесс. Чтобы снизить риск проблем с обновлением данных, привлекайте специалистов HFLabs к согласованию технического задания для ETL/ESB-системы.

Часто вместо обратного распространения из ЕК в исходные системы используют связку сценариев **исправить клиентские данные в онлайн** и **найти потенциальные дубли в онлайн**. Они проще и меньше подвержены ошибкам интеграции.

## ПРИМЕРЫ ПРОБЛЕМ С ОБРАТНЫМ ПОТОКОМ

### Ложная валидация адресов



На первом шаге в систему-источник внесли неоднозначный адрес. CDI предложил вариант разбора, но подсветил, что адрес негарантированный.

Потом источник, получив назад этот адрес, применяет его без учета кодов качества и отправляет его обратно в CDI.

CDI видит, что адрес изменился и теперь в нём нет однозначности. В результате потеряли реальные данные.

# ИДЕНТИФИЦИРОВАТЬ КЛИЕНТА

- Пример 1. Поиск по ФИО и ДР
- Пример 2. Поиск по ФИО и соглашению
- Пример 3. Поиск по ФИО и минимальному коэффициенту соответствия
- Пример 4. Поиск по ФИО, инн, ДР и набору правил дедупликации
- Пример 5. Поиск ограниченного числа дублей по ФИО, ДР и набору правил дедупликации

## Пример 1. Поиск по ФИО и ДР

Запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <includePartyFields>true</includePartyFields>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ (среди найденных контрагентов есть объединенные):

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearchResponse>
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
          <field name="actualityDate">22.02.2012</field>
          <field name="surname">МАМОНОВ</field>
          <field name="name">СЕРГЕЙ</field>
          <field name="patronymic">ВИКТОРОВИЧ</field>
          <field name="fullNameAuthor">AL:1</field>
          <field name="fullNameQC">EDITED</field>
          <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
          <field name="birthdate">03.12.1987</field>
          <field name="birthdateAuthor">AL:1</field>
          <field name="birthdateQC">EDITED</field>
          <field name="birthdateRawSource">03.12.1987</field>
        </party>
      </matchedParty>
    </fuzzySearchResponse>
  </soapenv:Body>
</soapenv:Envelope>
```



---

```
</matchedParty>
<matchedParty>
  <matchRule>11</matchRule>
  <matchScope>100</matchScope>
  <party type="PHYSICAL" hid="2026">
    <field name="actualityDate">22.02.2012</field>
    <field name="surname">МАМОHOБ</field>
    <field name="name">CEPГEЙ</field>
    <field name="patronymic">BИKTOPOBИЧ</field>
    <field name="fullNameAuthor">AL:2</field>
    <field name="fullNameQC">EDITED</field>
    <field name="fullNameRawSource">Курко Олег Игоревич</field>
    <field name="birthdate">12.12.1989</field>
    <field name="birthdateAuthor">BT:3</field>
    <field name="birthdateQC">EDITED</field>
    <field name="birthdateRawSource">12.12.1989</field>
    <field name="maritalStatus">SINGLE_MALE</field>
    <field name="maritalStatusAuthor">AL:2</field>
    <source hid="2024" sourceSystem="AL" rawId="2"/>
    <source hid="2025" sourceSystem="BT" rawId="3"/>
  </party>
</matchedParty>
<matchedParty>
  <matchRule>23</matchRule>
  <matchScope>60</matchScope>
  <party type="PHYSICAL" hid="1026" sourceSystem="AL" rawId="5">
  <party type="PHYSICAL" hid="1026" sourceSystem="AL" rawId="8">
    <field name="actualityDate">22.02.2012</field>
    <field name="surname">МАМОHOТОБ</field>
    <field name="name">CEPГEЙ</field>
    <field name="patronymic">BИKTOPOBИЧ</field>
    <field name="fullNameAuthor">AL:5</field>
    <field name="fullNameQC">EDITED</field>
    <field name="fullNameRawSource">Мамонтов Сергей Викторович</field>
    <field name="birthdate">03.12.1987</field>
    <field name="birthdateAuthor">AL:5</field>
    <field name="birthdateQC">EDITED</field>
    <field name="birthdateRawSource">03.12.1987</field>
  </party>
</matchedParty>
</fuzzySearchResponse>
</soapenv:Body>
</soapenv:Envelope>
```

---

## Пример 2. Поиск по ФИО и соглашению

Запрос:

---

---

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="fullNameRawSource">Алешин Владимир Владимирович</field>
        <field name="birthdate">13.05.1977</field>
      </party>
      <relatedEssences>
        <party type="AGREEMENT">
          <field name="number">3935/045/0001685/01</field>
          <relation type="102"/>
        </party>
      </relatedEssences>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

---

ОТВЕТ:

---

```
<fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
  <matchedParty>
    <matchRule>30</matchRule>
    <matchScope>95</matchScope>
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"/>
  </matchedParty>
</fuzzySearchResponse>
```

---

### *Пример 3. Поиск по ФИО и минимальному коэффициенту соответствия*

---

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="name">Антан</field>
        <field name="surname">Онкуежий</field>
        <field name="birthdate">01.11.2001</field>
      </party>
      <minMatchScope>60</minMatchScope>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

---

ОТВЕТ:

---

---

---

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2851">
          <source hid="1442" sourceSystem="AL" rawId="20180323074401"/>
          <source hid="2850" sourceSystem="AL" rawId="201808174235"/>
          <source hid="2855" sourceSystem="AL" rawId="201808174236"/>
          <source hid="2857" sourceSystem="AL" rawId="201808174237"/>
        </party>
      </matchedParty>
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2884" sourceSystem="GM" rawId="564"/>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>
```

---

#### *Пример 4. Поиск по ФИО, инн, ДР и набору правил дедупликации*

---

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.
ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="name">Иван</field>
        <field name="patronymic">Игоревич</field>
        <field name="surname">Кукков</field>
        <field name="inn">111111110000</field>
        <field name="birthdate">20.11.1987</field>
      </party>
      <matchRules>
        <rule>32</rule>
        <rule>46</rule>
      </matchRules>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

---

Ответ:

---

---

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>32</matchRule>
        <matchScope>88</matchScope>
        <party type="PHYSICAL" hid="2875">
          <source hid="2873" sourceSystem="AL" rawId="1053"/>
          <source hid="2874" sourceSystem="BT" rawId="1053"/>
        </party>
      </matchedParty>
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2881" sourceSystem="GM" rawId="101223"/>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>

```

---

*Пример 5. Поиск ограниченного числа дублей по ФИО, ДР и набору правил дедубликации*

---

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:_13="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="name">Полезен</field>
        <field name="surname">Овес</field>
        <field name="birthdate">01.03.1964</field>
      </party>
      <maxCount>1</maxCount>
      <minMatchScope>60</minMatchScope>
      <matchRules>
        <rule>46</rule>
      </matchRules>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>

```

---

ОТВЕТ

---

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2752">
          <source hid="1628" sourceSystem="AL" rawId="20170925084201"/>
          <source hid="1644" sourceSystem="AL" rawId="20170925084217"/>
        </party>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>


```

---



# ЗАГРУЗИТЬ СТОП-ЛИСТЫ И СВЕРИТЬ С КЛИЕНТСКОЙ БАЗОЙ

 Сценарий для модуля «Черные списки»

 В интерфейсе CDI можно увидеть все записи стоп-листов, в том числе и те, для которых нет соответствия в базе собственных клиентов

## Предусловия

### Заполнение

Заказчик на своей стороне выполняет один из шагов:

1. выкладывает файл со списком на сервер и указывает к нему путь в задаче **listFilesImport**;
2. записывает данные из списка в **буферные таблицы**:

`LIST_BUFFER_LOAD` — информация о загружаемом списке;

`LIST_BUFFER_PHYSICAL` — информация о ФЛ из списка;

`LIST_BUFFER_LEGAL` — информация о ЮЛ из списка;

`LIST_BUFFER_ADDRESS` — адреса для ФЛ или ЮЛ;

`LIST_BUFFER_BIRTHDATE` — даты рождения для ФЛ;

`LIST_BUFFER_BIRTHPLACE` — места рождения для ФЛ;

`LIST_BUFFER_DOCUMENT` — документы для ФЛ или ЮЛ;

`LIST_BUFFER_FIO` — ФИО для ФЛ;

`LIST_BUFFER_LEGAL_NAME` — наименования для ЮЛ.

### Особенности заполнения буферных таблиц

По каждой записи списка и её атрибутам всегда выгружается полная и актуальная информация, вне зависимости от того, что из передаваемых атрибутов изменилось.

Если какой-либо из ранее загруженных атрибутов записи (документ, адрес, дата рождения и т. д.) отсутствует в списке, CDI закрывает этот атрибут в своей базе.

Все записи таблиц с атрибутами (`LIST_BUFFER_ADDRESS`, `LIST_BUFFER_BIRTHDATE`, `LIST_BUFFER_BIRTHPLACE`, `LIST_BUFFER_DOCUMENT`, `LIST_BUFFER_FIO`, `LIST_BUFFER_LEGAL_NAME`) должны обязательно содержать ссылку на запись о ФЛ или ЮЛ, к которой они привязаны (поле `ID_RECORD`).

В этих таблицах не может быть атрибутов, для которых нет соответствующих записей в таблице LIST\_BUFFER\_PHYSICAL для ФЛ и LIST\_BUFFER\_LEGAL для ЮЛ.

### *Признак готовности списка к загрузке в CDI*

Перед началом загрузки списка в буферные таблицы, необходимо выбрать очередное значение загрузки из последовательности LIST\_BUFFER\_LOAD\_SEQ. Это значение нужно записывать в ID\_LOAD при добавлении записей во все буферные таблицы.

После успешного окончания загрузки списка в буферные таблицы ФЛ, ЮЛ и их атрибутов, необходимо создать в таблице LIST\_BUFFER\_LOAD запись с использовавшимся номером загрузки в ID\_LOAD, наименованием списка в LIST\_NAME и датой актуальности списка в LIST\_ACTUALITY\_DATE.

Появление записи в LIST\_BUFFER\_LOAD служит для CDI маркером готовности данных к обработке. Если при загрузке списка ETL-процессом произошла ошибка и список может быть неполным, то запись в таблицу CDI\_BUFFER\_LIST не добавляется, CDI такие данные будет игнорировать.

### **Загрузка списков в «Единый клиент»**

«Единый клиент» загружает данные через запуск одной из задач:

- **listFilesImport** — загрузка полного цикла: файл со списком — буферные таблицы — «Единый клиент»;
- **listCommonBufferLoad** — загрузка данных только из буферных таблиц.

### *Особенности загрузки в модель «Единого клиента»*

- Задача **listFilesImport** — загружает данные из файлов списков в таблицы LIST\_BUFFER\_\*, один файл — это один список. Задача составная, вторым этапом запускается загрузка из буфера всего, что было загружено из файлов. Нужно указать директорию с файлами через параметр **sourceFileName**.  
В таблице LIST\_BUFFER\_LOAD создается загрузка со статусом INITIAL (колонка **LOAD STATUS**), которая после загрузки из файла в буферные таблицы переходит в статус READY, после загрузки в «Единый клиент» — SUCCESS/ERROR.
- Задача **listCommonBufferLoad** — циклическая: загружает поочередно списки по таблице LIST\_BUFFER\_LOAD (один список — одна строка в LIST\_BUFFER\_LOAD и строки со ссылкой **ID\_LOAD** в таблицах реквизитов и атрибутов).

## **Основной сценарий**

### **Полная кросс-сверка ФЛ**

1. С помощью задачи **physicalListCheck** система выгружает все актуальные записи ФЛ и списков ФЛ для сверки по списку правил для физических лиц.

2. Для каждой найденной пары похожих ФЛ и записи из списка ФЛ проставляется связь. Тип и наименование связи для пары ФЛ — список ФЛ зависит от правила, по которому найдены похожие записи (если найдены сразу по нескольким правилам, то используется правило с максимальным весом).

## Полная кросс-сверка ЮЛ

1. С помощью задачи **legallistCheck** система выгружает все актуальные записи ЮЛ и списков ЮЛ для сверки по списку правил для юридических лиц.
2. Для каждой найденной пары похожих ЮЛ и записи из списка ЮЛ проставляется связь. Тип и наименование связи для пары ЮЛ — список ЮЛ зависит от правила, по которому найдены похожие записи (если найдены сразу по нескольким правилам, то используется правило с максимальным весом).



# МЕХАНИЗМЫ ИНТЕГРАЦИИ

Единый клиент предоставляет ряд интерфейсов, предназначенных для организации взаимодействия с внешними системами. Краткое описание каждого из этих интерфейсов приведено ниже.

- Интерфейсы пакетной загрузки данных
- SOAP/REST-интерфейсы
- JMS-интерфейсы
- Интерфейсы экспорта данных



## Интерфейсы пакетной загрузки данных

Для первичной загрузки данных из исходных систем, а также для получения периодических обновлений из них обычно используется так называемая буферная область БД. Она состоит из буферных таблиц, которые содержат исходные клиентские записи. Как правило, используется по одной буферной таблице на каждый тип клиентов (отдельная таблица для физических лиц, отдельная — для юридических). За пополнение буферной области отвечают ETL-процедуры (специально разрабатываемые в ходе проекта), которые занимаются выборкой данных из исходных систем и записью их в буферные таблицы. Каждая запись в буферной таблице представляет собой актуальный срез данных по одному клиенту из конкретной исходной системы.

Единый клиент периодически опрашивает буферные таблицы, и, если обнаруживает еще не обработанные записи, загружает их по **стандартной процедуре**. Если какие-либо записи не удалось загрузить, Единый клиент предпринимает попытку обработать их при следующем опросе буферных таблиц. После обработки Единым клиентом данные становятся доступны через АРМ Менеджера данных, а также через интерфейсы экспорта данных.

Для загрузки данных из исходных систем могут также использоваться CSV-файлы. При таком подходе ETL-процедуры по результатам каждого сеанса выборки клиентских записей из исходных систем формируют отдельный файл и размещают его на FTP-сервере, к которому имеет доступ Единый клиент. Дальнейшая обработка этих данных осуществляется аналогично описанному выше сценарию.

## ***SOAP/REST-интерфейсы***

Единый клиент предоставляет набор SOAP/REST-методов, которые могут использоваться как для выборки (методы **search** и **getBy\***), так и изменения данных (методы **save\*** и **close**). Предоставляется также отдельный сервис для онлайн-стандартизации клиентских данных без сохранения их в Едином клиенте (этот сервис удобно использовать при первичном вводе информации по клиенту в frontend-системах).

## ***JMS-интерфейсы***

Единый клиент поддерживает загрузку исходных клиентских данных из входящей очереди сообщений, а также передачу эталонных клиентских карточек в **выходную очередь сообщений**.

## ***Интерфейсы экспорта данных***

Для пакетного экспорта информации по эталонным клиентским карточкам во внешние системы обычно используются представления базы данных, по одному представлению на каждый компонент клиентских данных:

- реквизиты (ФИО, дата рождения, семейное положения и другие одиночные атрибуты);
- адреса;
- документы;
- телефоны;
- email;
- согласия.

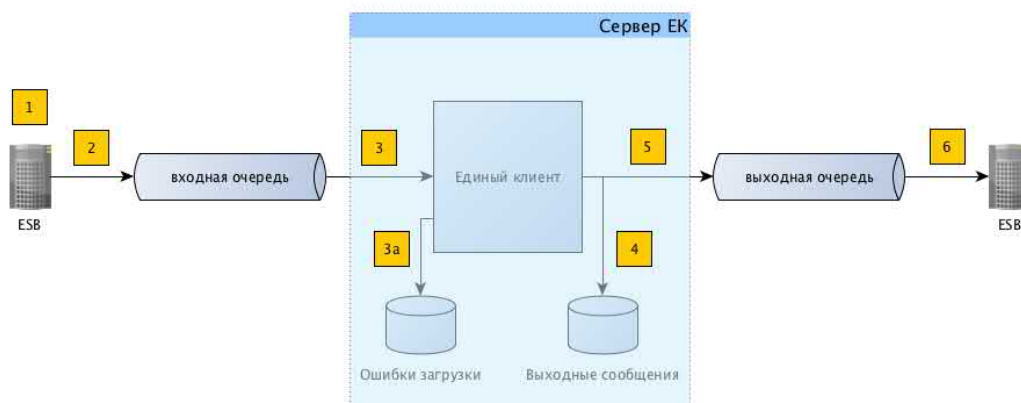
Также есть отдельное представление, описывающее соответствие между исходными клиентскими записями и "золотыми" клиентами. Оно используется для аналитики по продуктам, финансовой и прочей информации, которая не хранится в Едином клиенте, а также для организации обратного распространения клиентских данных в исходные системы.

# ПОЛНЫЙ ЦИКЛ ОБРАБОТКИ ДАННЫХ ЧЕРЕЗ JMS

В данном разделе рассмотрен сценарий работы, при котором исходные данные по контрагентам поступают в ЕК через асинхронную очередь сообщений, а результаты обработки ЕК также помещает в очередь сообщений.

- **Общий сценарий**
- **Обработка исключительных ситуаций**
  - Ошибка при записи сообщения во входную очередь
  - Потеря сообщения входной очередью до передачи его в ЕК
  - Ошибка загрузки сообщения в ЕК
  - Переполнение входной очереди
  - Ошибка записи в выходную очередь
  - Потеря сообщения выходной очередью до передачи его в ESB
  - Ошибка передачи сообщения из выходной очереди в ESB или далее по бизнес-процессу
  - Переполнение выходной очереди
  - Факт потери выходных сообщений обнаружен слишком поздно

## Общий сценарий



1. В исходной системе происходит создание или изменение клиентских данных.
2. ESB (интеграционная шина) извлекает клиентские данные из исходной системы и помещает сообщение с клиентской карточкой во входную JMS-очередь. Сообщение содержит сведения по одному клиенту, используется формат SOAP-метода *save*.
3. ЕК забирает сообщение из очереди и обрабатывает его должным образом (стандартизация, обогащение данных и т.д.), после чего переходит к обработке следующего сообщения входной очереди.

- a. Если при обработке сообщения произошла ошибка, ЕК сохраняет сообщение в своей БД и переходит к обработке следующего сообщения.
  - b. Если при обработке сообщения произошла ошибка и не удалось сохранить сообщение в БД, ЕК оставляет сообщение во входной очереди и не переходит к следующему сообщению.
  - c. Чтобы не накапливать ошибочные сообщения в БД до бесконечности, ЕК периодически удаляет "старые" сообщения (период устаревания настраивается).
4. На каждое событие изменения клиентской карточки (создание новой карточки, изменение существующей, слияние и т.д.) ЕК сохраняет соответствующее **сообщение** в своей БД.
- a. Если сохранить сообщение в БД не удалось, ЕК полностью откатывает бизнес-транзакцию и не изменяет данные клиентской карточки.
  - b. Чтобы не накапливать сообщения до бесконечности, ЕК периодически удаляет "старые" сообщения (период устаревания настраивается).
5. Одновременно с п. 4 ЕК помещает соответствующее **сообщение** в выходную JMS-очередь.
6. ESB забирает сообщение из выходной очереди и обрабатывает его в соответствии со своим бизнес-процессом.

## **Обработка исключительных ситуаций**

### **Ошибка при записи сообщения во входную очередь**

Решается вне ЕК (средствами ПО ESB и сервера очередей).

После идентификации "проблемных" сообщений, их следует повторно направить во входную очередь.

### **Потеря сообщения входной очередью до передачи его в ЕК**

Решается вне ЕК (средствами ПО сервера очередей).

Если средствами сервера очередей решить не удалось: после идентификации временного интервала, на котором наблюдались "потери" сообщений, соответствующие клиентские данные можно передать в ЕК с помощью механизма **пакетной загрузки** через БД.

### **Ошибка загрузки сообщения в ЕК**

См. шаг сценария За. Ошибочные сообщения ЕК сохраняет в своей БД. Факт ошибки фиксируется в журнале ЕК и направляется администратору по электронной почте. После устранения причины ошибки администратор загружает "проблемные" сообщения с помощью **специальной задачи** (запускается вручную через АРМ Администратора).

## ***Переполнение входной очереди***

Если БД ЕК недоступна в течение длительного времени (см. шаг сценария 3b), может произойти переполнение входной очереди и часть сообщений будет утеряна (если очередь настроена соответствующим образом). В этом случае после возобновления работы ЕК данные за "проблемный" период следует передать в ЕК с помощью механизма **пакетной загрузки** через БД.

## ***Ошибка записи в выходную очередь***

Факт ошибки фиксируется в журнале ЕК и направляется администратору по электронной почте. После устранения причины ошибки администратор повторно выгружает "проблемные" сообщения с помощью **специальной задачи** (запускается вручную через АРМ Администратора).

## ***Потеря сообщения выходной очередью до передачи его в ESB***

Решается вне ЕК (средствами ПО сервера очередей).

Если средствами сервера очередей решить не удалось: после идентификации временного интервала, на котором наблюдались "потери" сообщений, администратор повторно выгружает сообщения из ЕК с помощью специальной задачи (запускается вручную через АРМ Администратора).

## ***Ошибка передачи сообщения из выходной очереди в ESB или далее по бизнес-процессу***

Факт ошибки должен быть зафиксирован средствами ПО ESB или сервера очередей и далее передан администраторам.

После устранения причины ошибки администратор повторно выгружает сообщения из ЕК с помощью специальной задачи (запускается вручную через АРМ Администратора).

## ***Переполнение выходной очереди***

Если ESB недоступна в течение длительного времени, может произойти переполнение выходной очереди и часть сообщений будет утеряна (если очередь настроена соответствующим образом). В этом случае после возобновления работы ESB администратор повторно выгружает сообщения из ЕК с помощью специальной задачи (запускается вручную через АРМ Администратора).

## ***Факт потери выходных сообщений обнаружен слишком поздно***

Если потеря сообщений выявлена уже после того, как эти сообщения удалены из БД ЕК, то соответствующие клиентские данные можно выгрузить из ЕК с помощью механизма **пакетного экспорта** через представления БД.

# РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

- Процесс работы дата-стюардов и типовые вопросы
  - Процесс работы дата-стюардов
  - Типовые вопросы при работе дата стюардов
- Требования к менеджеру данных
- Общие сценарии
  - Начало работы с системой
  - Работа со списком контрагентов
  - Просмотр карточки контрагента
    - Просмотр карточки. Общая информация
    - Просмотр карточки. Документы
    - Просмотр карточки. Адреса
    - Просмотр карточки. Телефоны
    - Просмотр карточки. Email
    - Просмотр карточки. Аккаунты
    - Просмотр карточки. Документы ТС
    - Просмотр карточки. Согласия
    - Просмотр карточки. Дополнительные атрибуты
    - Просмотр карточки. Табличный режим просмотра атрибутов
    - Просмотр карточки. Связи
    - Просмотр карточки. История
    - Просмотр карточки. Исходные
    - Просмотр карточки. Отображение атрибутов
    - Просмотр карточки. Отклоненные дубликаты
    - Просмотр карточки. Обновление информации о дубликатах
    - Просмотр карточки. Связанные атрибуты
    - Просмотр карточки. Обращение в поддержку
    - Просмотр карточки. Уведомления об обновлении карточки
  - Работа со списком групп дубликатов
    - Отображение дубликатов ЮЛ и ЕГРЮЛ
    - Отобрать группы с отклоненными дубликатами
    - Отобрать недавно созданные дубли
    - Поиск дубликатов по вхождению в реестр
    - Поиск дубликатов по вхождению в сегмент
    - Поиск дубликатов по исходной системе
  - Просмотр группы дубликатов
  - Работа с черными списками

Выход в АРМ Администратора  
Подтверждение и отклонение связей

- Сценарии работы менеджера данных
  - Проверка информации о контрагентах
    - Копирование адресов по образцу
  - Работа с группами дубликатов
    - Подтверждение или отклонение дубликатов
    - Прогноз результатов слияния
  - Работа с конфликтующими изменениями
  - Разъединение дубликатов контрагентов
- Сценарии работы оператора
- Расширенный поиск
  - Поиск по реквизитам контрагента
  - Поиск по реквизитам юрлица
  - Поиск по адресам
  - Поиск по телефонам
  - Поиск по email
  - Поиск по документам
  - Поиск по дубликатам
  - Поиск по группам дубликатов
  - Поиск по аккаунтам
  - Поиск по реквизитам любого контрагента
- Описание кодов качества
  - Коды качества полей документа ФЛ (ДУЛ)
    - Код качества документа ФЛ
    - Отображение кода качества документа
    - Маркеры проверок документа
    - Статус актуальности документа
    - Отображение гранулярных кодов качества и маркеров ДУЛ
    - Статус недействительности для документа
    - Скоринг полноты ДУЛ
    - Код качества ИНН ФЛ
  - Коды качества email
    - Код качества email
    - Маркеры проверок и изменений email
    - Причины признания email плохим
  - Коды качества адреса
    - Код валидации адреса

- Проставление дополнительных кодов проверок
- Отображение расширения валидатора в интерфейсе
- Код проверки адреса (устаревшие значения)

Код полноты адреса

Маркер изменения адреса

Маркер невалидности адреса

Код качества координат

Статус разбора адреса

- Отображение в интерфейсе
- Проставление статусов преобразований

Соответствие уровню иерархии в ФИАС

Коды качества телефона

Код качества телефона

Маркеры проверок и изменений телефона

Причины, почему телефон не распознан

Коды качества ФИО и пола

Общий код качества ФИО

Статус распознавания ФИО

Статус распознавания ФИО (без гранулярных КК ФИО)

Причины отправки ФИО на ручной разбор

Маркеры проверки и изменения ФИО

Маркеры проверки и изменения фамилии

Маркеры проверки и изменения имени

Маркеры проверки и изменения отчества

Код качества пола

Прочие коды качества

Код качества наименования ЮЛ

Код качества ИНН ЮЛ

Код качества КИО

Код качества КПП

Код качества ОГРН

Код качества ОКПО

Код качества СНИЛС

Код качества ОКВЭД

Код качества VIN-номера

Код качества даты рождения

Популярность атрибута

Признак террориста

Код качества SWIFT



Код качества БИК  
Код качества реквизита  
Коды качества документа

- Горячие клавиши

# ПРОЦЕСС РАБОТЫ ДАТА-СТЮАРДОВ И ТИПОВЫЕ ВОПРОСЫ

- Общие сведения
  - Основные обязанности дата-стюардов:
  - Пре-реквизиты для начала работы дата-стюардов в «Едином клиенте»
- Разбор конфликтов
- Разбор серой зоны по дубликатам
  - Стратегии работы с серой зоной
  - Типовой сценарий разбора серой зоны
- Выверка данных перед их использованием
- Разбор инцидентов качества данных
  - Сценарий 1. Клиент не видит продукт в личном кабинете
  - Сценарий 2. Клиент видит чужой продукт в личном кабинете
  - Сценарий 3. Ошибка в персональных/контактных данных клиента
  - Сценарий 4. Неправильный скоринг
- Типовые вопросы при работе дата стюардов
  - Доступ в систему
    - Как попасть в систему?
    - Что нужно для доступа нового человека к системе?
    - Не могу залогиниться из-за ошибки «Неверный пароль»
    - Система недоступна
  - Разбор серой зоны
    - Как понять, какие данные выбрать победителем при признании карточек дубликатами?
  - Поиск клиента
    - Почему я не вижу клиента в ЕК? В исходной системе он есть, а в ЕК нет?
    - Почему не объединились клиенты? Визуально видно, что это одинаковые записи, но они остались в системе как разные карточки.
  - Анализ данных на карточке клиента
    - Почему в Золотой записи выиграла те или иные данные. Что делать, если они не верные?
    - В карточке клиента я вижу данные другого человека, что делать?
    - Почему я вижу, что в исходной системе у клиента есть адрес/телефон/документ, а в ЕК этих данных нет?
    - Почему не проходят обновления на карточку? В исходной системе изменили данные, а в ЕК их не видно, почему?
    - Не могу отредактировать карточку клиента или внести новые данные клиента

Система неправильно распознает данные после нажатия на кнопку «Стандартизировать»

Система выдала ошибку

Появилась ошибка «Что-то пошло не так»

При работе с дубликатами появилась ошибка «Неактуальная группа дубликатов»

## Общие сведения

Дата-стюарды — это люди, которые работают в интерфейсе «Менеджера данных» и разбирают данные, которые «Единый клиент» пометил плохими кодами качества, а также те ситуации, которые «Единый клиент» не смог разрешить автоматически.

Обычно это группа людей от 5 до 20 человек, у которой есть руководитель. Задачи дата-стюардов требуют разной квалификации, поэтому оптимальный состав группы — это 20% опытных квалифицированных сотрудников, которые могут решить любые проблемы с качеством данных — старшие дата-стюарды, и 80% обычных сотрудников, которые работают по шаблонам, а те проблемы, которые не укладываются в шаблоны, отдают старшим дата-стюардам.

### **Основные обязанности дата-стюардов:**

1. Регулярный разбор конфликтов.
2. Разбор серой зоны по дубликатам.
3. Выверка данных перед их использованием.
4. Разбор инцидентов качества данных.

### **Пре-реквизиты для начала работы дата-стюардов в «Едином клиенте»**

1. «Единый клиент» интегрирован с LDAP-системой Заказчика, распределяющей права учетных записей пользователей.
2. На стороне Заказчика выделен администратор для обслуживания и поддержки системы.
3. Администраторы прошли обучение по сопровождению системы у сотрудников HFLabs.
4. Дата-стюарды прошли обучение по работе с системой у сотрудников HFLabs.
5. Выделен руководитель группы дата-стюардов, который будет основным контактным лицом для HFLabs по работе с данными.
6. Для администратора и руководителя группы дата-стюардов созданы учетные записи в системе сопровождения HFLabs (Jira).

## Разбор конфликтов

**И** **Ответственность: высокая.** Неверные действия приведут к тому, что другой человек может увидеть продукты и персональные данные клиента, а также к созданию кентавров в базе «Единого клиента». «Кентавры» — объединенные карточки с перемешанными данными разных клиентов.

**Сложность: высокая.** Необходимо расследовать, почему в исходной системе или в процессе интеграции были сильно изменены данные клиента и принимать решение, что делать с проблемным обновлением.

Бывают ситуации, когда на исходную карточку, уже сохраненную в «Едином клиенте», приходит обновление, содержащее абсолютно иную информацию, нежели хранилась в карточке до этого. Например, раньше в карточке было указано ФИО — Смирнова Наталия Петровна и ДР — 12.07.1986, а в обновлении ФИО — Керчинский Игорь Дмитриевич с ДР — 04.12.1993. В таких случаях «Единый клиент» блокирует обновление, т.к. оно похоже на фрод или ошибку интеграции и может содержать вперемешку данные разных клиентов. Такая ситуация называется конфликтом обновления.

Частота возникновения зависит от качества процессов работы с данными внутри компании. В идеальном мире должно быть ровно 0 таких ситуаций. В реальности — от 5 до 100 ситуаций в месяц.

В «Едином клиенте» есть 2 стратегии разбора конфликтов:

**Автоматическая** — «Разлить и обновить» — карточка, на которую пришло конфликтное обновление, автоматически отсоединяется от объединенной записи, после чего автоматически обновляется на новые данные, даже если они принадлежат другому человеку.

Данная стратегия используется в компаниях телекоммуникационной индустрии, где передача сим-карты от одного человека другому — стандартный процесс и нередки случаи, когда вместо заведения новой карточки клиента просто перебиваются данные на старой клиентской карточке, которая привязана к передаваемой sim-карте.

При этой стратегии конфликты не отображаются в интерфейсе, и дата-стюарды не участвуют в разрешении конфликтов.

**Ручная** — при возникновении конфликта «Единый клиент» запоминает конфликт, блокирует любые изменения как исходной карточки, на которую поступил конфликт, так и объединенной, куда влита карточка с конфликтом обновления. В интерфейсе «Менеджера данных» обновляется счетчик текущих конфликтов для привлечения внимания дата-стюарда.

Данная стратегия часто используется в банках, где резкое изменение данных клиента с одного человека на другого недопустимо и обычно сигнализирует о мошенничестве или об ошибке на стороне исходной системы или интеграции.

В случае включения данной стратегии необходимо учитывать, что конфликты должны быть разобраны оперативно (желательно в тот же день, когда они возникли), так как пока не разрешён конфликт, объединенная карточка заблокирована от всех других обновлений (даже из других исходных систем) и слияний с другими карточками, даже если они являются гарантированными дублями к этой карточке.

В страховых компаниях в зависимости от процессов внутри компании может использоваться любая из этих двух стратегий.

## Разбор серой зоны по дубликатам

**i** Ответственность: **высокая**. Неверные действия приведут к тому, что другой человек может увидеть продукты и персональные данные клиента, а также к созданию кентавров в базе «Единого клиента». «Кентавры» — объединенные карточки с перемешанными данными разных клиентов.

Сложность: средняя, высокая. Необходимо получать недостающую информацию из исходных систем, сканов документов, архива. Уметь сопоставлять информацию из разных источников.

### Стратегии работы с серой зоной

Обычно сразу после внедрения CDI появляется большая серая зона потенциальных дублей. Это записи, которые похожи друг на друга, но недостаточно сильно, чтобы система автоматически признала их дублями. Объем таких записей колеблется от 500 тысяч до десятков миллионов. Ручной разбор абсолютно всех записей из этой зоны обычно невозможен из-за недостатка в данных и нецелесообразен, так как далеко не все записи относятся к активной базе клиентов. Выделяют несколько стратегий работы с серой зоной:

1. Оставить как есть. Ничего не делать. В этом случае серая зона постоянно увеличивается в процессе жизни системы. Анализ записей серой зоны происходит только в случае обращения клиентов по инцидентам.
2. Разбирать новое, старое оставить как есть. В этом случае потенциальные дубликаты, образовавшиеся при заливке инициализирующего объема записей в «Единый клиент», остаются без изменений. Дата-стюарды подключаются к разбору потенциальных дубликатов, которые образуются в результате обновления записей в процессе жизни системы или появления новых записей из исходных систем. Такой подход позволяет зафиксировать размер серой зоны (не происходит увеличения) и разрешать проблемы потенциальных дублей по активным клиентам по мере поступления информации от них. Данная стратегия является наиболее популярной у Заказчиков.
3. Разбирать новое и наиболее важный кусочек инициализирующей загрузки. Аналогично п. 2, но среди первичного объема серой зоны выделяют наиболее важных клиентов (по статусу, активности или каким-то другим критериям) и разбирают вручную те группы, в которые попали эти клиенты вдобавок к разбору вновь появляющихся потенциальных дубликатов.
4. Разобрать вообще всю серую зону.

## Типовой сценарий разбора серой зоны

1. Дата-стюард отбирает через фильтр те группы, с которыми ему необходимо работать, например, «Потенциальные дубликаты, созданные вчера».
2. Открывает первую группу и анализирует данные на карточках.
3. В случае, если информации достаточно для принятия решения «Дубликат» или «Не дубликат», принимает соответствующее решение.
4. Если информации недостаточно, то:
  - a. Проверяет наличие сканов документов клиентов, приложенных к карточкам, и информацию в них.
  - b. Ищет информацию в карточках клиентов исходных систем.
  - c. Ищет информацию в карточках договоров исходных систем.
  - d. Запрашивает информацию из архива.
  - e. Отдает карточку для разбора старшему дата-стюарду, если самостоятельно не удалось решить проблему.
5. В случае, если для получения дополнительной информации необходимо время, дата-стюард принимает решение «Отложить» с указанием причины: запрошена информация в архиве, передано старшему дата-стюарду, недостаточно информации для принятия решения и т.п.
6. После получения новой информации дата-стюард возвращается к отложенной карточке и принимает решение, являются ли карточки дубликатами, при необходимости добавляет недостающие данные в карточку.
7. Если принято решение «Дубликат», дата-стюард проверяет, какая информация попадет в золотую карточку на экране «Прогноз слияния». При необходимости корректирует автоматически выбранные системой данные для золотой карточки.
8. Дата-стюард подтверждает объединение карточек.
9. Дата-стюард переходит к разбору следующей группы потенциальных дубликатов.

## Выверка данных перед их использованием

 **Ответственность:** низкая

**Сложность:** низкая, средняя

Часто перед проведением рекламных кампаний, маркетинговых рассылок или сдачей отчёта необходимо полностью выверить данные небольшой части клиентов, которые попадут в эту кампанию, рассылку или отчёт. Для этого в системе «Единый клиент» преднастроена группа фильтров «На ручной разбор». В эти фильтры попадают клиенты, в данных которых система не уверена.

Типовой сценарий проверки данных дата-стюардом:

1. Дата-стюард выставляет фильтр «На ручной разбор» для того типа атрибутов, которые надо выверить.
2. Дата-стюард дополнительно отфильтровывает по идентификаторам те записи клиентов, которые попадали в целевую группу (подготовленную рассылку, отчёт и т.п.).
3. Дата-стюард открывает карточку клиента.
4. Дата-стюард переходит на атрибут, который надо проверить.
5. Дата-стюард раскрывает подробности атрибута и проверяет информацию:
  - a. Если дата-стюард согласен с предположениями, которые сделала система при разборе данных, то он подтверждает изменения, произведенные системой, нажимая кнопку «Подтвердить».
  - b. Если дата-стюард видит, что система распознала данные неверно, то:
    - i. Он вносит необходимые изменения в данные: заполняет поля, изменяет данные в полях, удаляет лишнюю информацию.
    - ii. Нажимает кнопку «Стандартизировать».
    - iii. Система показывает дата-стюарду результат стандартизации измененных данных.
    - iv. Если дата-стюард согласен с вариантом разбора новых данных системой, он нажимает кнопку «Подтвердить».
    - v. Если система неверно распознала новые данные, то:
      1. Дата-стюард вносит необходимые исправления.
      2. Подтверждает данные кнопкой «Подтвердить» без повторной стандартизации.

## Разбор инцидентов качества данных

**i** **Ответственность:** средняя

**Сложность:** средняя

Дата-стюарды работают по инцидентам:

1. Клиент не видит продукт в личном кабинете.
2. Клиент видит незнакомый продукт в личном кабинете.
3. Клиент видит ошибку в данных.
4. Неправильный скоринг.

### **Сценарий 1. Клиент не видит продукт в личном кабинете**

1. Клиент обращается в поддержку ЛК.
2. Поддержка адресует вопрос дата-стюардам.
3. Дата-стюард проверят данные клиента:

- a. Если обе записи есть и являются потенциальными дубликатами — объединяет их.
- b. Если записи не являются потенциальными дубликатами — выясняет причину ошибки:
  - i. Если указаны разные клиентские данные в карточках, но на самом деле они относятся к одному клиенту, что явствует из запроса клиента, то исправляет данные в карточке таким образом, чтобы их хватило для автоматического объединения этих записей, когда отработает регулярная задача поиска и объединения дубликатов.
  - ii. Если клиенты уже содержат похожую информацию и должны были найтись как дубли, но не нашлись, передаёт инцидент руководителю группы для отправки запроса и данных в HFLabs.
- c. Одного или обоих записей нет в ЕК передаёт запрос команде интеграции для проверки, были ли переданы такие данные в ЕК.

## ***Сценарий 2. Клиент видит чужой продукт в личном кабинете***

1. Клиент обращается в поддержку ЛК.
2. Поддержка адресует вопрос дата-стюардам.
3. Дата-стюард проверяет данные клиента:
  - a. Если в золотой записи клиента ошибочно присутствуют карточки других клиентов, то вручную отсоединяет эти карточки.
  - b. Если невозможно идентифицировать чужие карточки по клиентским данным, то анализирует данные в исходной системе: возможно, что неверно отработала интеграция, и договор привязали к чужой карточке клиента. Для решения проблемы с отвязкой договора в ЕК передает запрос руководителю группы для создания инцидента в системе сопровождения HFLabs
  - c. Если карточка не найдена в ЕК, передает запрос команде интеграции.

## ***Сценарий 3. Ошибка в персональных/контактных данных клиента***

1. Клиент обращается в поддержку ЛК.
2. Поддержка адресует вопрос дата-стюардам.
3. Дата-стюард исправляет данные в ЕК и уточняет источники, из которых пришли ошибочные данные.
4. Дата-стюард отправляет данные по ошибке в источник для исправления в первичной записи.

## ***Сценарий 4. Неправильный скоринг***

1. Клиент обращается в поддержку компании с жалобой о неправильном скоринге.



2. Если причина в том, что не все данные клиента были учтены, то поддержка адресует вопрос дата-стюардам.
3. Дата-стюард проверяет данные в карточке клиента ЕК и при необходимости:
  - а. Анализирует серую зону: возможно что старые карточки клиента не были учтены, т.к. попали в негарантированные дубли. При необходимости подтверждает дубликаты и объединяет карточки.
  - б. Исправляет данные в ЕК и уточняет источники, из которых пришли ошибочные данные.  
Дата-стюард отправляет данные по ошибке в источник для исправления в первичной записи.

## Типовые вопросы при работе дата стюардов

HFLabs как вендор оказывает третью линию поддержки по вопросам работы системы и качества обработанных данных.

Все запросы в поддержку HFLabs от дата-стюардов передаются через вторую линию, роль которой выполняют администраторы приложения заказчика или руководитель группы дата-стюардов, если вопросы касаются качества данных.

Администратор перед отправкой запроса в HFLabs собирает **диагностическую информацию** по нужным клиентам.

### ***Доступ в систему***

#### ***Как попасть в систему?***

1. Для того чтобы попасть в систему «Единый клиент» необходимо перейти по ССЫЛКЕ.
2. Для входа в систему необходимо ввести логин и пароль от доменной учетной записи.

#### ***Что нужно для доступа нового человека к системе?***

Для предоставления доступа к системе «Единый клиент» необходима заявка в службу технической поддержки на включение сотрудника в доменную группу с указанием ролей, которые необходимы сотруднику:

1. Только просмотр карточек.
2. Просмотр и редактирование карточек.
3. Работа с потенциальными дубликатами.

## *Не могу залогиниться из-за ошибки «Неверный пароль»*

1. Для входа в систему необходимо ввести логин и пароль от доменной учетной записи.
2. Если дата-стюард видит ошибку «Неверный пароль», необходимо убедиться, что учетная запись пользователя активна и включена в необходимую доменную группу. Обратитесь к администратору системы.

## *Система недоступна*

«Единый клиент» — отказоустойчивая и стабильная система, однако бывают технические моменты связанные с сетевой доступностью, работоспособностью инфраструктуры и проведением технических работ.

Решение ? Перезагрузить станцию в браузере без кеша через Ctrl+F5 или Ctrl+R. Если ошибка не исчезла, обратиться к администратору системы.

## *Разбор серой зоны*

### *Как понять, какие данные выбрать победителем при признании карточек дубликатами?*

Нужно составить «шпаргалки», в которых:

- Описать логику обработки слияния дубликатов по кейсам. Например, если в дубликатах 2 клиента со связанными автомобилями и водительским удостоверением и у одного дата рождения 1985, а у второго 2005, то при прочих равных выбираем победителем дату рождения 1985 год. Т.е. предполагаем, что в 14 лет у человека не может быть ВУ и автомобиля.
- Составить для дата-стюардов матрицу приоритетных систем (если в группе дубликатов есть карточка из системы А и из системы Б, то данным из А верить больше, чем из Б).

## *Поиск клиента*

### *Почему я не вижу клиента в ЕК? В исходной системе он есть, а в ЕК нет?*

1. Попробовать найти клиента по другим данным:
  - a. по идентификатору в исходной системе с указанием кода исходной системы, например, SYS:124578
  - b. по фамилии, имени и дате рождения в формате **Фамилия Имя ДД.ММ.ГГГГ** без других разделителей. (**Фамилия, имя, дд-мм-гггг** — неправильно!)
2. Убедиться, что в исходной системе клиент не отмечен ни одним из vip-признаков. Для работы с vip-клиентами в системе может быть предусмотрена отдельная роль, остальным сотрудникам данные по vip-клиентам в этом случае будут не видны, и через поиск найти такого клиента не удастся.

3. Если клиента не удалось найти, то уточнить дату создания клиента в исходной системе.
  - a. Если он создан «сейчас», возможно создание еще не дошло до ЕК.
  - b. Если клиентская запись в исходной системе старше 24 часов при пакетной интеграции или старше 5 минут при онлайн-интеграции, то обратиться в поддержку HFLabs с указанием:
    - i. Полной информации по клиенту и его идентификатора в исходной системе.
    - ii. Описанием, каким способом пробовали найти клиента в «Едином клиенте».

### ***Почему не объединились клиенты? Визуально видно, что это одинаковые записи, но они остались в системе как разные карточки.***

1. Если записи найдены как потенциальные дубликаты, значит, системе не хватило данных для их автоматического объединения. В этом случае необходимо объединить записи вручную, подтвердив, что они являются дублями на экране групп дубликатов.
2. Если записи похожие или одинаковые, но при этом система не считает их дубликатами:
  - a. Проверить, что записи не были ранее отмечены дата-стюардом как «Не дубликаты».
  - b. Если запись создана или обновлена менее 24 часов назад, то дождаться регулярной (обычно ночной) задачи по поиску и объединению дубликатов.
  - c. Если запись создана или обновлена последний раз более 24 часов назад, то собрать всю диагностическую информацию по обеим записям и обратиться в поддержку HFLabs.

## ***Анализ данных на карточке клиента***

### ***Почему в Золотой записи выиграла те или иные данные. Что делать, если они не верные?***

ЕК выбирает победителем более актуальные или более достоверные с точки зрения заложенных в него алгоритмов данные. Например, более свежий по дате выдачи паспорт вероятнее всего победит более старый.

В реальности бывают разные ошибки и опечатки в данных, и паспорт, выданный раньше, может оказаться правильным. В этом случае дата-стюард исправляет данные в золотой записи, если он уверен в своей правоте.

### ***В карточке клиента я вижу данные другого человека, что делать?***

1. Выяснить, какая исходная карточка является источником чужих данных.
2. Определить, является ли исходная карточка с чужими данными карточкой другого клиента:
  - a. Если это карточка другого клиента, то:
    - i. Отсоединить данную карточку из объединенной записи. Информация в объединенной записи обновится автоматически.
    - ii. Проверить, кто подтвердил, что карточки являются дубликатами:

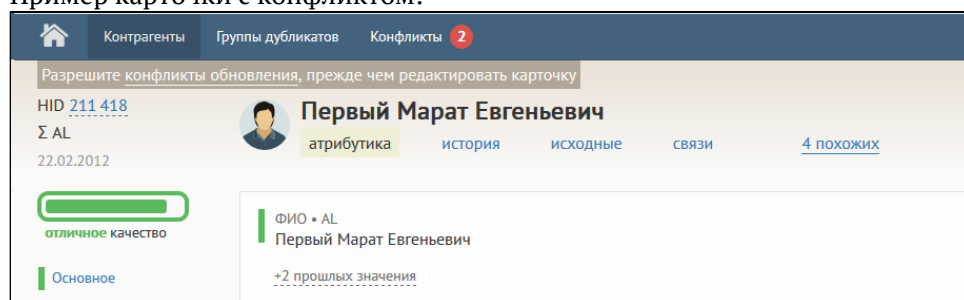
1. Если дубликаты подтверждены автоматически (автор подтверждения — HFLabs), то отправить запрос с диагностическими данными в поддержку HFLabs.
  2. Если дубликаты были подтверждены вручную, то отправить инцидент на руководителя дата-стюардов.
- b.** Если чужие данные попали в исходную карточку по ошибке, а сама карточка принадлежит рассматриваемому клиенту, то:
- i.** Удалить чужие данные на объединенной карточке.
  - ii.** Отправить запрос на исправление данных в исходную систему.

### ***Почему я вижу, что в исходной системе у клиента есть адрес/телефон /документ, а в ЕК этих данных нет?***

Если атрибут в исходной системе осмысленно заполнен, он должен быть в ЕК.

1. Проверить, что значение не является мусорным или тестовым с точки зрения «Единого клиента» через документацию по правилам стандартизации ненайденного атрибута. Как правило, тестовые и придуманные значения удаляются системой на этапе очистки данных.
2. Проверить, что на обновляемой клиентской карточке нет **конфликтов**. Конфликты блокируют карточку от любых последующих обновлений.

Пример карточки с конфликтом:



Если обнаружен конфликт, то:

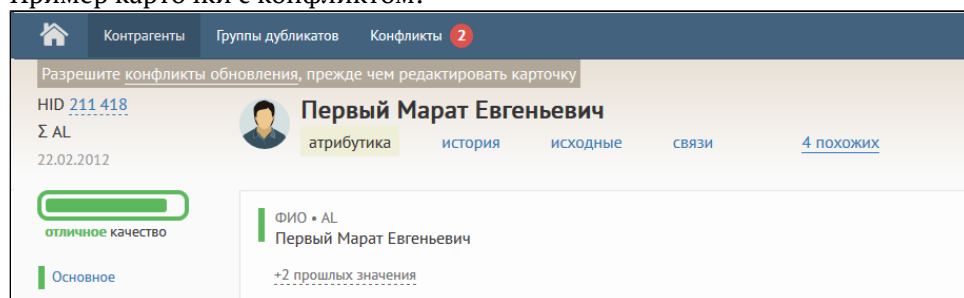
- a.** Разрешить его в соответствии с разделом «Работа с конфликтами».
  - b.** Проверить, появились ли необходимые данные на карточке после разрешения конфликта.
- 3.** Проверить, что данные не были ранее удалены вручную из системы:
- a.** Зайти на вкладку «история».
  - b.** Попробовать найти искомые данные.
  - c.** Если данные были удалены вручную, то:
    - i.** Выяснить причину удаления.
    - ii.** Если данные удалены верно в «Едином клиенте», составить запрос в исходную систему на удаление данных.
    - iii.** Если данные удалены ошибочно, то собрать диагностическую информацию и обратиться в поддержку HFLabs.

4. Если предыдущие шаги не помогли найти причину отсутствия данных, то собрать диагностическую информацию и обратиться в поддержку HFLabs.

### ***Почему не проходят обновления на карточку? В исходной системе изменили данные, а в ЕК их не видно, почему?***

1. Проверить, что новое значение:
  - a. Не является мусорным или тестовым с точки зрения «Единого клиента» через документацию по правилам стандартизации ненайденного атрибута. Как правило, тестовые и придуманные значения удаляются системой на этапе очистки данных.
  - b. Не является пустым. Для большинства полей в «Едином клиенте» настроены правила, которые препятствуют обновлению на пустое значение (обычно это означает ошибку в интеграции).
    - i. Проверить, что правила обновления для этого поля допускают обновление на пустое значение.
    - ii. Если правила не допускают обновления на пустое значение, но при этом действительно есть ситуации, когда значение может быть удалено из поля в источнике, обратиться в поддержку HFLabs с описанием, почему поле должно допускать обновление на пустые значения.
2. Проверить, что на обновляемой клиентской карточке нет **конфликтов**. Конфликты блокируют карточку от любых последующих обновлений.

Пример карточки с конфликтом:



Если обнаружен конфликт, то:

- a. Разрешить его в соответствии с разделом «Работа с конфликтами».
  - b. Проверить, появились ли необходимые данные на карточке после разрешения конфликта.
3. Если предыдущие шаги не помогли найти причину отсутствия данных, то собрать диагностическую информацию и обратиться в поддержку HFLabs.

### ***Не могу отредактировать карточку клиента или внести новые данные клиента***

Для поиска причин этой ситуации необходимо:

1. Проверить, что открыта именно золотая запись клиента, а не исходная карточка. В случае, если открыта исходная карточка, система напишет в верхнем левом углу предупреждение «Неактуальная карточка: влита в HID XXXX». Решение ? Перейти на актуальную карточку по предложенной системой ссылке.
2. Убедиться, что другой сотрудник не работает сейчас с этой карточкой. В этом случае в левом верхнем углу экрана будет сообщение «Карточка редактируется пользователем <username>». Одновременное редактирование карточек запрещено. Решение ? Попросить другого сотрудника закрыть карточку или перейти на работу со следующей карточкой.
3. Убедиться, что карточка не закрыта ранее в системе. В случае просмотра закрытой карточки информация об этом будет в левом верхнем углу. Редактирование закрытых карточек запрещено. Решение ? Выяснить, зачем нужно редактировать закрытую карточку. Если карточка закрыта по ошибке, то собрать диагностическую информацию и обратиться в поддержку HFLabs.
4. Убедиться в отсутствии конфликта на карточке. При его наличии в левом верхнем углу будет предупреждение «Разрешите конфликты обновления, прежде чем редактировать карточку». Решение ? Перейти на конфликт по ссылке из сообщения и разрешить конфликты в соответствии с разделом «Работа с конфликтами».
5. Проверить, есть ли возможность редактировать другие карточки:
  - a. Если нет возможности отредактировать ни одну карточку, то обратиться к администратору системы с просьбой предоставить права на редактирование карточек. В системе настроены разные роли:
    - i. Роль оператора — этой роли доступен только просмотр карточек.
    - ii. Роль менеджера данных — для редактирования карточек.
  - b. Если проблема с редактированием только одной или нескольких карточек, то собрать диагностическую информацию по проблемным карточкам и обратиться в поддержку HFLabs.

### ***Система неправильно распознает данные после нажатия на кнопку «Стандартизировать»***

В этом случае необходимо внести данные в гранулярные поля так, как дата-стюард считает правильным (например, на основе скана документа) и нажать кнопку «Подтвердить», не проводя стандартизацию данных.

Система сохранит введённые данные без выполнения алгоритма очистки и стандартизации ровно в том виде, как их внёс дата-стюард.

### ***Система выдала ошибку***

#### ***Появилась ошибка «Что-то пошло не так»***

В системе возникла непредвиденная ситуация, но не стоит беспокоиться. В 99,9% случаев перезагрузка страницы в браузере помогает ее исправить. Если ошибка не исчезла после

перезагрузки или появляется несколько раз в течение дня, необходимо обратиться к администратору системы.

### ***При работе с дубликатами появилась ошибка «Неактуальная группа дубликатов»***

Чаще всего эта ошибка говорит о том, что клиентские запись уже объединены либо системой автоматически, либо другим дата-стюардом. Следует нажать кнопку «Следующая группа» и перейти к разбору следующей необработанной группы.

# ПРОЦЕСС РАБОТЫ ДАТА-СТЮАРДОВ

- Общие сведения
  - Основные обязанности дата-стюардов:
  - Пре-реквизиты для начала работы дата-стюардов в «Едином клиенте»
- Разбор конфликтов
- Разбор серой зоны по дубликатам
  - Стратегии работы с серой зоной
  - Типовой сценарий разбора серой зоны
- Выверка данных перед их использованием
- Разбор инцидентов качества данных
  - Сценарий 1. Клиент не видит продукт в личном кабинете
  - Сценарий 2. Клиент видит чужой продукт в личном кабинете
  - Сценарий 3. Ошибка в персональных/контактных данных клиента
  - Сценарий 4. Неправильный скоринг

## Общие сведения

Дата-стюарды — это люди, которые работают в интерфейсе «Менеджера данных» и разбирают данные, которые «Единый клиент» пометил плохими кодами качества, а также те ситуации, которые «Единый клиент» не смог разрешить автоматически.

Обычно это группа людей от 5 до 20 человек, у которой есть руководитель. Задачи дата-стюардов требуют разной квалификации, поэтому оптимальный состав группы — это 20% опытных квалифицированных сотрудников, которые могут решить любые проблемы с качеством данных — старшие дата-стюарды, и 80% обычных сотрудников, которые работают по шаблонам, а те проблемы, которые не укладываются в шаблоны, отдают старшим дата-стюардам.

## Основные обязанности дата-стюардов:

1. Регулярный разбор конфликтов.
2. Разбор серой зоны по дубликатам.
3. Выверка данных перед их использованием.
4. Разбор инцидентов качества данных.



## Пре-реквизиты для начала работы дата-стюардов в «Едином клиенте»

1. «Единый клиент» интегрирован с LDAP-системой Заказчика, распределяющей права учетных записей пользователей.
2. На стороне Заказчика выделен администратор для обслуживания и поддержки системы.
3. Администраторы прошли обучение по сопровождению системы у сотрудников HFLabs.
4. Дата-стюарды прошли обучение по работе с системой у сотрудников HFLabs.
5. Выделен руководитель группы дата-стюардов, который будет основным контактным лицом для HFLabs по работе с данными.
6. Для администратора и руководителя группы дата-стюардов созданы учетные записи в системе сопровождения HFLabs (Jira).

## Разбор конфликтов

**i** **Ответственность: высокая.** Неверные действия приведут к тому, что другой человек может увидеть продукты и персональные данные клиента, а также к созданию кентавров в базе «Единого клиента». «Кентавры» — объединенные карточки с перемешанными данными разных клиентов.

**Сложность: высокая.** Необходимо расследовать, почему в исходной системе или в процессе интеграции были сильно изменены данные клиента и принимать решение, что делать с проблемным обновлением.

Бывают ситуации, когда на исходную карточку, уже сохраненную в «Едином клиенте», приходит обновление, содержащее абсолютно иную информацию, нежели хранилась в карточке до этого. Например, раньше в карточке было указано ФИО — Смирнова Наталия Петровна и ДР — 12.07.1986, а в обновлении ФИО — Керчинский Игорь Дмитриевич с ДР — 04.12.1993. В таких случаях «Единый клиент» блокирует обновление, т.к. оно похоже на фрод или ошибку интеграции и может содержать вперемешку данные разных клиентов. Такая ситуация называется конфликтом обновления.

Частота возникновения зависит от качества процессов работы с данными внутри компании. В идеальном мире должно быть ровно 0 таких ситуаций. В реальности — от 5 до 100 ситуаций в месяц.

В «Едином клиенте» есть 2 стратегии разбора конфликтов:

**Автоматическая** — «Разлить и обновить» — карточка, на которую пришло конфликтное обновление, автоматически отсоединяется от объединенной записи, после чего автоматически обновляется на новые данные, даже если они принадлежат другому человеку.

Данная стратегия используется в компаниях телекоммуникационной индустрии, где передача сим-карты от одного человека другому — стандартный процесс и нередки случаи, когда вместо

заведения новой карточки клиента просто перебиваются данные на старой клиентской карточке, которая привязана к передаваемой sim-карте.

При этой стратегии конфликты не отображаются в интерфейсе, и дата-стюарды не участвуют в разрешении конфликтов.

**Ручная** — при возникновении конфликта «Единый клиент» запоминает конфликт, блокирует любые изменения как исходной карточки, на которую поступил конфликт, так и объединенной, куда влита карточка с конфликтом обновления. В интерфейсе «Менеджера данных» обновляется счетчик текущих конфликтов для привлечения внимания дата-стюарда.

Данная стратегия часто используется в банках, где резкое изменение данных клиента с одного человека на другого недопустимо и обычно сигнализирует о мошенничестве или об ошибке на стороне исходной системы или интеграции.

В случае включения данной стратегии необходимо учитывать, что конфликты должны быть разобраны оперативно (желательно в тот же день, когда они возникли), так как пока не разрешён конфликт, объединенная карточка заблокирована от всех других обновлений (даже из других исходных систем) и слияний с другими карточками, даже если они являются гарантированными дублями к этой карточке.

В страховых компаниях в зависимости от процессов внутри компании может использоваться любая из этих двух стратегий.

## Разбор серой зоны по дубликатам

**i** Ответственность: **высокая**. Неверные действия приведут к тому, что другой человек может увидеть продукты и персональные данные клиента, а также к созданию кентавров в базе «Единого клиента». «Кентавры» — объединенные карточки с перемешанными данными разных клиентов.

Сложность: средняя, высокая. Необходимо получать недостающую информацию из исходных систем, сканов документов, архива. Уметь сопоставлять информацию из разных источников.

### Стратегии работы с серой зоной

Обычно сразу после внедрения CDI появляется большая серая зона потенциальных дублей. Это записи, которые похожи друг на друга, но недостаточно сильно, чтобы система автоматически признала их дублями. Объем таких записей колеблется от 500 тысяч до десятков миллионов. Ручной разбор абсолютно всех записей из этой зоны обычно невозможен из-за недостатка в данных и нецелесообразен, так как далеко не все записи относятся к активной базе клиентов. Выделяют несколько стратегий работы с серой зоной:

1. Оставить как есть. Ничего не делать. В этом случае серая зона постоянно увеличивается в процессе жизни системы. Анализ записей серой зоны происходит только в случае обращения клиентов по инцидентам.
2. Разбирать новое, старое оставить как есть. В этом случае потенциальные дубликаты, образовавшиеся при заливке инициализирующего объема записей в «Единый клиент»,

остаются без изменений. Дата-стюарды подключаются к разбору потенциальных дубликатов, которые образуются в результате обновления записей в процессе жизни системы или появления новых записей из исходных систем. Такой подход позволяет зафиксировать размер серой зоны (не происходит увеличения) и разрешать проблемы потенциальных дублей по активным клиентам по мере поступления информации от них. Данная стратегия является наиболее популярной у Заказчиков.

3. Разбирать новое и наиболее важный кусочек инициализирующей загрузки. Аналогично п. 2, но среди первичного объема серой зоны выделяют наиболее важных клиентов (по статусу, активности или каким-то другим критериям) и разбирают вручную те группы, в которые попали эти клиенты вдобавок к разбору вновь появляющихся потенциальных дубликатов.
4. Разобрать вообще всю серую зону.

## **Типовой сценарий разбора серой зоны**

1. Дата-стюард отбирает через фильтр те группы, с которыми ему необходимо работать, например, «Потенциальные дубликаты, созданные вчера».
2. Открывает первую группу и анализирует данные на карточках.
3. В случае, если информации достаточно для принятия решения «Дубликат» или «Не дубликат», принимает соответствующее решение.
4. Если информации недостаточно, то:
  - a. Проверяет наличие сканов документов клиентов, приложенных к карточкам, и информацию в них.
  - b. Ищет информацию в карточках клиентов исходных систем.
  - c. Ищет информацию в карточках договоров исходных систем.
  - d. Запрашивает информацию из архива.
  - e. Отдает карточку для разбора старшему дата-стюарду, если самостоятельно не удалось решить проблему.
5. В случае, если для получения дополнительной информации необходимо время, дата-стюард принимает решение «Отложить» с указанием причины: запрошена информация в архиве, передано старшему дата-стюарду, недостаточно информации для принятия решения и т.п.
6. После получения новой информации дата-стюард возвращается к отложенной карточке и принимает решение, являются ли карточки дубликатами, при необходимости добавляет недостающие данные в карточку.
7. Если принято решение «Дубликат», дата-стюард проверяет, какая информация попадет в золотую карточку на экране «Прогноз слияния». При необходимости корректирует автоматически выбранные системой данные для золотой карточки.
8. Дата-стюард подтверждает объединение карточек.
9. Дата-стюард переходит к разбору следующей группы потенциальных дубликатов.

## Выверка данных перед их использованием

**i** **Ответственность:** низкая

**Сложность:** низкая, средняя

Часто перед проведением рекламных кампаний, маркетинговых рассылок или сдачей отчётности необходимо полностью выверить данные небольшой части клиентов, которые попадут в эту кампанию, рассылку или отчёт. Для этого в системе «Единый клиент» настроена группа фильтров «На ручной разбор». В эти фильтры попадают клиенты, в данных которых система не уверена.

Типовой сценарий проверки данных дата-стюардом:

1. Дата-стюард выставляет фильтр «На ручной разбор» для того типа атрибутов, которые надо выверить.
2. Дата-стюард дополнительно отфильтровывает по идентификаторам те записи клиентов, которые попадали в целевую группу (подготовленную рассылку, отчёт и т.п.).
3. Дата-стюард открывает карточку клиента.
4. Дата-стюард переходит на атрибут, который надо проверить.
5. Дата-стюард раскрывает подробности атрибута и проверяет информацию:
  - a. Если дата-стюард согласен с предположениями, которые сделала система при разборе данных, то он подтверждает изменения, произведенные системой, нажимая кнопку «Подтвердить».
  - b. Если дата-стюард видит, что система распознала данные неверно, то:
    - i. Он вносит необходимые изменения в данные: заполняет поля, изменяет данные в полях, удаляет лишнюю информацию.
    - ii. Нажимает кнопку «Стандартизировать».
    - iii. Система показывает дата-стюарду результат стандартизации измененных данных.
    - iv. Если дата-стюард согласен с вариантом разбора новых данных системой, он нажимает кнопку «Подтвердить».
    - v. Если система неверно распознала новые данные, то:
      1. Дата-стюард вносит необходимые исправления.
      2. Подтверждает данные кнопкой «Подтвердить» без повторной стандартизации.

## Разбор инцидентов качества данных

**i** **Ответственность:** средняя

**Сложность:** средняя

Дата-стюарды работают по инцидентам:

1. Клиент не видит продукт в личном кабинете.
2. Клиент видит незнакомый продукт в личном кабинете.
3. Клиент видит ошибку в данных.
4. Неправильный скоринг.

### **Сценарий 1. Клиент не видит продукт в личном кабинете**

1. Клиент обращается в поддержку ЛК.
2. Поддержка адресует вопрос дата-стюардам.
3. Дата-стюард проверяет данные клиента:
  - a. Если обе записи есть и являются потенциальными дубликатами — объединяет их.
  - b. Если записи не являются потенциальными дубликатами — выясняет причину ошибки:
    - i. Если указаны разные клиентские данные в карточках, но на самом деле они относятся к одному клиенту, что явствует из запроса клиента, то исправляет данные в карточке таким образом, чтобы их хватило для автоматического объединения этих записей, когда отработает регулярная задача поиска и объединения дубликатов.
    - ii. Если клиенты уже содержат похожую информацию и должны были найтись как дубли, но не нашлись, передаёт инцидент руководителю группы для отправки запроса и данных в HFLabs.
  - c. Одно или обоих записей нет в ЕК передаёт запрос команде интеграции для проверки, были ли переданы такие данные в ЕК.

### **Сценарий 2. Клиент видит чужой продукт в личном кабинете**

1. Клиент обращается в поддержку ЛК.
2. Поддержка адресует вопрос дата-стюардам.
3. Дата-стюард проверяет данные клиента:
  - a. Если в золотой записи клиента ошибочно присутствуют карточки других клиентов, то вручную отсоединяет эти карточки.

- b. Если невозможно идентифицировать чужие карточки по клиентским данным, то анализирует данные в исходной системе: возможно, что неверно отработала интеграция, и договор привязали к чужой карточке клиента. Для решения проблемы с отвязкой договора в ЕК передает запрос руководителю группы для создания инцидента в системе сопровождения HFLabs
- c. Если карточка не найдена в ЕК, передает запрос команде интеграции.

### **Сценарий 3. Ошибка в персональных/контактных данных клиента**

1. Клиент обращается в поддержку ЛК.
2. Поддержка адресует вопрос дата-стюардам.
3. Дата-стюард исправляет данные в ЕК и уточняет источники, из которых пришли ошибочные данные.
4. Дата-стюард отправляет данные по ошибке в источник для исправления в первичной записи.

### **Сценарий 4. Неправильный скоринг**

1. Клиент обращается в поддержку компании с жалобой о неправильном скоринге.
2. Если причина в том, что не все данные клиента были учтены, то поддержка адресует вопрос дата-стюардам.
3. Дата-стюард проверяет данные в карточке клиента ЕК и при необходимости:
  - a. Анализирует серую зону: возможно что старые карточки клиента не были учтены, т.к. попали в негарантированные дубли. При необходимости подтверждает дубликаты и объединяет карточки.
  - b. Исправляет данные в ЕК и уточняет источники, из которых пришли ошибочные данные.  
Дата-стюард отправляет данные по ошибке в источник для исправления в первичной записи.

# ТИПОВЫЕ ВОПРОСЫ ПРИ РАБОТЕ ДАТА СТЮАРДОВ

- Доступ в систему
  - Как попасть в систему?
  - Что нужно для доступа нового человека к системе?
  - Не могу залогиниться из-за ошибки «Неверный пароль»
  - Система недоступна
- Разбор серой зоны
  - Как понять, какие данные выбрать победителем при признании карточек дубликатами?
- Поиск клиента
  - Почему я не вижу клиента в ЕК? В исходной системе он есть, а в ЕК нет?
  - Почему не объединились клиенты? Визуально видно, что это одинаковые записи, но они остались в системе как разные карточки.
- Анализ данных на карточке клиента
  - Почему в Золотой записи выиграли те или иные данные. Что делать, если они не верные?
  - В карточке клиента я вижу данные другого человека, что делать?
  - Почему я вижу, что в исходной системе у клиента есть адрес/телефон/документ, а в ЕК этих данных нет?
  - Почему не проходят обновления на карточку? В исходной системе изменили данные, а в ЕК их не видно, почему?
  - Не могу отредактировать карточку клиента или внести новые данные клиента
  - Система неправильно распознает данные после нажатия на кнопку «Стандартизировать»
- Система выдала ошибку
  - Появилась ошибка «Что-то пошло не так»
  - При работе с дубликатами появилась ошибка «Неактуальная группа дубликатов»

HFLabs как вендор оказывает третью линию поддержки по вопросам работы системы и качества обработанных данных.

Все запросы в поддержку HFLabs от дата-стюардов передаются через вторую линию, роль которой выполняют администраторы приложения заказчика или руководитель группы дата-стюардов, если вопросы касаются качества данных.

Администратор перед отправкой запроса в HFLabs собирает диагностическую информацию по нужным клиентам.

## ***Доступ в систему***

### ***Как попасть в систему?***

1. Для того чтобы попасть в систему «Единый клиент» необходимо перейти по ССЫЛКЕ.
2. Для входа в систему необходимо ввести логин и пароль от доменной учетной записи.

### ***Что нужно для доступа нового человека к системе?***

Для предоставления доступа к системе «Единый клиент» необходима заявка в службу технической поддержки на включение сотрудника в доменную группу с указанием ролей, которые необходимы сотруднику:

1. Только просмотр карточек.
2. Просмотр и редактирование карточек.
3. Работа с потенциальными дубликатами.

### ***Не могу залогиниться из-за ошибки «Неверный пароль»***

1. Для входа в систему необходимо ввести логин и пароль от доменной учетной записи.
2. Если дата-стюард видит ошибку «Неверный пароль», необходимо убедиться, что учетная запись пользователя активна и включена в необходимую доменную группу. Обратитесь к администратору системы.

### ***Система недоступна***

«Единый клиент» — отказоустойчивая и стабильная система, однако бывают технические моменты связанные с сетевой доступностью, работоспособностью инфраструктуры и проведением технических работ.

Решение ? Перезагрузить станицу в браузере без кеша через Ctrl+F5 или Ctrl+R. Если ошибка не исчезла, обратиться к администратору системы.

## ***Разбор серой зоны***

### ***Как понять, какие данные выбрать победителем при признании карточек дубликатами?***

Нужно составить «шпаргалки», в которых:

- Описать логику обработки слияния дубликатов по кейсам. Например, если в дубликатах 2 клиента со связанными автомобилями и водительским удостоверением и у одного дата рождения 1985, а у второго 2005, то при прочих равных выбираем победителем дату рождения 1985 год. Т.е. предполагаем, что в 14 лет у человека не может быть ВУ и автомобиля.



- Составить для дата-стюардов матрицу приоритетных систем (если в группе дубликатов есть карточка из системы А и из системы Б, то данным из А верить больше, чем из Б).

## **Поиск клиента**

### **Почему я не вижу клиента в ЕК? В исходной системе он есть, а в ЕК нет?**

1. Попробовать найти клиента по другим данным:
  - a. по идентификатору в исходной системе с указанием кода исходной системы, например, SYS:124578
  - b. по фамилии, имени и дате рождения в формате **Фамилия Имя ДД.ММ.ГГГГ** без других разделителей. (**Фамилия, имя, дд-мм-гггг** — неправильно!)
2. Убедиться, что в исходной системе клиент не отмечен ни одним из vip-признаков. Для работы с vip-клиентами в системе может быть предусмотрена отдельная роль, остальным сотрудникам данные по vip-клиентам в этом случае будут не видны, и через поиск найти такого клиента не удастся.
3. Если клиента не удалось найти, то уточнить дату создания клиента в исходной системе.
  - a. Если он создан «сейчас», возможно создание еще не дошло до ЕК.
  - b. Если клиентская запись в исходной системе старше 24 часов при пакетной интеграции или старше 5 минут при онлайн-интеграции, то обратиться в поддержку HF Labs с указанием:
    - i. Полной информации по клиенту и его идентификатора в исходной системе.
    - ii. Описанием, каким способом пробовали найти клиента в «Едином клиенте».

### **Почему не объединились клиенты? Визуально видно, что это одинаковые записи, но они остались в системе как разные карточки.**

1. Если записи найдены как потенциальные дубликаты, значит, системе не хватило данных для их автоматического объединения. В этом случае необходимо объединить записи вручную, подтвердив, что они являются дублями на экране групп дубликатов.
2. Если записи похожие или одинаковые, но при этом система не считает их дубликатами:
  - a. Проверить, что записи не были ранее отмечены дата-стюардом как «Не дубликаты».
  - b. Если запись создана или обновлена менее 24 часов назад, то дождаться регулярной (обычно ночной) задачи по поиску и объединению дубликатов.
  - c. Если запись создана или обновлена последний раз более 24 часов назад, то собрать всю диагностическую информацию по обеим записям и обратиться в поддержку HF Labs.

## ***Анализ данных на карточке клиента***

### ***Почему в Золотой записи выиграли те или иные данные. Что делать, если они не верные?***

ЕК выбирает победителем более актуальные или более достоверные с точки зрения заложенных в него алгоритмов данные. Например, более свежий по дате выдачи паспорт вероятнее всего победит более старый.

В реальности бывают разные ошибки и опечатки в данных, и паспорт, выданный раньше, может оказаться правильным. В этом случае дата-стюард исправляет данные в золотой записи, если он уверен в своей правоте.

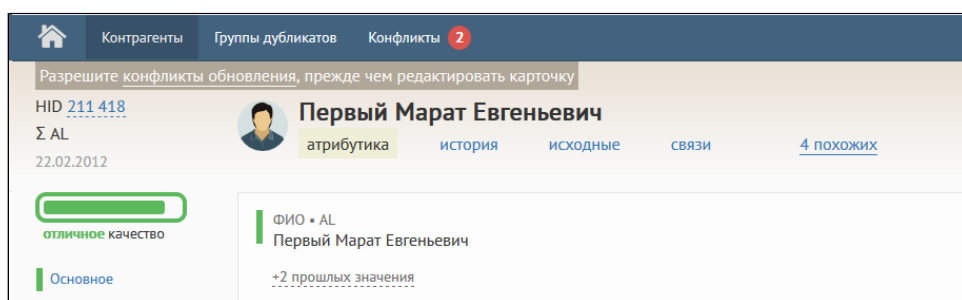
### ***В карточке клиента я вижу данные другого человека, что делать?***

1. Выяснить, какая исходная карточка является источником чужих данных.
2. Определить, является ли исходная карточка с чужими данными карточкой другого клиента:
  - a. Если это карточка другого клиента, то:
    - i. Отсоединить данную карточку из объединенной записи. Информация в объединенной записи обновится автоматически.
    - ii. Проверить, кто подтвердил, что карточки являются дубликатами:
      1. Если дубликаты подтверждены автоматически (автор подтверждения — HFLabs), то отправить запрос с диагностическими данными в поддержку HFLabs.
      2. Если дубликаты были подтверждены вручную, то отправить инцидент на руководителя дата-стюардов.
  - b. Если чужие данные попали в исходную карточку по ошибке, а сама карточка принадлежит рассматриваемому клиенту, то:
    - i. Удалить чужие данные на объединенной карточке.
    - ii. Отправить запрос на исправление данных в исходную систему.

### ***Почему я вижу, что в исходной системе у клиента есть адрес/телефон /документ, а в ЕК этих данных нет?***

Если атрибут в исходной системе осмысленно заполнен, он должен быть в ЕК.

1. Проверить, что значение не является мусорным или тестовым с точки зрения «Единого клиента» через документацию по правилам стандартизации ненайденного атрибута. Как правило, тестовые и придуманные значения удаляются системой на этапе очистки данных.
2. Проверить, что на обновляемой клиентской карточке нет **конфликтов**. Конфликты блокируют карточку от любых последующих обновлений.  
Пример карточки с конфликтом:



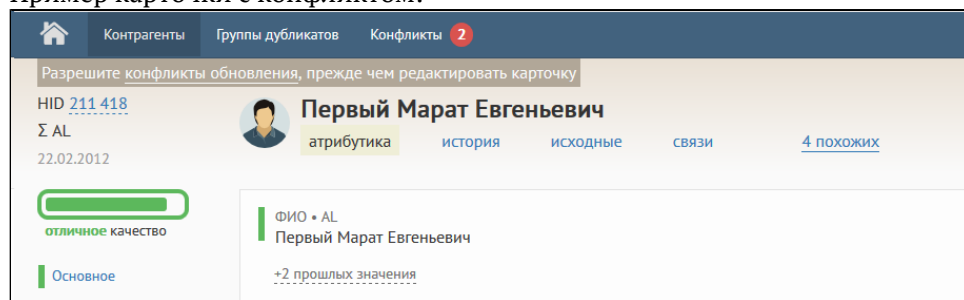
Если обнаружен конфликт, то:

- a. Разрешить его в соответствии с разделом «Работа с конфликтами».
  - b. Проверить, появились ли необходимые данные на карточке после разрешения конфликта.
3. Проверить, что данные не были ранее удалены вручную из системы:
- a. Зайти на вкладку «история».
  - b. Попробовать найти искомые данные.
  - c. Если данные были удалены вручную, то:
    - i. Выяснить причину удаления.
    - ii. Если данные удалены верно в «Едином клиенте», составить запрос в исходную систему на удаление данных.
    - iii. Если данные удалены ошибочно, то собрать диагностическую информацию и обратиться в поддержку HF Labs.
4. Если предыдущие шаги не помогли найти причину отсутствия данных, то собрать диагностическую информацию и обратиться в поддержку HF Labs.

### ***Почему не проходят обновления на карточку? В исходной системе изменили данные, а в ЕК их не видно, почему?***

1. Проверить, что новое значение:
  - a. Не является мусорным или тестовым с точки зрения «Единого клиента» через документацию по правилам стандартизации ненайденного атрибута. Как правило, тестовые и придуманные значения удаляются системой на этапе очистки данных.
  - b. Не является пустым. Для большинства полей в «Едином клиенте» настроены правила, которые препятствуют обновлению на пустое значение (обычно это означает ошибку в интеграции).
    - i. Проверить, что правила обновления для этого поля допускают обновление на пустое значение.
    - ii. Если правила не допускают обновления на пустое значение, но при этом действительно есть ситуации, когда значение может быть удалено из поля в источнике, обратиться в поддержку HF Labs с описанием, почему поле должно допускать обновление на пустые значения.
2. Проверить, что на обновляемой клиентской карточке нет **конфликтов**. Конфликты блокируют карточку от любых последующих обновлений.

Пример карточки с конфликтом:



Если обнаружен конфликт, то:

- a. Разрешить его в соответствии с разделом «Работа с конфликтами».
  - b. Проверить, появились ли необходимые данные на карточке после разрешения конфликта.
3. Если предыдущие шаги не помогли найти причину отсутствия данных, то собрать диагностическую информацию и обратиться в поддержку HFLabs.

### ***Не могу отредактировать карточку клиента или внести новые данные клиента***

Для поиска причин этой ситуации необходимо:

1. Проверить, что открыта именно золотая запись клиента, а не исходная карточка. В случае, если открыта исходная карточка, система напишет в верхнем левом углу предупреждение «Неактуальная карточка: влита в HID XXXX». Решение ? Перейти на актуальную карточку по предложенной системой ссылке.
2. Убедиться, что другой сотрудник не работает сейчас с этой карточкой. В этом случае в левом верхнем углу экрана будет сообщение «Карточка редактируется пользователем <username>». Одновременное редактирование карточек запрещено. Решение ? Попросить другого сотрудника закрыть карточку или перейти на работу со следующей карточкой.
3. Убедиться, что карточка не закрыта ранее в системе. В случае просмотра закрытой карточки информация об этом будет в левом верхнем углу. Редактирование закрытых карточек запрещено. Решение ? Выяснить, зачем нужно редактировать закрытую карточку. Если карточка закрыта по ошибке, то собрать диагностическую информацию и обратиться в поддержку HFLabs.
4. Убедиться в отсутствии конфликта на карточке. При его наличии в левом верхнем углу будет предупреждение «Разрешите конфликты обновления, прежде чем редактировать карточку». Решение ? Перейти на конфликт по ссылке из сообщения и разрешить конфликты в соответствии с разделом «Работа с конфликтами».
5. Проверить, есть ли возможность редактировать другие карточки:
  - a. Если нет возможности отредактировать ни одну карточку, то обратиться к администратору системы с просьбой предоставить права на редактирование карточек. В системе настроены разные роли:
    - i. Роль оператора — этой роли доступен только просмотр карточек.

- ii. Роль менеджера данных — для редактирования карточек.
- b. Если проблема с редактированием только одной или нескольких карточек, то собрать диагностическую информацию по проблемным карточкам и обратиться в поддержку HFLabs.

### ***Система неправильно распознает данные после нажатия на кнопку «Стандартизировать»***

В этом случае необходимо внести данные в гранулярные поля так, как дата-стюард считает правильным (например, на основе скана документа) и нажать кнопку «Подтвердить», не проводя стандартизацию данных.

Система сохранит введённые данные без выполнения алгоритма очистки и стандартизации ровно в том виде, как их внёс дата-стюард.

### ***Система выдала ошибку***

#### ***Появилась ошибка «Что-то пошло не так»***

В системе возникла непредвиденная ситуация, но не стоит беспокоиться. В 99,9% случаев перезагрузка страницы в браузере помогает ее исправить. Если ошибка не исчезла после перезагрузки или появляется несколько раз в течение дня, необходимо обратиться к администратору системы.

#### ***При работе с дубликатами появилась ошибка «Неактуальная группа дубликатов»***

Чаще всего эта ошибка говорит о том, что клиентские запись уже объединены либо системой автоматически, либо другим дата-стюардом. Следует нажать кнопку «Следующая группа» и перейти к разбору следующей необработанной группы.

# ТРЕБОВАНИЯ К МЕНЕДЖЕРУ ДАННЫХ

## *Требования*

- Общая компьютерная грамотность (владение мышкой и клавиатурой, навыки работы на компьютере на уровне уверенного пользователя).
- Грамотность (работа с русскоязычными контактными данными).
- Усидчивость, внимательность (выверка данных — процесс монотонный, внимание не должно рассеиваться со временем).
- Умение формулировать вопросы (на случай возникновения затруднений).

Plusom будет:

- Опыт работы с контактными данными, наличие представления о предметной области.
- Умение анализировать данные, выделять группы данных с однотипными проблемами.

## *Обязанности*

- Проверка и изменение информации о контрагентах в АРМ Менеджера данных;
- Объединение и разъединение дубликатов в АРМ Менеджера данных.

# ОБЩИЕ СЦЕНАРИИ

# НАЧАЛО РАБОТЫ С СИСТЕМОЙ

1. Откройте браузер и перейдите по адресу, где работает система. Узнать адрес вы можете у системного администратора (адрес имеет вид **http://xxx.xxx.xxx.xxx:8080/cdi**)
2. Дождитесь появления окна с авторизацией. В поле **Логин** укажите имя пользователя, под которым вы вошли в операционную систему, а в поле **Пароль** – ваш пароль.

Пожалуйста, укажите логин и пароль для входа в интерфейс управления.

Программный комплекс «Физлиц» является производным для централизованного хранения, обеспечения качества и синхронизации информации с клиентами между различными информационными системами.

Логин:

Пароль:

Запомнить меня

3. Нажмите на кнопку **Войти**. Откроется основной экран приложения.

ID	Автор	Дата акт.	Фамилия	Имя	Отчество	Пол	Дата рождения
774 105	AL:094156, AL:325215356	06.05.2017	Олегова	Иванова	Валентиновна	Женский	
774 104	BT:99921321511, BT:99921523497		Колесникова	Ольга	Владимировна	Женский	08.08.1987
774 103	AL:92044, AL:932044	06.05.2017	Жиглов	Ильдэр	Олегович	Мужской	01.01.1960
774 102	AL:10393337, AL:1093337	06.05.2017	Петров	Евгений	Андреевич	Мужской	03.08.1993
774 101	AL:10937, AL:39055448	15.07.2019	Викторов	Виктор	Викторович	Мужской	03.08.1995
774 100	AL:80334333294, AL:803433394	06.05.2017	Документов	Документ	Документович	Мужской	01.01.1960
774 099	AL:210999, AL:211002	09.11.2016	Уильям	Ростям	Николасевич	Мужской	
774 098	BT:99921321501, BT:99921321502, BT:99921321503, BT...		Иванова	Елена	Анатольевна	Женский	08.08.1987
772 539	AL:8030294	06.05.2017	Иванов	Александр	Бондариновна	Неизвестно	01.01.1960



# РАБОТА СО СПИСКОМ КОНТРАГЕНТОВ

- Просмотр списка

- Работать со списком клиентов

- Поиск клиента

- Найти конкретного человека

- Поиск клиента по исходному идентификатору

- Поиск клиентов по списку НІD

- Поиск неактуального клиента по НІD

- Поиск по прошлому ФИО или документу

- Предопределенные фильтры

- Синтаксис для поиска и выборки

- Расширенный фильтр

- Фильтр по коду качества документа

- Фильтр по «Популярности» атрибутов

- Фильтр по исходной системе

- Фильтр по вхождению в реестр

- Фильтр по проблемам на карточке

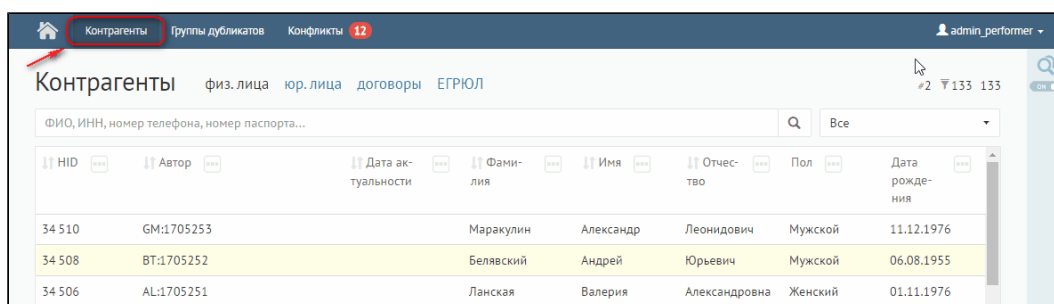
- Фильтр для юрлиц

- Переход на карточку контрагента

## ПРОСМОТР СПИСКА

Список контрагентов содержит основную идентифицирующую информацию по контрагентам, а также набор инструментов, позволяющий найти необходимого контрагента.

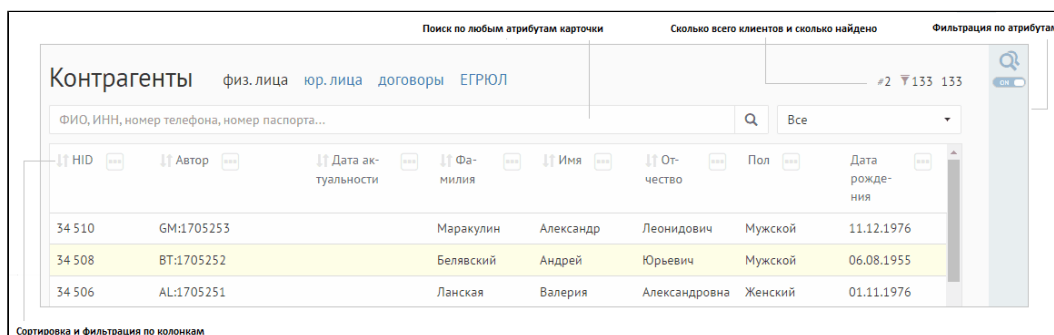
Чтобы открыть список, нужно перейти на вкладку «Контрагенты»



HID	Автор	Дата актуальности	Фамилия	Имя	Отчество	Пол	Дата рождения
34 510	GM:1705253		Маракулин	Александр	Леонидович	Мужской	11.12.1976
34 508	BT:1705252		Белявский	Андрей	Юрьевич	Мужской	06.08.1955
34 506	AL:1705251		Ланская	Валерия	Александровна	Женский	01.11.1976

### *Работать со списком клиентов*

В Менеджере данных дата-стюард находит интересующих его клиентов и исправляет контактные данные, отмеченные на ручную проверку.

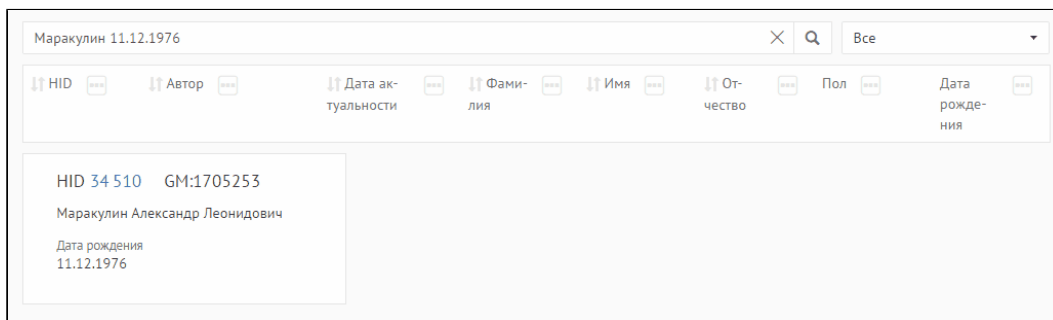


HID	Автор	Дата актуальности	Фамилия	Имя	Отчество	Пол	Дата рождения
34 510	GM:1705253		Маракулин	Александр	Леонидович	Мужской	11.12.1976
34 508	BT:1705252		Белявский	Андрей	Юрьевич	Мужской	06.08.1955
34 506	AL:1705251		Ланская	Валерия	Александровна	Женский	01.11.1976

## Поиск клиента

### *Найти конкретного человека*

Единый клиент найдет человека по любым признакам — ФИО, дата рождения, ИНН, адрес, телефон, паспорт. Укажите их в свободном формате и нажмите Энтер.



Особенности поиска по атрибутам:

### 1. По телефону:

- номер телефона следует указать в формате "{код страны} {код} {номер телефона}", например, "7 926 1234567";
- можно указать телефон без разделителей, в формате "{код страны}{код}{номер телефона}", например, "79261234567";
- в строке поиска можно ввести номер телефона без кодов, например, "1234567";
- допустим вариант с указанием кода города и номера телефона без разделителя, например, "9261234567";

Поиск введенной в поле номера строки будет производиться как во всему телефону целиком, так и по каждому компоненту отдельно (код страны, код города, номер телефона);

**2. По документу** — серию и номер в строке поиска следует указывать так, как они сохранены в карточке контрагента. Пример строки поиска для прошедшего стандартизацию паспорта РФ: "45 06 123456".

**3. По адресу** — без точек и запятых, через пробел. Например, клиента с адресом "ул. Ленина, д. 1, к. 1, кв 55" можно найти двумя способами:

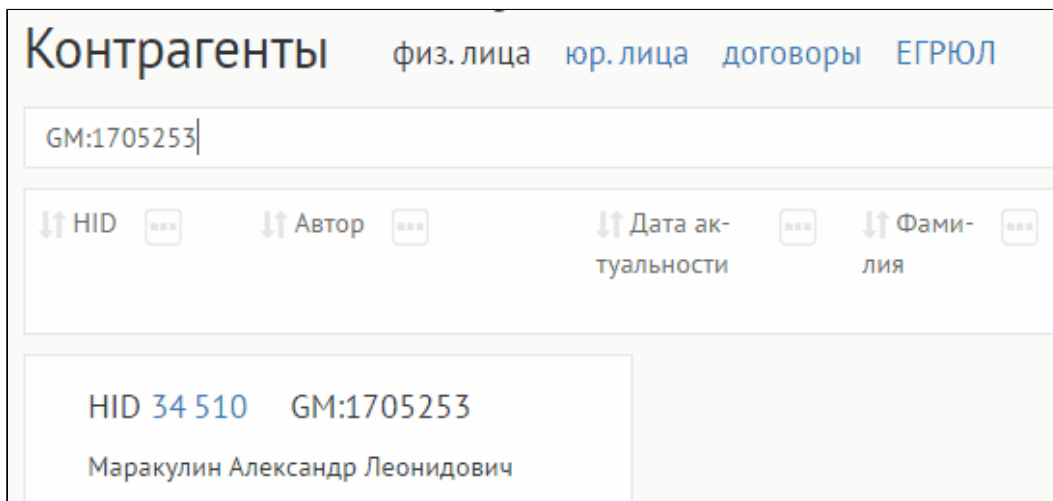
- в строке поиска указать только название улицы: "Ленина";
- по полному адресу, разделенному пробелами: "Ленина 1 1 55".

## **Примеры поиска по телефону 7 921 5555555**

1. 7 921 5555555
2. 79215555555
3. %79215555555 — эквивалентно 79215555555 и +79215555555
4. %92155555555 — эквивалентно 79215555555, +79215555555, 89215555555, 92155555555
5. 92155555555 — допустимо искать по коду города и номеру телефону без разделителя

## **Поиск клиента по исходному идентификатору**

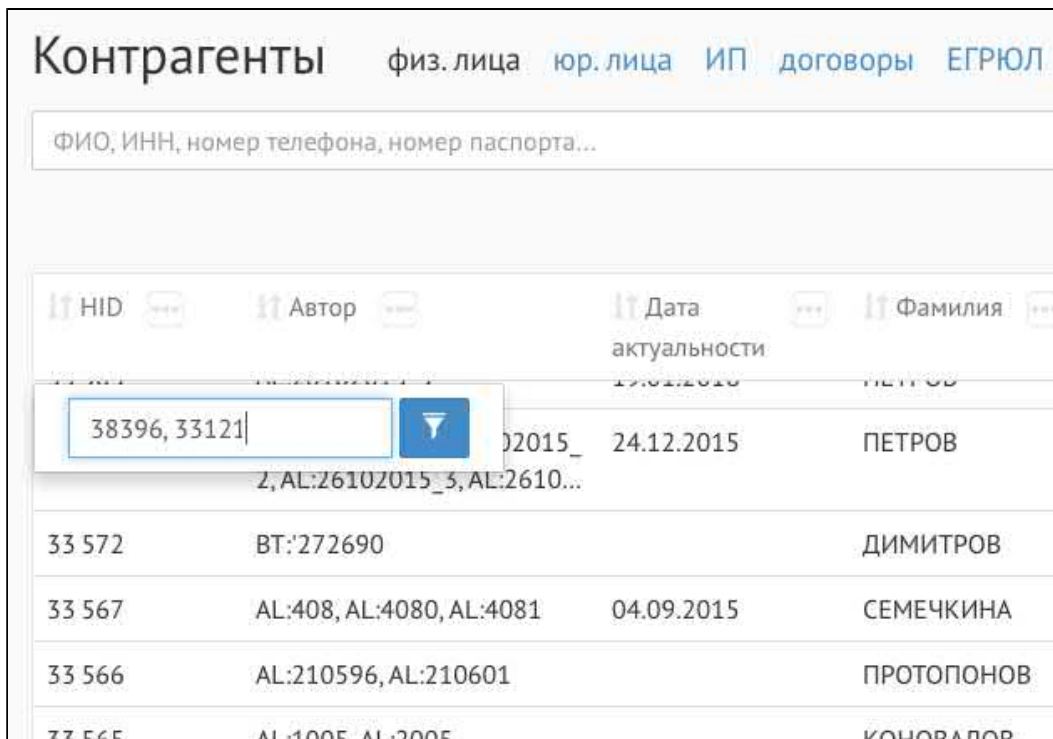
Иногда все, что известно дата-стюарду о клиенте — это исходный идентификатор: уникальный номер, под которым клиент записан в исходной системе. Чтобы найти такого клиента, укажите исходный идентификатор в поле поиска на списке контрагентов:



Обязательно укажите код исходной системы и двоеточие (GM:), Единый клиент умеет искать только с ним.

### Поиск клиентов по списку НID

Чтобы найти клиентов по списку НIDов, скопируйте их (через запятую, пробел или в столбик — неважно) и вставьте в фильтр на столбце НID. Единый клиент вернет всех подходящих клиентов:



???

Контрагенты [физ. лица](#) [юр. лица](#) [ИП](#) [договоры](#) [ЕГРЮЛ](#)

ФИО, ИНН, номер телефона, номер паспорта...

HID  Автор  Дата  Фамилия  Имя 
  
 актуальности

38396, 33121 ✕

<p>HID <b>38 396</b> AL:25003</p> <p>ИРИШКИНА ЛЮБОВЬ</p> <p>Дата рождения 18.03.1975</p> <p>Иной 119034, РОССИЯ, Г МОСКВА, Г МОСКВА, ПЕР ТУРЧАНИНОВ, д. 6, стр. 2, кв. 115</p> <p>Домашний 7 495 776-47-00</p>	<p>HID <b>33 121</b> AL:1027,AL:2027</p> <p>КУМЕНКО ИВАН ВИКТОРОВИЧ</p> <p>Дата рождения 17.04.1976</p> <p>Домашний 7 495 776-47-74</p>
--	---

### ***Поиск неактуального клиента по HID***

Бывает, что во внешней системе хранится HID клиента, а сам клиент в ЕК уже закрыт или влился в кого-то. Такого клиента можно найти через поиск на списке:

Контрагенты [физ. лица](#) [юр. лица](#)

42455 ✕ 🔍

???

Неактуальная карточка: влита в НID 49396

НID 42 455  
27.09.2016

**Глобенко Андрей Алекс**  
атрибутика история

хорошее качество

- Основное
- 1 документ
- 1 адрес
- 12 согласий

Доп. атрибуты

ФИО • admin\_performer  
Глобенко Андрей Александрович

Дата рождения • ВТ  
12.12.1987

ИНН • ВТ  
025004781276

СНИЛС • ВТ

Неактуальный клиент найдется, только если:

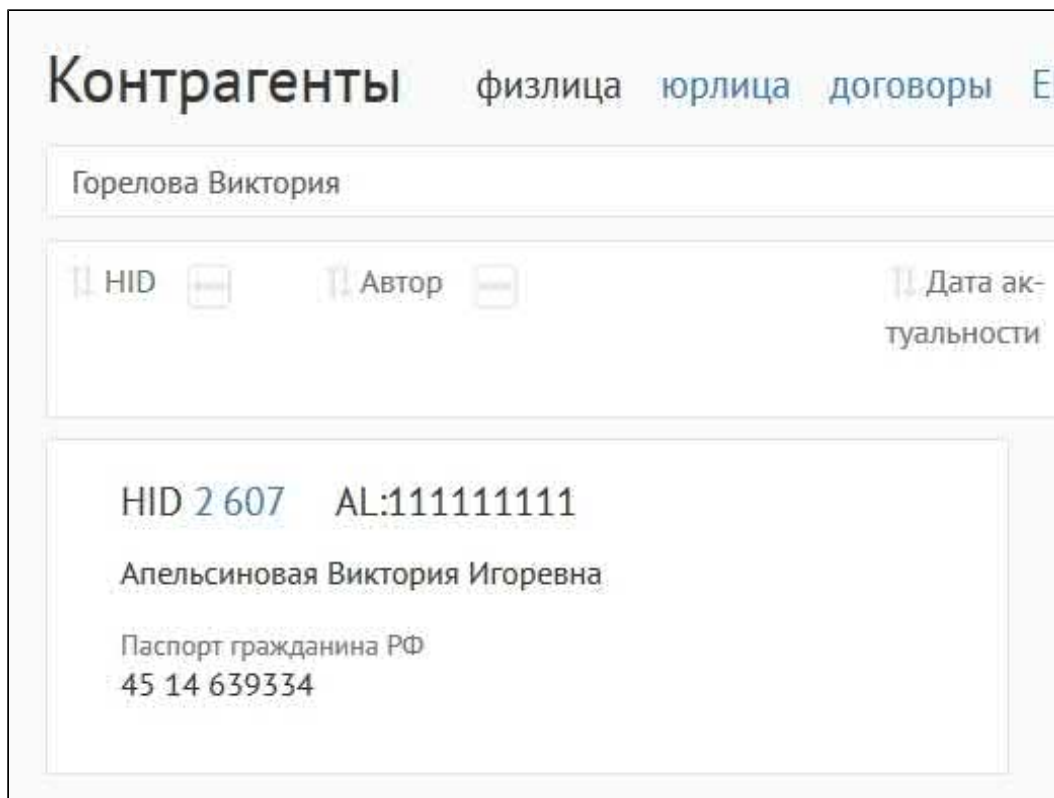
- на списке *не установлены* другие критерии отбора (на столбцах или в расширенном фильтре);
- по поиску не нашлись актуальные клиенты (по документу, телефону или другому полю) — если поиск ничего не нашел и в запросе введены цифры, он поищет среди неактуальных карточек.

### ***Поиск по прошлому ФИО или документу***

Бывает, что у клиента изменилось ФИО или документ, а прошлые значения сохранены в ЕК. «Единый клиент» будет искать человека по прошлым значениям ФИО и документа точно так же, как по актуальным. Прошлые значения можно искать с помощью строки поиска и расширенного поиска. В фильтре ручной обработки они не учитываются.

## Строка поиска

Клиента, прошлое ФИО или документ которого сохранены в ЕК можно с помощью строки поиска. Например, Горелова Виктория сменила фамилию и стала Апельсиновой. Но ЕК помнит ее прошлую фамилию и находит. Простое значение можно увидеть при переходе на карточку Апельсиновой Виктории.



Контрагенты физлица юрлица договоры ЕК

Горелова Виктория

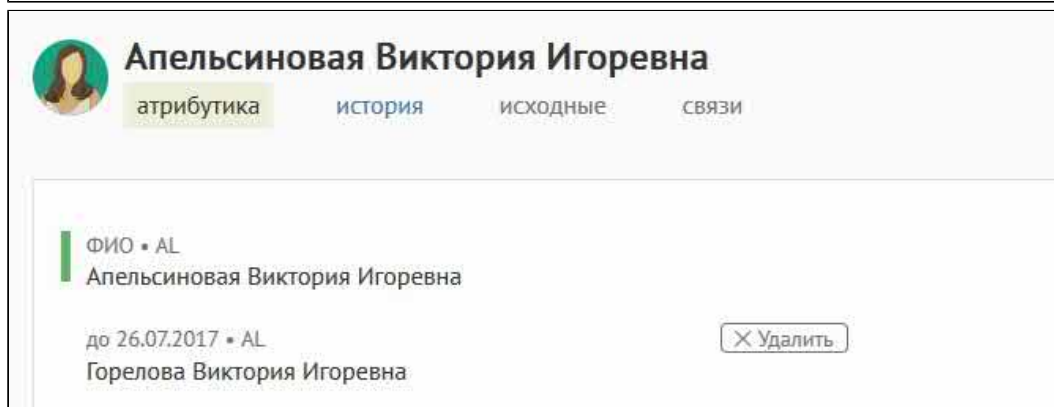
HID Автор Дата актуальности

HID 2 607 AL:111111111

Апельсиновая Виктория Игоревна

Паспорт гражданина РФ  
45 14 639334

? ? ?



Апельсиновая Виктория Игоревна

атрибутика история исходные связи

ФИО • AL  
Апельсиновая Виктория Игоревна

до 26.07.2017 • AL

Горелова Виктория Игоревна

Удалить

Аналогично работаем с прошлыми документами. Например, ищем прошлый паспорт:

# Контрагенты

физлица

юрлица

договоры

ЕГРЮЛ

45 14 639334

|| HID

|| Автор

|| Дата ак-  
туальности


HID 2 607 AL:111111111

Апельсиновая Виктория Игоревна

Паспорт гражданина РФ  
42 17 916222

? ? ?





## Апельсиновая Виктория Игоревна

атрибутика
история
исходные
связи

ФИО • AL  
Апельсиновая Виктория Игоревна

+1 прошлое значение

Предыдущее ФИО • AL

Дата рождения • AL

ИНН • AL

СНИЛС • AL

1 документ +

Паспорт гражданина РФ • AL  
42 17 916222

→

до 26.07.2017 • AL

45 14 639334

X Удалить

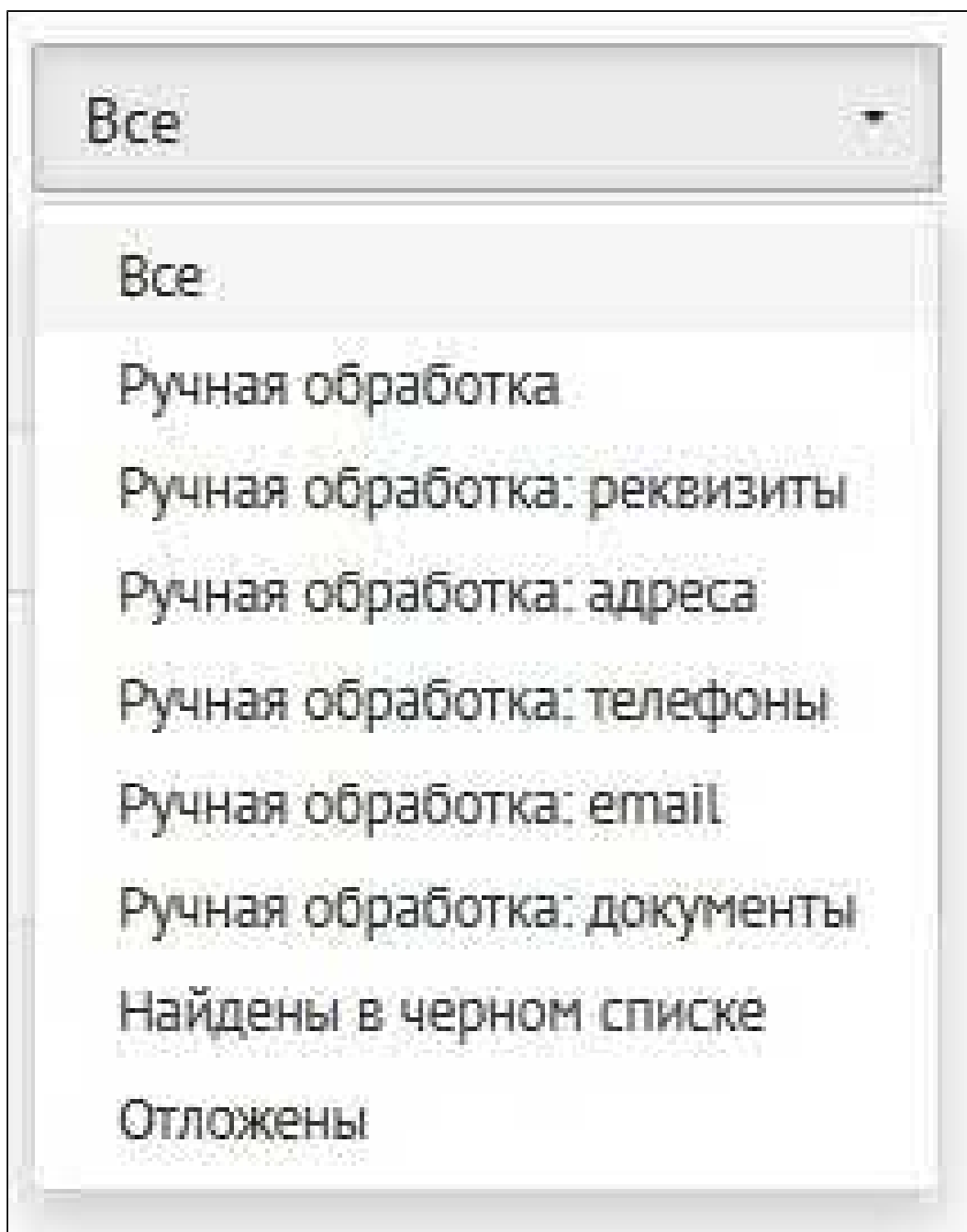
Также в строке поиска работает **расширенный поиск**. Для ФИО прошлое значение возвращается по запросу **? PhysicalParty.fullName = (прошлое ФИО)**, прошлый документ по запросу **? Passport.fullDocument = прошлый документ**.

## Расширенный фильтр

Расширенный фильтр для поиска прошлых документов работает несколько иначе, чем строка поиска:

- при фильтрации по полям «тип», «дата актуальности» и «серия / номер» — учитывает и прошлые, и актуальные значения документов;
- «код качества», «актуальность» и «популярность» — учитывает только актуальные значения документов.

## Предопределенные фильтры



- **Все** — показываются все записи.
- **Ручная обработка** — показываются контрагенты, у которых хотя бы один из реквизитов, адресов, телефонов, email или документов контрагента отмечен на ручную обработку.
- **Ручная обработка: реквизиты** — показываются контрагенты, у которых **ФИО** или **другие реквизиты** контрагента отмечены на ручную обработку.

- **Ручная обработка: адреса** — показываются контрагенты, у которых хотя бы один из адресов отмечен на [ручную обработку](#).
- **Ручная обработка: телефоны** — показываются контрагенты, у которых хотя бы один из телефонов отмечен на [ручную обработку](#).
- **Ручная обработка: email** — показываются контрагенты, у которых хотя бы один из адресов электронной почты отмечен на [ручную обработку](#).
- **Ручная обработка: документы** — показываются контрагенты, у которых хотя бы один из документов отмечен на [ручную обработку](#).
- **Найдены в черном списке** — показываются контрагенты, найденные в [черных списках](#).
- **Отложены** — показываются контрагенты, у которых хотя бы один из реквизитов или атрибутов отложен.

## Синтаксис для поиска и выборки

При использовании *поиска* и *отбора по значению колонки* можно использовать подстановочные символы:

- **\_** — один произвольный символ;
- **%** — произвольная последовательность символов (ноль или более символов).

Краткая памятка по использованию подстановочных символов:

- **%ИВАН%** — будут найдены все значения, содержащие подстроку **ИВАН**. Например: **ИВАНОВ, ИВАННИКОВА, СЕЛИВАНОВ**.
- **ИВАН%** — будут найдены все значения, начинающиеся с **ИВАН**. Например: **ИВАНОВ, ИВАННИКОВА**. Но не **СЕЛИВАНОВ**.
- **%ВАНОВ** — будут найдены все значения, заканчивающиеся на **ВАНОВ**. Например: **ИВАНОВ, СЕЛИВАНОВ**. Но не **ИВАНОВА, ГОЛОВАНОВА**.
- **АЛЕКС%ВИЧ** — будут найдены все значения, которые содержат подстроку **АЛЕКС**, затем произвольная последовательность символов, затем **ВИЧ**. Например: **АЛЕКСЕЕВИЧ, АЛЕКСАНДРОВИЧ**.
- **А\_А** — будут найдены все значения, которые начинаются на А, затем содержат любые 2 буквы, после которых снова идет А. Например: **АЛЛА, АННА**.

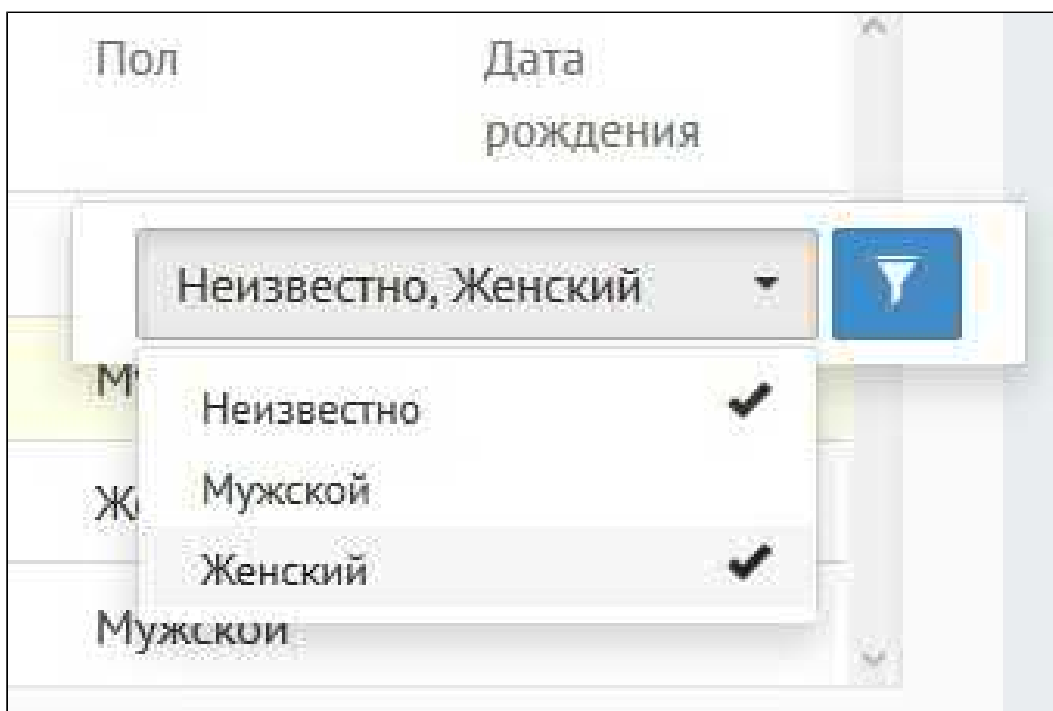
Для полей, содержащих даты, существует возможность задать интервал дат для выборки. Проверка на корректность введенных дат не осуществляется.

ФИО, ИНН, номер телефона, номер паспорта...

По каждой колонке можно отфильтровать данные

НID	Автор	Дата актуальности	Фамилия
3 009	AL:20180820173409	30.07.2018 - 30.10.2018	Шайтан
3 004	AL:20180820173406	За последний час	Шайтан
3 003	AL:20180820173405	Сегодня	Шайтан
3 002	AL:444001	Вчера	Болотова
2 965	AL:20180820173307	В этом месяце	Шайтан
2 949	AL:20181010115901	В этом году	
		Пустые значения	
		Непустые значения	

Для полей, содержащих набор predetermined значений (например - пол), можно установить фильтрацию по нескольким значениям.



## Расширенный фильтр

В расширенном фильтре во всех выпадающих списках с мультिवыбором можно выбрать все значения пунктом «?Выбрать всё?», при условии, что в списке больше 5 пунктов.

### ***Фильтр по коду качества документа***

Единый клиент возвращает причину, по которой документ признан плохим:

- *Плохая серия или номер.* Некорректный формат серии, недействительная серия, некорректный формат номера.
- *Пустая или плохая дата выдачи.* Плохая — значит, сильно в прошлом или из будущего.
- *Плохой код подразделения.* Некорректный формат кода подразделения или несуществующий код подразделения.

Отдельным признаком «Единый клиент» отмечает просроченные и недействительные документы:

- *Недействительный.* Найден в справочнике недействительных документов МВД.
- *Просроченный.* Определяется по сравнению даты выдачи с датой рождения.

На ручную обработку при этом уходят только документы с плохой серией, номером, датой выдачи или кодом подразделения. Недействительные, просроченные и документы с пустой датой выдачи на ручную обработку не попадают — хоть они и плохие, исправлять нечего.

Чтобы выбрать документы с конкретным кодом качества, используйте расширенный фильтр на списке клиентов:

▼ Документ, удостоверяющий личность ON

Тип

(Не задано) ▼

Дата акт.

(Не задано) ▼

Серия/номер

Код качества

- Подтверждено контролером
- Правильное значение
- Сомнительный (устар.)
- Плохой код подразделения
- Плохая дата выдачи
- Пустая дата выдачи
- Плохая серия или номер

Актуальность

- Актуальный
- Недействительный
- Просроченный

## Фильтр по «Популярности» атрибутов

Чтобы выбрать популярные адреса, телефоны, емейлы или документы, используйте расширенный фильтр на списке клиентов:

▼ Телефон ON

Тип  
(Не задано)

Дата акт.  
(Не задано)

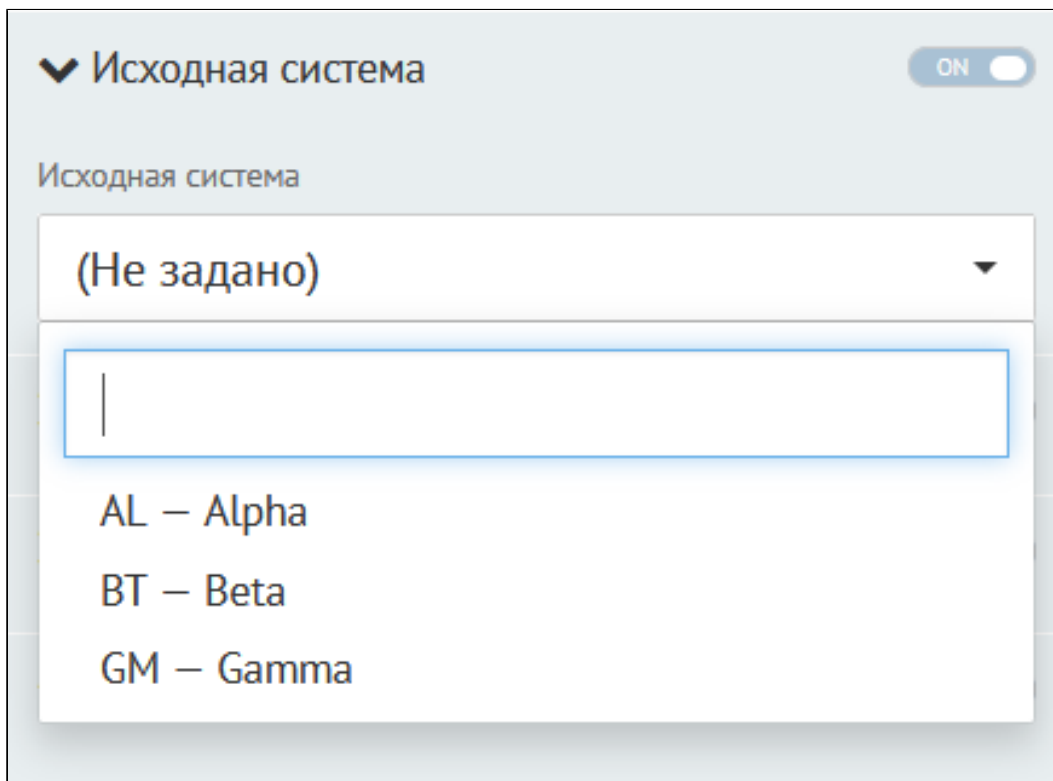
Номер

Популярность

- обычный
- популярный
- массовый

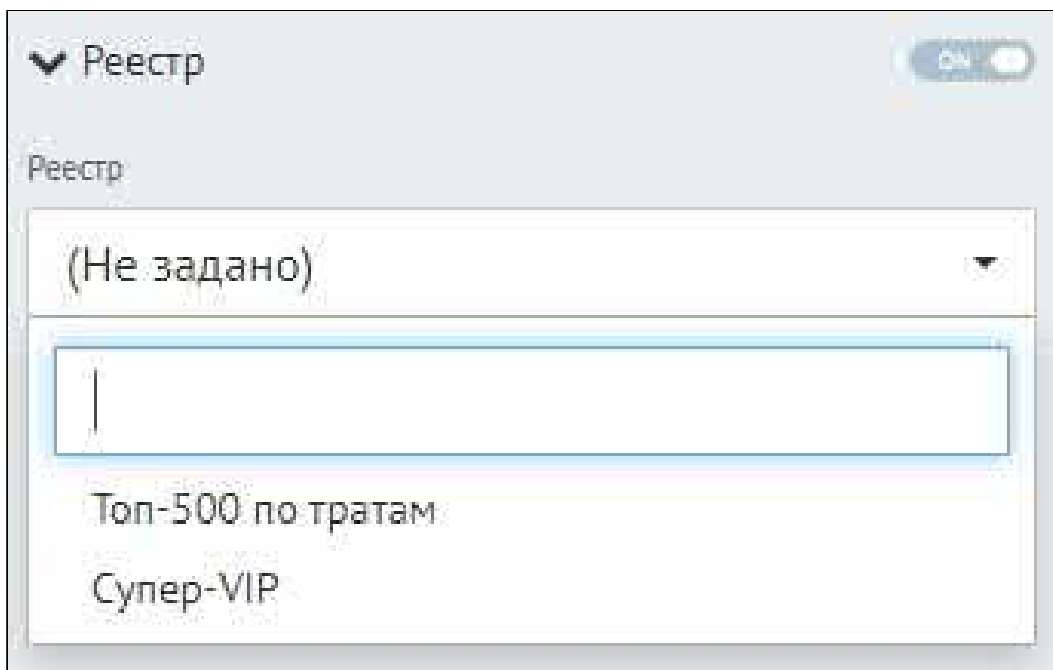
## Фильтр по исходной системе

Фильтрация списка клиентов по системе-источнику доступна в блоке «Исходная система» расширенного фильтра:



### ***Фильтр по вхождению в реестр***

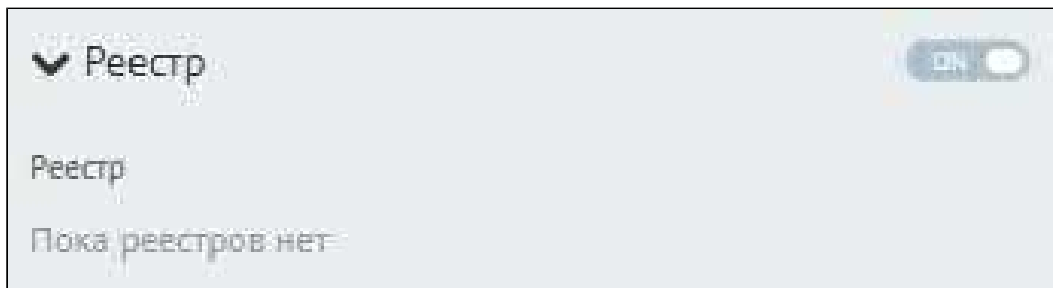
Фильтрация списка клиентов по внешнему реестру доступна в блоке «Реестр» расширенного фильтра:





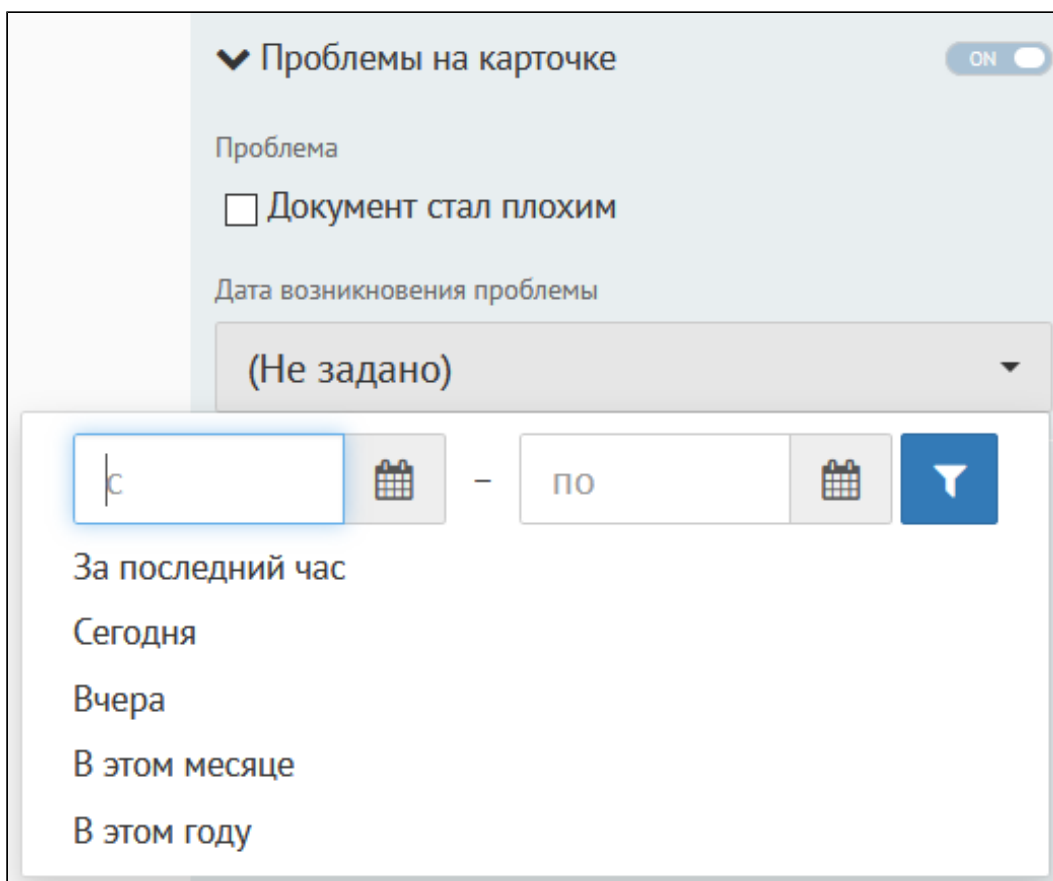
Чтобы список реестров появился в веб-менеджере данных, Администратор предварительно загружает их в специальную таблицу и запускает задачу актуализации реестров.

При отсутствии реестров в Едином клиенте фильтрация недоступна:



### ***Фильтр по проблемам на карточке***

Фильтрация списка клиентов по подключенным проблемам на карточке доступна в блоке «Проблемы на карточке» расширенного фильтра:



При отсутствии подключенных проблем блок не отображается.

## Фильтр для юрлиц

При подключенном модуле проверки по ЕГРЮЛ, на списке юрлиц доступна дополнительная фильтрация:

- *Тип ЮЛ* — клиент является Юрлицом / ИП / Нотариусом / Адвокатом;
- *Объединен с ЕГР* — признак объединения клиента с данными из ЕГРЮЛа;
- *Статус* — статус активности организации.

▼ Юрлицо ON

Тип ЮЛ

(Не задано) ▼

Объединен с ЕГР

(Не задано) ▼

Статус

(Не задано) ▼

## ПЕРЕХОД НА КАРТОЧКУ КОНТРАГЕНТА

Переход на [карточку контрагента](#) осуществляется:

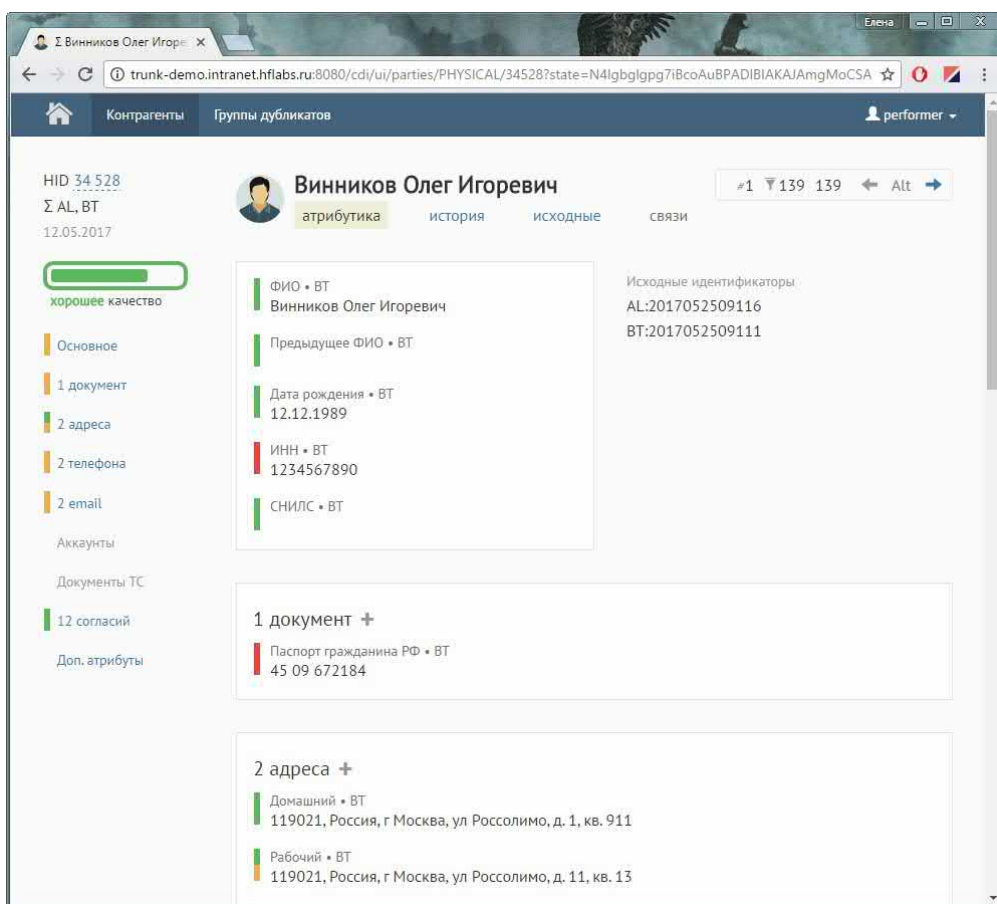
- двойным нажатием левой кнопки мыши по записи в списке;
- установкой курсора на нужную запись и нажатием клавиши **Enter**.

# ПРОСМОТР КАРТОЧКИ КОНТРАГЕНТА

- Увидеть всю информацию по золотой карточке
- Качество клиентской карточки
- Блоки на основной странице
  - Общая информация
  - Прошлые ФИО
  - Подсветка в исходном значении атрибутов «чужеродных» символов
  - Документы
  - Прошлые документы
  - Адреса
  - Телефоны
  - Email
  - Аккаунты
  - Документы ТС
  - Согласия
  - Дополнительные атрибуты
  - Отображение атрибутов
  - Включение табличного режима отображения
  - Управление карточкой в табличном режиме
  - Управление карточкой в табличном режиме с клавиатуры
- Отдельные вкладки
  - Связи
  - История изменения клиентов
    - История изменения золотой карточки
  - Отклоненные дубликаты
- Панель навигации
  - По атрибутам
  - По карточкам клиентов
  - Закрепление секции при листании карточек
- Скопировать идентификатор в один клик

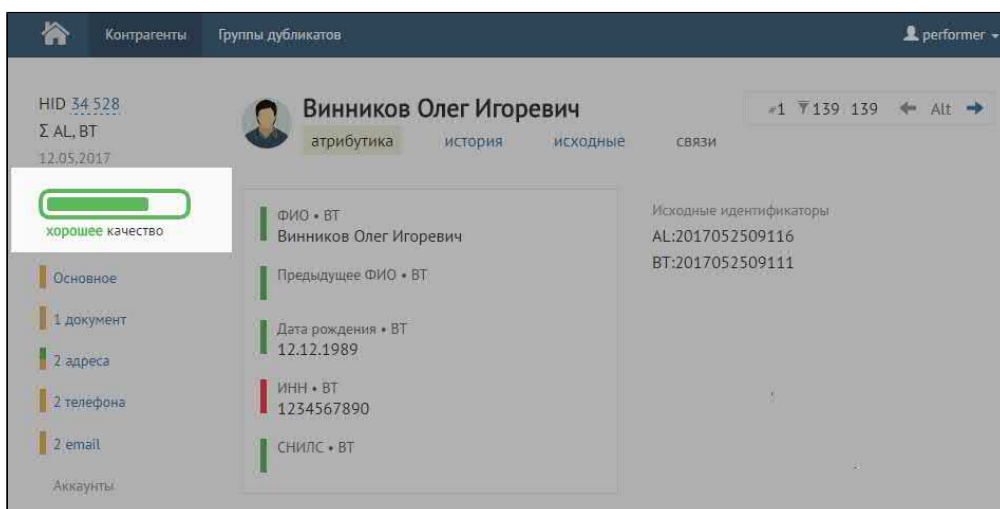
## Увидеть всю информацию по золотой карточке

На карточке клиента дата-стюард видит все атрибуты единым списком. Проблемные данные отмечены желтыми и красными кодами качества.



## Качество клиентской карточки

Качество карточки помогает с одного взгляда оценить, насколько хороша информация по клиенту:



Качество карточки Единый клиент **рассчитывает** по наличию атрибутов (ФИО, ИНН, адресов, телефонов, документов) и их кодам качества.

## Блоки на основной странице

### Общая информация

Блок *Общая информация* содержит главные реквизиты контрагента — ФИО, дата рождения, ИНН, СНИЛС. Они отображаются все, даже если реквизит пустой

ФИО • ВТ	Винников Олег Игоревич
Предыдущее ФИО • ВТ	
Дата рождения • ВТ	12.12.1989
ИНН • ВТ	1234567890
СНИЛС • ВТ	

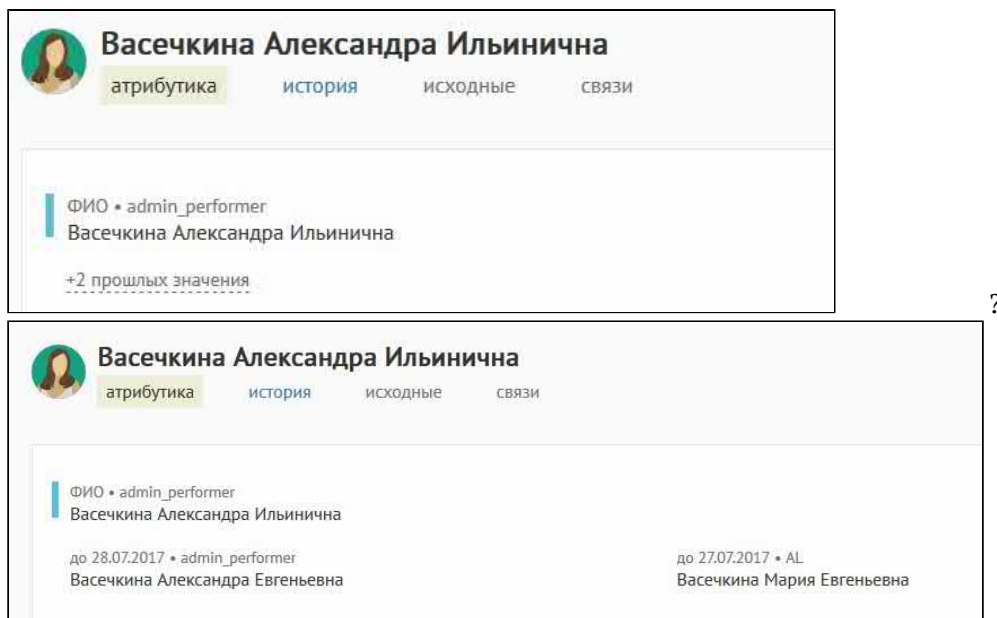
Все остальные реквизиты перенесены в отдельный блок [«Дополнительные атрибуты»](#)

Для золотых карточек дополнительно перечисляются исходные идентификаторы справа от блока с ФИО:

ФИО • GM	Уважаемый Олег Иванович	Исходные идентификаторы AL:20160618-00-131000 AL:20160618-00-131001 AL:20160618-00-1310100 AL:20160622-01-131002 AL:20160622-02-131003 AL:20160622-03-131004 AL:20160622-04-131005 AL:20160622-05-131006 <a href="#">+ еще 297</a>
Предыдущее ФИО • AL		
Дата рождения • GM	01.03.1964	
ИНН • GM	111111110000	
СНИЛС • AL		

## Прошлые ФИО

В блоке *Общая информация* под актуальным ФИО показывается ссылка с количеством прошлых значений (если они вообще есть). Если нажать на ссылку, показываются прошлые значения:



The image shows two screenshots of a user profile for 'Васечкина Александра Ильинична'. The top screenshot shows the current name and a link '+2 прошлых значения'. The bottom screenshot shows the history of names: 'Васечкина Александра Евгеньевна' (from 28.07.2017) and 'Васечкина Мария Евгеньевна' (from 27.07.2017).

**Васечкина Александра Ильинична**  
атрибутика история исходные связи

ФИО • admin\_performer  
Васечкина Александра Ильинична  
[+2 прошлых значения](#)

**Васечкина Александра Ильинична**  
атрибутика история исходные связи

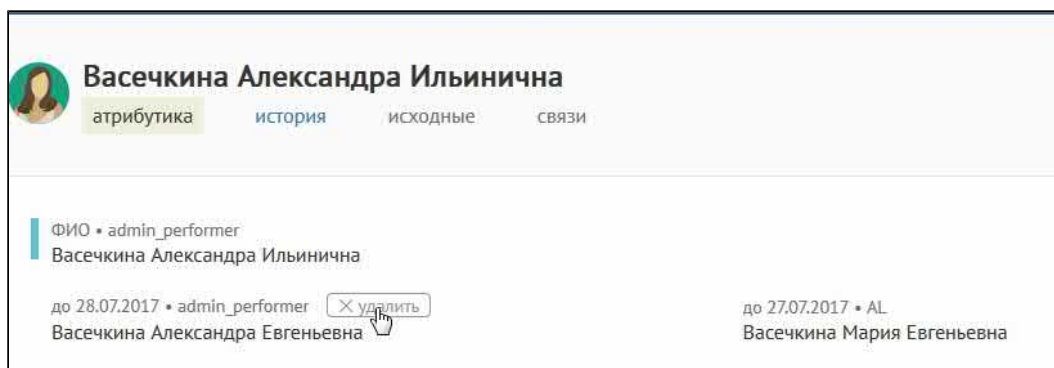
ФИО • admin\_performer  
Васечкина Александра Ильинична

до 28.07.2017 • admin\_performer  
Васечкина Александра Евгеньевна

до 27.07.2017 • AL  
Васечкина Мария Евгеньевна

У них нет кода качества, показывается только дата окончания действия («до 27.07.2017»), исходная система («AL») и значение одной строкой («Васечкина Мария Евгеньевна»). Прошлые значения упорядочены по дате окончания действия в обратном порядке (от новых к старым).

Прошрое значение удаляется без дополнительных подтверждений с помощью кнопки "удалить", которая появляется при наведении курсора на прошлое значение. Удалять прошлые значения могут только дата-стюарды.



The image shows the user profile with a 'удалить' button appearing over the 'Васечкина Александра Евгеньевна' entry.

**Васечкина Александра Ильинична**  
атрибутика история исходные связи

ФИО • admin\_performer  
Васечкина Александра Ильинична

до 28.07.2017 • admin\_performer

Васечкина Александра Евгеньевна

до 27.07.2017 • AL  
Васечкина Мария Евгеньевна

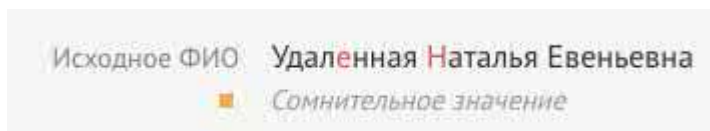
## **Подсветка в исходном значении атрибутов «чужеродных» символов**

Для визуального отличия символов кириллицы от латиницы, в исходном значении атрибута настроена подсветка «чужеродных» символов.

Чужеродные — это:

- если большая часть значения набрана кириллическими символами, то чужеродные — латинские;
- если большая часть набрана латиницей, то чужеродные — кириллические;
- если их равное количество, чужеродные — латинские;
- если кириллических или латинских символов вовсе нет — чужеродных нет.

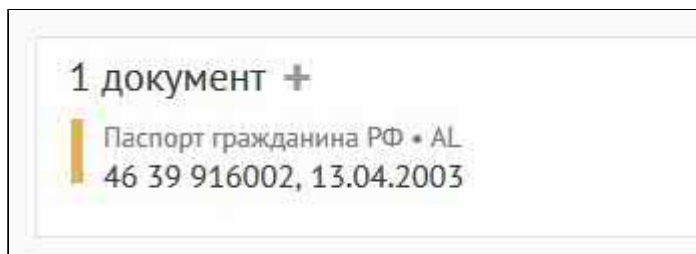
Чужеродные символы, в отличие от обычных — красные:



Чужеродные символы подсвечиваются только на формах редактирования для перечисленных атрибутов — ФИО, Наименование, адрес, email.

## **Документы**

Блок *Документы* содержит список документов контрагента. Для каждого документа доступны: тип, представление в гранулярной форме, исходная информация о документе (одной строкой), код качества.





## Прошлые документы

В блоке *Документы* под актуальным документом показывается ссылка с количеством прошлых значений (если они вообще есть). Если нажать на ссылку, показываются прошлые значения:

1 документ +

Паспорт гражданина РФ • admin\_performer  
42 17 123456, ОВД Мневники, 13.12.2016

[+2 прошлых значения](#)

? ? ?

1 документ +

Паспорт гражданина РФ • admin\_performer  
42 17 123456, ОВД Мневники, 13.12.2016

до 28.07.2017 • admin_performer 41 16 123456	до 28.07.2017 • admin_performer 40 16 123456
---	---

У них нет кода качества, показывается только дата окончания действия («до 28.07.2017»), исходная система («admin\_performer») и значение одной строкой («40 16 123456»). Прошлые значения упорядочены по дате окончания действия в обратном порядке (от новых к старым).

Прошрое значение удаляется без дополнительных подтверждений с помощью кнопки "удалить", которая появляется при наведении курсора на прошлое значение. Удалять прошлые значения могут только дата-стюарды.

1 документ +

Паспорт гражданина РФ • admin\_performer  
42 17 123456, ОВД Мневники, 13.12.2016

до 28.07.2017 • admin_performer 41 16 123456	<input type="button" value="X удалить"/>	до 28.07.2017 • admin_performer 40 16 123456
---	--	---

Прошлые документы не учитываются при подсчёте количества атрибутов. Один актуальный паспорт + 5 прошлых значений = 1 документ.

Если документ был закрыт, то в блоке *Документы* создаётся новое прошлое значение (= значению закрытого документа). Закрытые документы отображаются после актуальных документов. Работа с ними аналогична работе с прошлыми значениями актуальных документов.

### 1 документ +

Паспорт гражданина РФ • admin\_performer  
40 12 664645, ОВД Ненайдено, 13.12.2012

+4 прошлых значения

+3 прошлых значения (свидетельство о рождении)

## Адреса

Блок *Адреса* содержит список адресов контрагента. Для каждого адреса доступны: тип, дата акт. ФИАС, представление в гранулярной форме, исходный адрес (одной строкой), коды качества и проверки.

### 1 адрес +

Иной • AL  
119034, РОССИЯ, Г МОСКВА, Г МОСКВА, ПЕР ТУРЧАНИНОВ, д. 6, стр. 2, кв. 17

## Телефоны

Блок *Телефоны* содержит список телефонов контрагента. Для каждого телефона доступны: тип, представление в гранулярной форме, исходный телефон, код качества.

### 2 телефона +

Мобильный • AL  
7 916 5345622

Домашний • BT  
7 495 2283412

## Email

Блок *Email* содержит список адресов электронной почты контрагента. Для каждого адреса доступны: тип, распознанное значение, исходный адрес электронной почты, код качества, тип домена, дополнительная информация.

2 email +

Домашний • performer vasen`ka@mail.ru	Рабочий • performer ivanov@mail.com
--	--

1 email +

Домашний • admin\_performer • 25.11.2020 • HFLabs-CDI:891000  
firstemail13@mail.ru

Тип

Основной


Исходный

Email

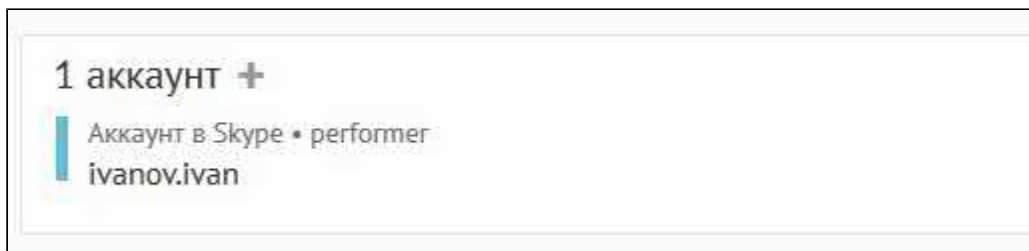
Тип домена

Доп.


## Аккаунты

 Тип атрибута "Аккаунт" не входит в стандартную модель данных и подключается отдельно

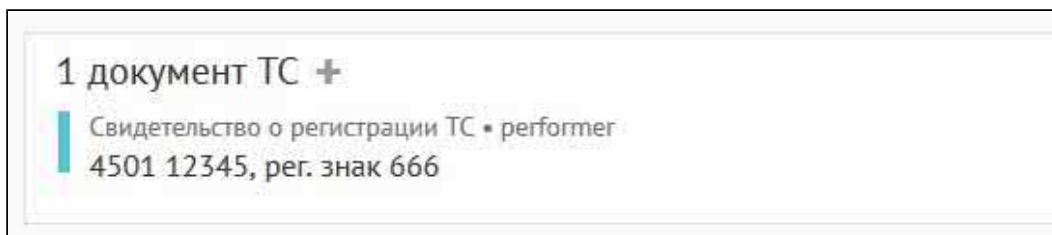
Блок *Аккаунты* содержит список аккаунтов контрагента в социальных сетях и прочих интернет-сервисах. Для каждого аккаунта доступны: тип, распознанное значение, исходное значение, код качества.



## ***Документы ТС***

 Тип атрибута "Документ ТС" не входит в стандартную модель данных и подключается отдельно

Блок *Документы ТС* содержит список документов транспортного средства контрагента. Для каждого документа доступны: тип, серия, номер, регистрационный знак, исходное значение, код качества.



## ***Согласия***

Блок *Согласия* содержит список согласий данного контрагента и их значения.

12 согласий +

Обработка персональных данных ..... предоставлено ▾

Передача данных третьим лицам ..... предоставлено ▾

Реклама товаров и услуг ..... предоставлено ▾

Выдача карты ..... нет данных ▾

Контакт по

SMS ..... предоставлено ▾

почте ..... предоставлено ▾

телефону ..... предоставлено ▾

Не беспокоить по

домашнему телефону ..... нет данных ▾

мобильному телефону ..... нет данных ▾

рабочему телефону ..... нет данных ▾

[Все согласия](#)

Нажмите на кнопку «Все согласия» и увидите полную информацию: тип, значение, автора и дату актуальности.

Пустое поле "Автор" означает, что согласие является согласием по умолчанию.

Общая информация | Документы | 1 адрес | Телефоны | Email | Аккаунты | Документы ТС | 12 согласий ✕ | Дополнительные атрибуты

Согласие на обработку перс данных менял операционист, остальные -> значения по умолчанию, так как автор пустой

Автор	Тип	Согласие	Дата акт.
performer	Обработка персональных данных	Получен отказ ▾	15.01.2016
	Реклама товаров и услуг	Предоставлено ▾	
	Обработка персональных данных с целью продвижения ...	Предоставлено ▾	
	Передача данных третьим лицам	Предоставлено ▾	
	Контакт по телефону	Предоставлено ▾	
	Контакт по SMS	Предоставлено ▾	

Автор: Все

Тип: Все

Выдача карты

Контакт по SMS

Контакт по почте

Контакт по телефону

Контакт по электронной почте

Не беспокоить по домашнему телефону

Не беспокоить по мобильному телефону

## Дополнительные атрибуты

Блок «Дополнительные атрибуты» содержит атомарные (не составные) атрибуты контрагента. Если поле не заполнено, отображается название атрибута без автора. Если атрибут заполнен, то отображается его название, значение и автор.

## Доп. атрибуты

Идентификатор во внешней системе • ВТ  
AL:13

Количество иждивенцев • admin\_performer  
1

Место работы • ВТ  
ООО Рога и Копыта

Дата начала работы • admin\_performer  
01.01.2011

Отделение • admin\_performer  
отделение

+ Резидентство

Автор

Резидентство

Неизвестно ▾

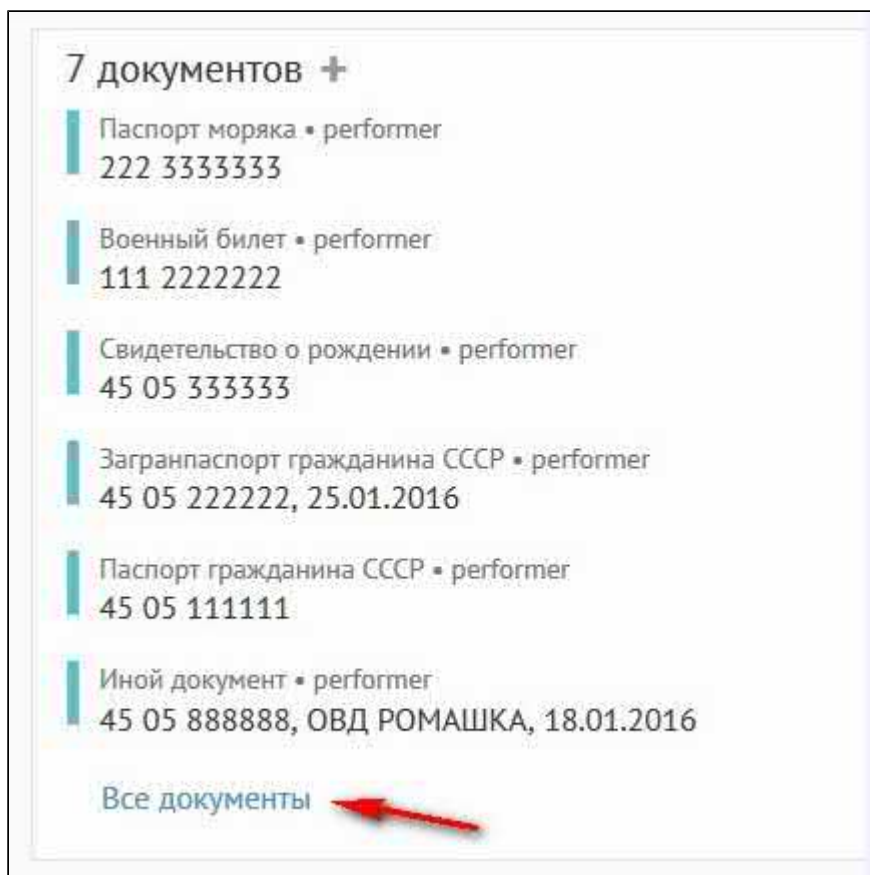
Подтвердить

+ Полученное образование

## Отображение атрибутов

Показываем стандартизованное значение атрибута одной строкой, а если оно пустое, то исходное.

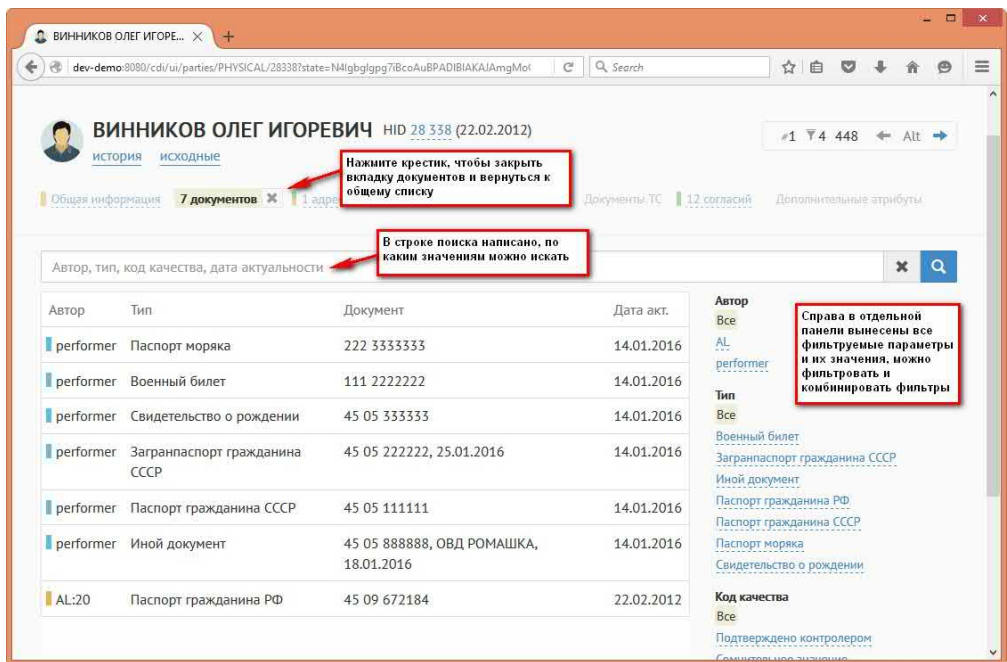
Адреса и документы показываем в один столбец (максимум 6 штук), телефоны, email и аккаунты — в три (максимум 11 штук). Остальное — по клику на «все атрибуты».



В режиме «Все атрибуты» отображаются только выбранные атрибуты клиента, общая информация и все остальные атрибуты скрыты.

Есть строка поиска. Справа расположены фильтры.

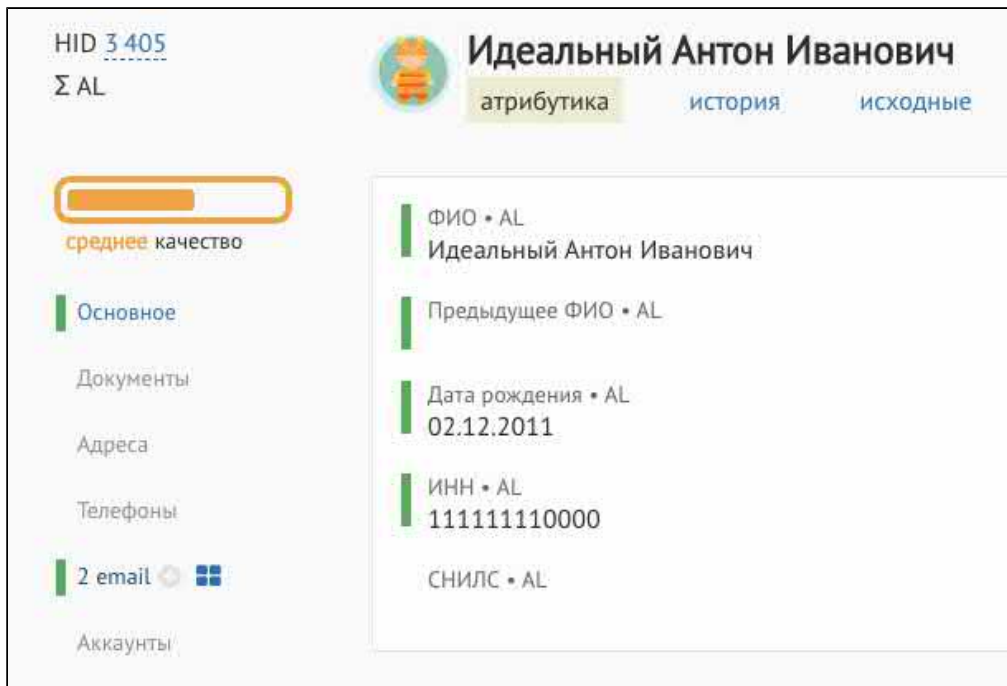
Чтобы вернуться к полной карточке клиента — нажмите крестик около атрибутов в верхней навигационной панели



## Включение табличного режима отображения

Если дата-стюарду требуется концентрированно разбирать телефоны, адреса или атрибуты другого типа, в веб-менеджере предусмотрен табличный режим работы.

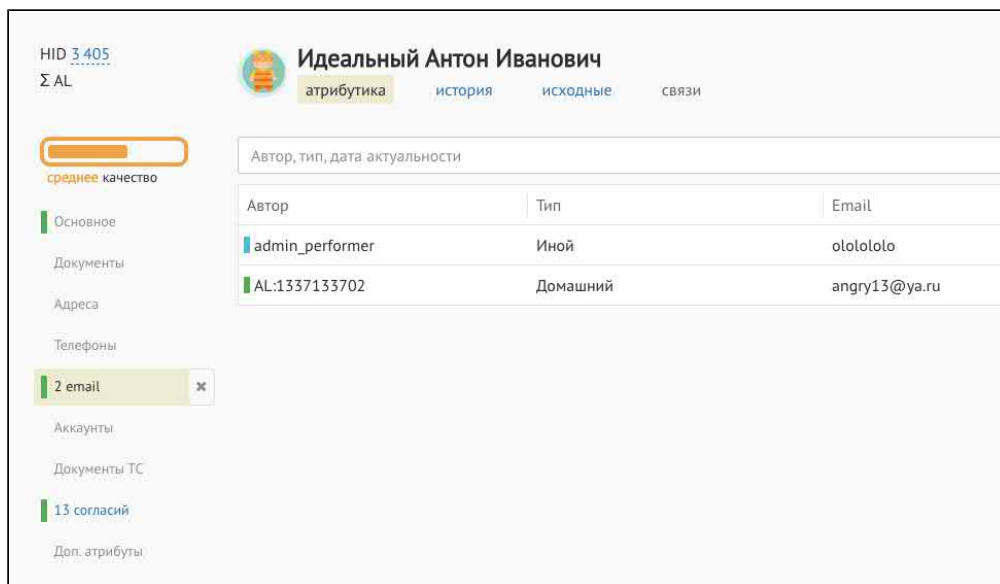
Перейти в него можно, нажав на специальную иконку напротив нужного атрибута.





## Управление карточкой в табличном режиме

Находясь в табличном режиме можно видеть все атрибуты в виде таблицы, изменять ширину колонок таблицы, фильтровать в ней данные.



Если для табличного отображения доступны атрибуты другого типа, то переключиться на них можно, используя левое меню. При этом нет необходимости покидать табличный режим. При этом настройки таблицы сохраняются для каждого конкретного типа атрибутов по отдельности.

Переключаясь между контрагентами, отображение нужного табличного режима сохраняется.

Чтобы вернуться к стандартному отображению карточки контрагента, нужно нажать на «крестик» в левом меню.

## Управление карточкой в табличном режиме с клавиатуры

Перемещение по элементам в табличном режиме возможно с помощью [горячих клавиш](#).

## Отдельные вкладки

### Связи

Часто Единый клиент знает:

- о договорах, которые заключил клиент;

взаимосвязях клиентов между собой (сотрудник-работодатель, страхователь-застрахованный, муж-жена).

Эта информация помогает дата-стюардам лучше разрешать потенциальные дубликаты, а маркетологам — строить аналитику по клиентской базе.

Дата-стюард переходит к связям по ссылке из карточки клиента:

**МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ** ID **2048** (07.07.2015)

[история](#) [3 связи](#)

Общая информация | 2 документа | 6 адресов | 10 телефонов | 1 email | Аккаунты | Докум

ФИО • admin\_performer  
МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ

Дата рождения • performer  
26.11.1990

ИНН • performer  
743000228005

Список связей дата-стюард фильтрует по виду связи или связанному объекту:

**МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ** СВЯЗИ

3 связи: [застрахованный \(1\)](#) [созаемщик по отношению к \(1\)](#) [страхователь \(1\)](#)

Связаны: [физ. лица \(1\)](#) [договоры \(2\)](#)

Созаемщик по отношению к	РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ
Страхователь	Договор 130000025
Застрахованный	Договор 130000022


По клику на строку списка Единый клиент показывает подробности:

**МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ** СВЯЗИ

3 связи: [застрахованный \(1\)](#) [созаемщик по отношению к \(1\)](#) [страхователь \(1\)](#)

Связаны: [физ. лица \(1\)](#) [договоры \(2\)](#)

Созаемщик по отношению к **РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ**

 **РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ** ИД 2051  
 Источник: AL:11,AL:12,GM:2015-03-1554

ФИО  
 РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ

Дата рождения  
 05.03.1987

ИНН  
 753001686893

СНИЛС  
 123-456-789 01

<b>Телефоны</b>	<b>Адреса</b>
Иной 7 495 4886881 доб. 1	Иной 309311, РОССИЯ, ОБЛ БЕЛГОРОДСКАЯ, Р-Н РАКИТЯН...
Для SMS-оповещений 7 495 4887777 доб. 2	Регистрация 309311, РОССИЯ, ОБЛ БЕЛГОРОДСКАЯ, Р-Н РАКИТЯН...
Телефон родственников/знак 7 495 4886883 доб. 3	Прописка 309311, РОССИЯ, ОБЛ БЕЛГОРОДСКАЯ, Р-Н РАКИТЯН...

**Документы**

Загранпаспорт гражданина РФ  
 0 7899877, ВОЗЛЕ ДОМА, 02.02.2002


## ***История изменения клиентов***


Хороший клиент остается с компанией надолго. Пока идут годы, люди женятся и разводятся, переезжают, меняют номера телефонов, работу и документы, покупают новые продукты компании. Эти события попадают в информационные системы организации, а оттуда — в «Единый клиент».

С другой стороны, дата-стюарды выверяют контактные и паспортные данные, исправляют ошибки и опечатки, объединяют одинаковых клиентов. Эти события «Единый клиент» тоже запоминает.

Сводную историю изменений по человеку видно на карточке:

HID [2 203](#)  
 AL:2017071511071


**Московский Юрий Олегович**  
 атрибутика | история | исходные | связи

  
 среднее качество


Основное  
 1 документ

ФИО • AL  
 Московский Юрий Олегович

Предыдущее ФИО • AL

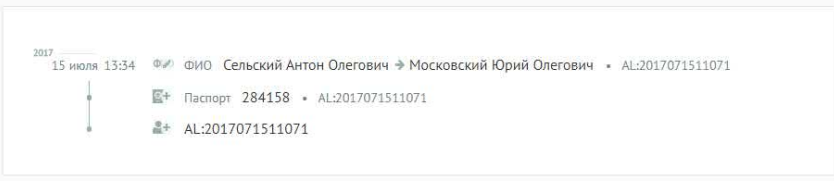
«Единый клиент» увязывает все события в единую последовательность; показывает, что и когда изменилось:


HID [2 203](#)  
 AL:2017071511071


**Московский Юрий Олегович**  
 атрибутика | история | исходные | связи

#82 | 85 | 85 | ← Alt →

**События**  
 создание (2)  
 изменение (1)  
 удаление  
 слияние  
 разлияние  
 восстановление






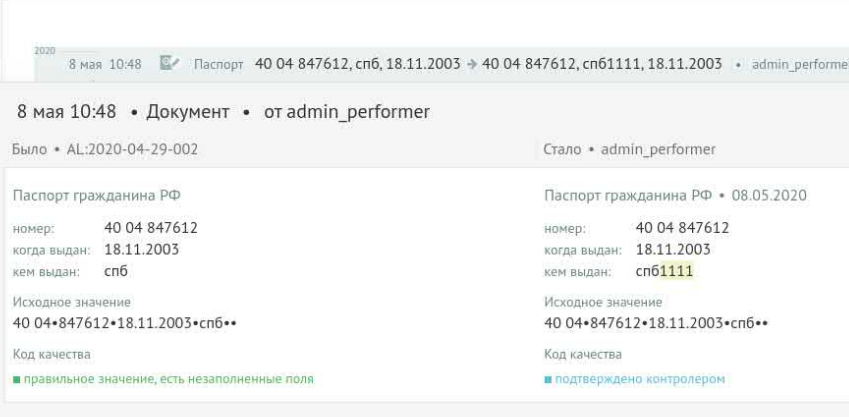
Если развернуть строку с событием, то показывает не только, что изменилось, но и кто автор изменения:


Контрагенты | Группы дубликатов | Конфликты | Отчёты | Администрирование

HID [890 482](#)  
 Σ AL


**Стежкин Владимир Николаевич**  
 атрибутика | история | исходные | связи | нет похожих

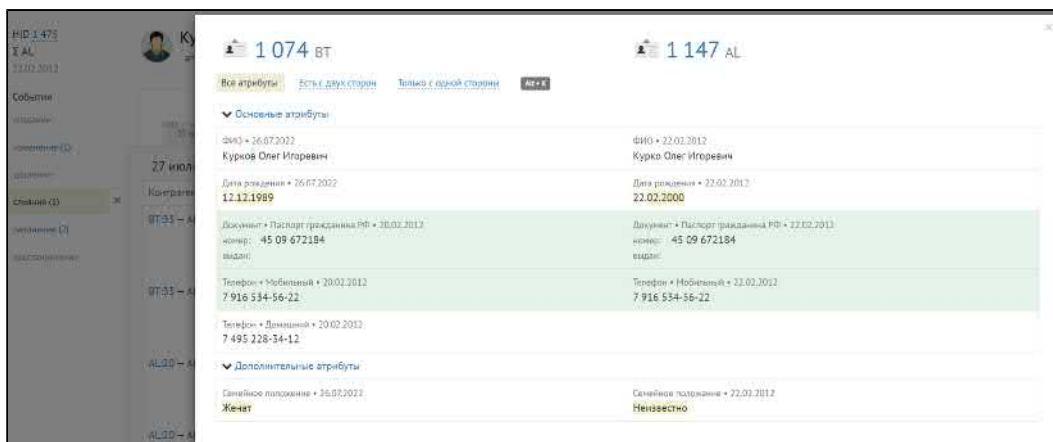
**События**  
 создание  
 изменение (1)  
 удаление  
 слияние (1)  
 разлияние  
 восстановление





29 апр. 17:44 | AL:001 + AL:002 → Σ AL

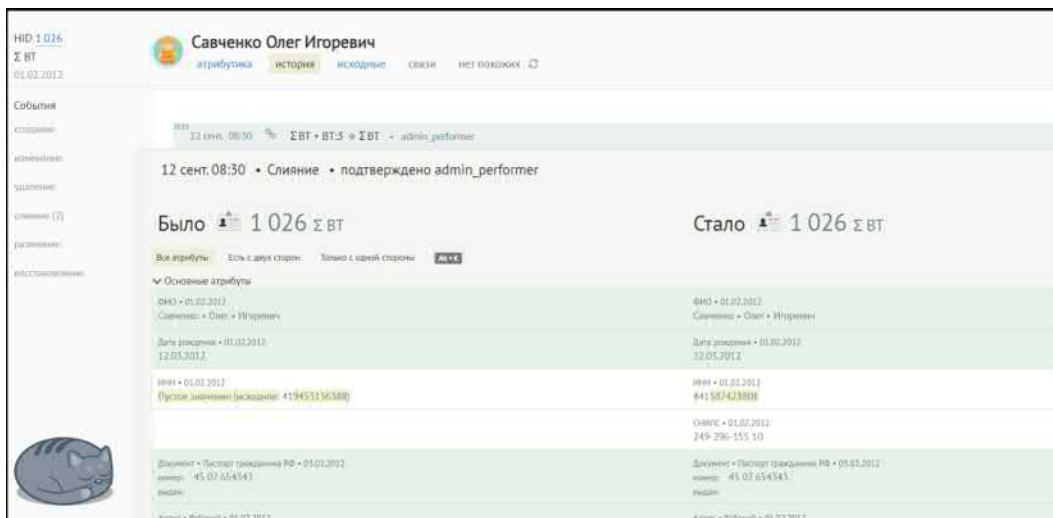
При клике на событие слияния открывается окно со сравнением дубликатов — исходных карточек именно на момент слияния:



## История изменения золотой карточки

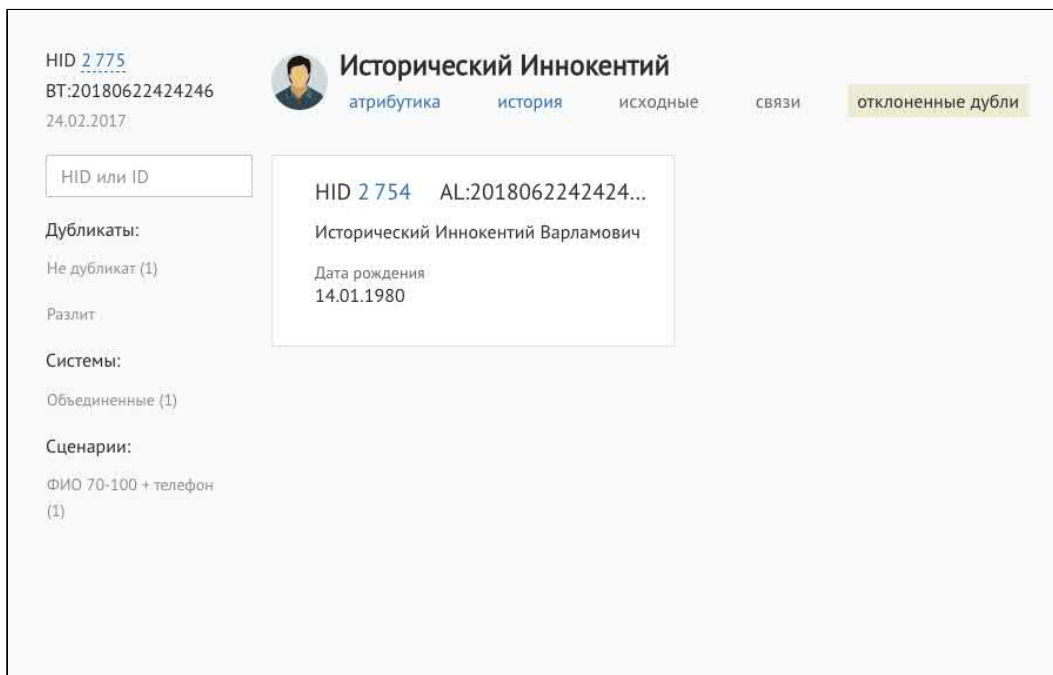
В случаях, когда в золотую карточку подливается одна или несколько новых карточек, чьи атрибуты или реквизиты выигрывают, в истории ЗК для события слияния отображаются изменения в реквизитах и атрибутах золотой карточки.

При клике на событие слияния ЗК с новыми карточками, появляется выпадающее окно со сравнением золотой карточки — до и после слияния:



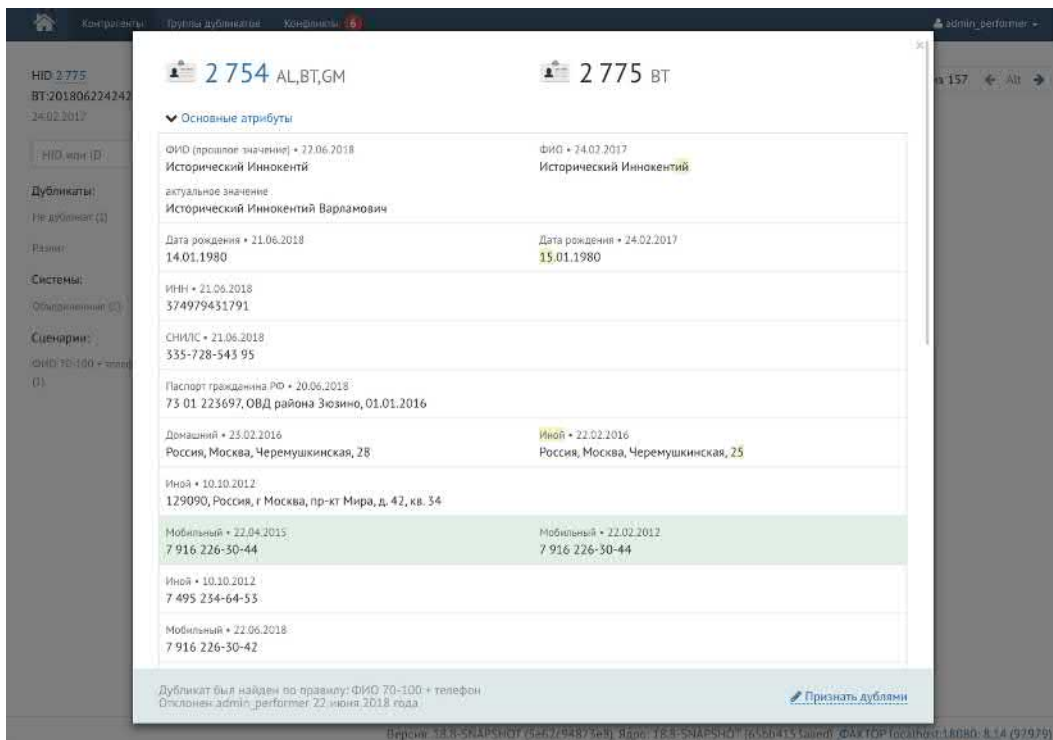
## Отклоненные дубликаты

Если «Единый клиент» находил похожие карточки, но дата-стюард сказал, что они «Не дубликаты» — такие пары больше не появятся в разделе «Группы дубликатов». Найти их можно в разделе «Отклоненные дубликаты» на карточке контрагента.



Кроме того, в этом разделе показаны и ранее разъединенные карточки. В левой части есть удобный фильтр по типу пары, исходным системам и правилам.

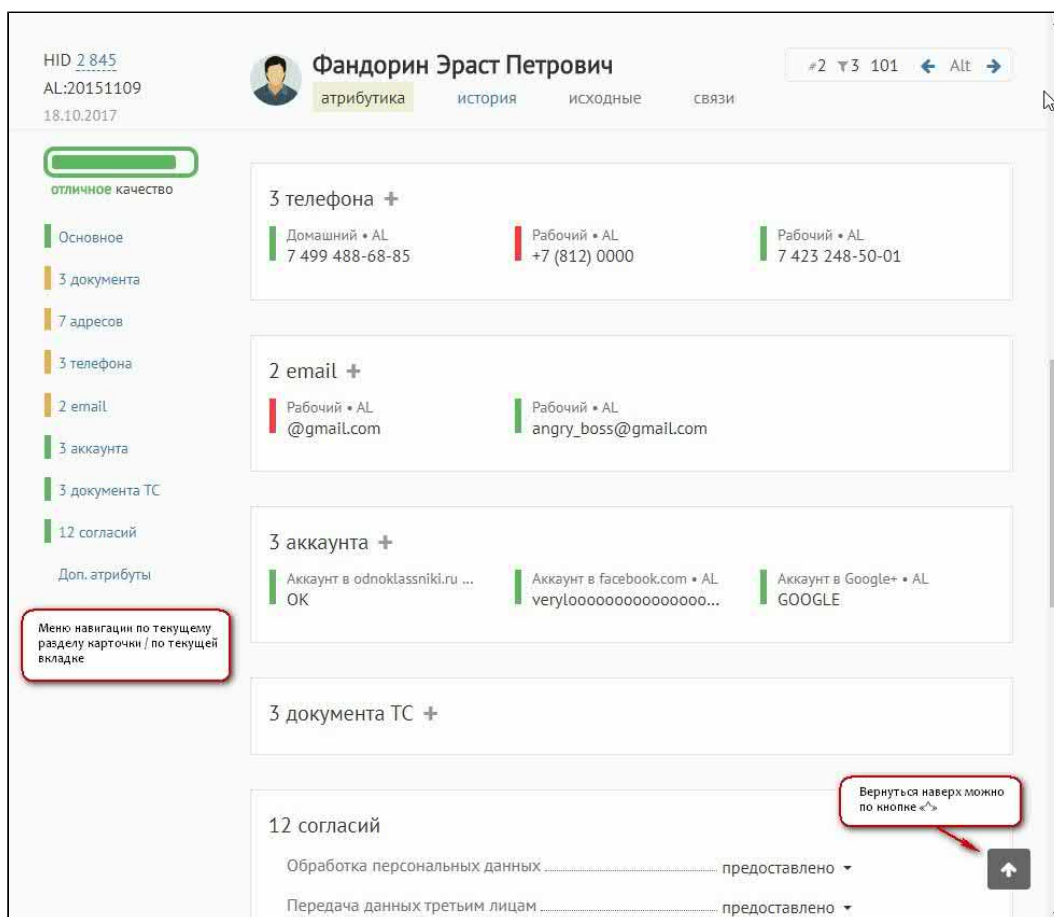
Нажав на карточку, можно увидеть подробности: сравнение, информацию об истории статуса, а также возможность изменить решение.



## Панель навигации

### По атрибутам

На карточке закреплена панель навигации по атрибутам клиента. Нажмите на любой атрибут — и вы переместитесь прямо к нему

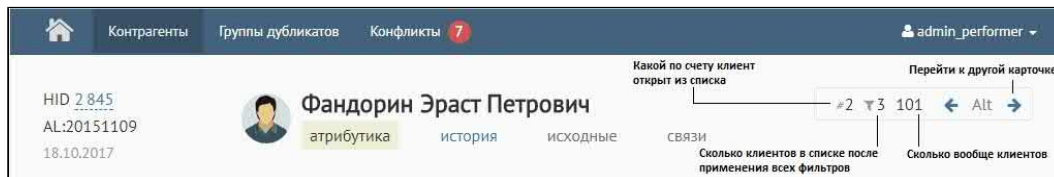


Панель не пропадает, всегда можно вернуться обратно или перейти дальше.

Так можно быстро пройтись по некорректным атрибутам:

- видим в панели желтую или красную плашку — как минимум один атрибут некорректен;
- переходим к атрибуту, исправляем;
- проверяем по панели, где еще остались желтые и красные плашки.

## По карточкам клиентов



Панель навигации располагается в верхней части окна, содержит набор кнопок для навигации по карточкам клиентов и информацию о текущих фильтрации и положении в списке.

Кнопки панели навигации (в скобках приведены соответствующие "горячие клавиши"):

- **Назад** — просмотреть предыдущую карточку контрагента (**Alt + Left**).
- **Обновить** — обновить текущую закладку карточки контрагента (**F5**).
- **Вперёд** — просмотреть следующую карточку контрагента (**Alt + Right**).
- **Вернуться к списку** — вернуться обратно к [списку контрагентов](#) (**Alt + Up**).

Информация о текущем положении в списке контрагентов выводится в виде **А В С**, где:

- **А** - текущая позиция в списке;
- **В** - количество записей, удовлетворяющих установленным условиям фильтрации / поиска;
- **С** - общее количество записей о контрагентах.

## Закрепление секции при листании карточек

Клик по ссылке на группу в левом меню закрепляет секцию для листания карточек.

При переходе к следующей / предыдущей карточке через ? **Alt** ? срабатывает автоскролл к закрепленной ранее секции атрибутов.

Если данного вида атрибутов у клиента нет, то закрепление не сбрасывается, срабатывает автоскролл к предыдущей видимой секции. Как только при листании дальше нужная секция появляется ? показываем её.

Сброс закрепления секции выполняется при:



- выходе в список,
- нажатии кнопки «Наверх»,
- ручном скролле на самый верх карточки (когда хедер отображается).

Без закрепления секции следующая карточка открывается с секции «Основное», при это неважно куда пользователь вручную проскроллил текущую.



## Скопировать идентификатор в один клик

1. Когда дата-стюард хочет прислать коллеге список клиентов, проще всего использовать исторический идентификатор (HID). Скопировать его с карточки клиента можно в один клик.
2. Скопировать можно и RawID для поиска контрагента в исходной системе, что особенно полезно, когда идентификатор слишком длинный и отображается не полностью.

<b>HID 889 165</b> AL:987236 06.05.2017		<b>Римский Василий Федорович</b> атрибутика	история	исходные	связи	нет похожих 
---	---	--	---------	----------	-------	---

# ПРОСМОТР КАРТОЧКИ. ОБЩАЯ ИНФОРМАЦИЯ

## Общая информация

Блок *Общая информация* содержит главные реквизиты контрагента — ФИО, дата рождения, ИНН, СНИЛС. Они отображаются все, даже если реквизит пустой

ФИО • ВТ	Винников Олег Игоревич
Предыдущее ФИО • ВТ	
Дата рождения • ВТ	12.12.1989
ИНН • ВТ	1234567890
СНИЛС • ВТ	

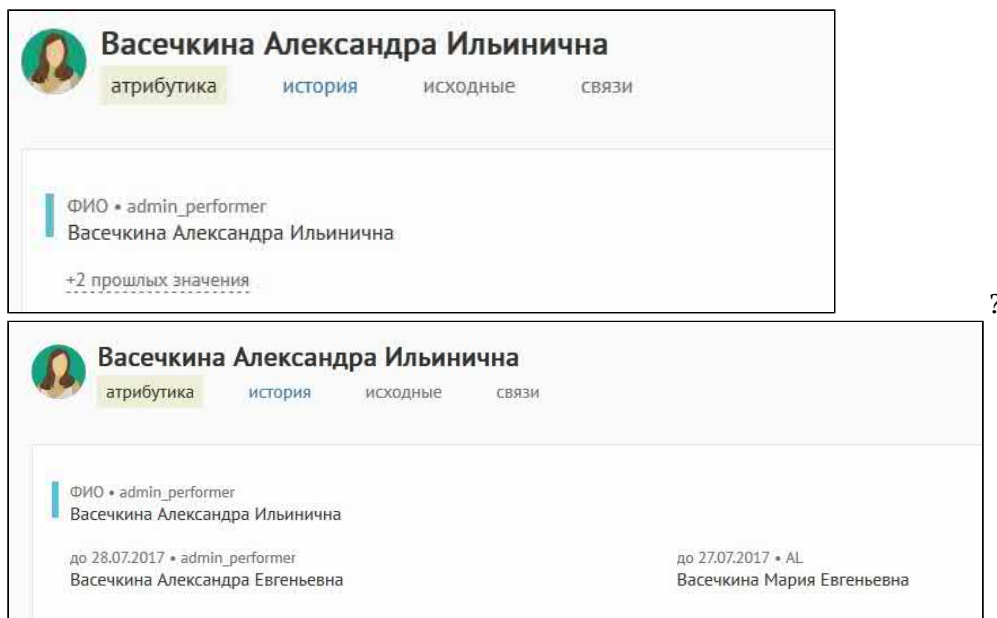
Все остальные реквизиты перенесены в отдельный блок [«Дополнительные атрибуты»](#)

Для золотых карточек дополнительно перечисляются исходные идентификаторы справа от блока с ФИО:

ФИО • GM	Уважаемый Олег Иванович	Исходные идентификаторы AL:20160618-00-131000 AL:20160618-00-131001 AL:20160618-00-1310100 AL:20160622-01-131002 AL:20160622-02-131003 AL:20160622-03-131004 AL:20160622-04-131005 AL:20160622-05-131006 + еще 297
Предыдущее ФИО • AL		
Дата рождения • GM	01.03.1964	
ИНН • GM	111111110000	
СНИЛС • AL		
		<a href="#">Ссылка на вкладку исходные</a>

## Прошлые ФИО

В блоке *Общая информация* под актуальным ФИО показывается ссылка с количеством прошлых значений (если они вообще есть). Если нажать на ссылку, показываются прошлые значения:



The image shows two screenshots of a user profile for 'Васечкина Александра Ильинична'. The top screenshot shows the current name and a link '+2 прошлых значения'. The bottom screenshot shows the history of names: 'Васечкина Александра Ильинична' (до 28.07.2017) and 'Васечкина Мария Евгеньевна' (до 27.07.2017).

**Васечкина Александра Ильинична**  
атрибутика история исходные связи

ФИО • admin\_performer  
Васечкина Александра Ильинична  
[+2 прошлых значения](#)

**Васечкина Александра Ильинична**  
атрибутика история исходные связи

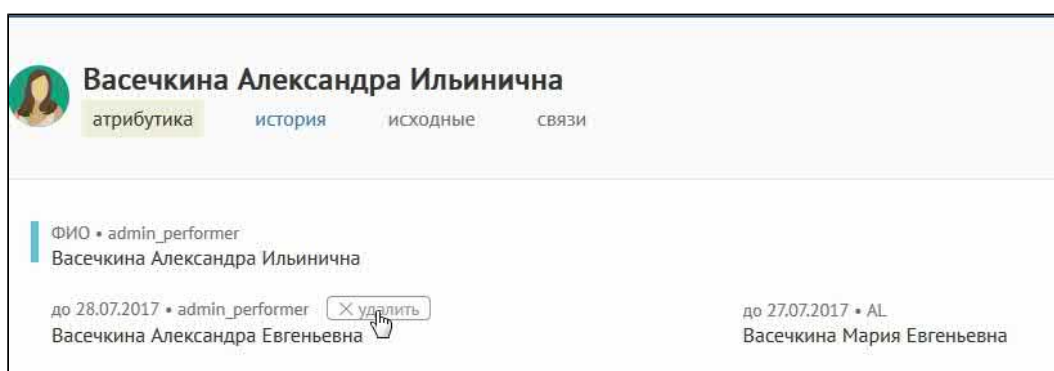
ФИО • admin\_performer  
Васечкина Александра Ильинична

до 28.07.2017 • admin\_performer  
Васечкина Александра Евгеньевна

до 27.07.2017 • AL  
Васечкина Мария Евгеньевна

У них нет кода качества, показывается только дата окончания действия («до 27.07.2017»), исходная система («AL») и значение одной строкой («Васечкина Мария Евгеньевна»). Прошлые значения упорядочены по дате окончания действия в обратном порядке (от новых к старым).

Прошрое значение удаляется без дополнительных подтверждений с помощью кнопки "удалить", которая появляется при наведении курсора на прошлое значение. Удалять прошлые значения могут только дата-стюарды.



The image shows the user profile with a 'удалить' button next to the past name 'Васечкина Александра Евгеньевна'.

**Васечкина Александра Ильинична**  
атрибутика история исходные связи

ФИО • admin\_performer  
Васечкина Александра Ильинична

до 28.07.2017 • admin\_performer

Васечкина Александра Евгеньевна

до 27.07.2017 • AL  
Васечкина Мария Евгеньевна

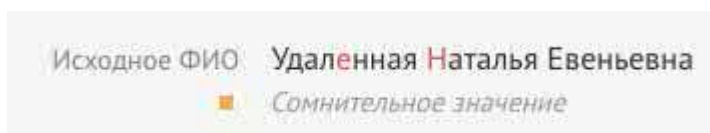
## ***Подсветка в исходном значении атрибутов «чужеродных» символов***

Для визуального отличия символов кириллицы от латиницы, в исходном значении атрибута настроена подсветка «чужеродных» символов.

Чужеродные — это:

- если большая часть значения набрана кириллическими символами, то чужеродные — латинские;
- если большая часть набрана латиницей, то чужеродные — кириллические;
- если их равное количество, чужеродные — латинские;
- если кириллических или латинских символов вовсе нет — чужеродных нет.

Чужеродные символы, в отличие от обычных — красные:

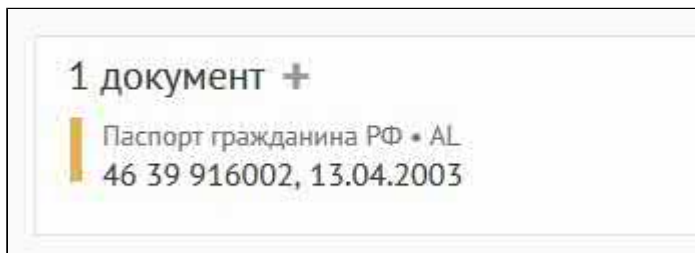


Чужеродные символы подсвечиваются только на формах редактирования для перечисленных атрибутов — ФИО, Наименование, адрес, email.

# ПРОСМОТР КАРТОЧКИ. ДОКУМЕНТЫ

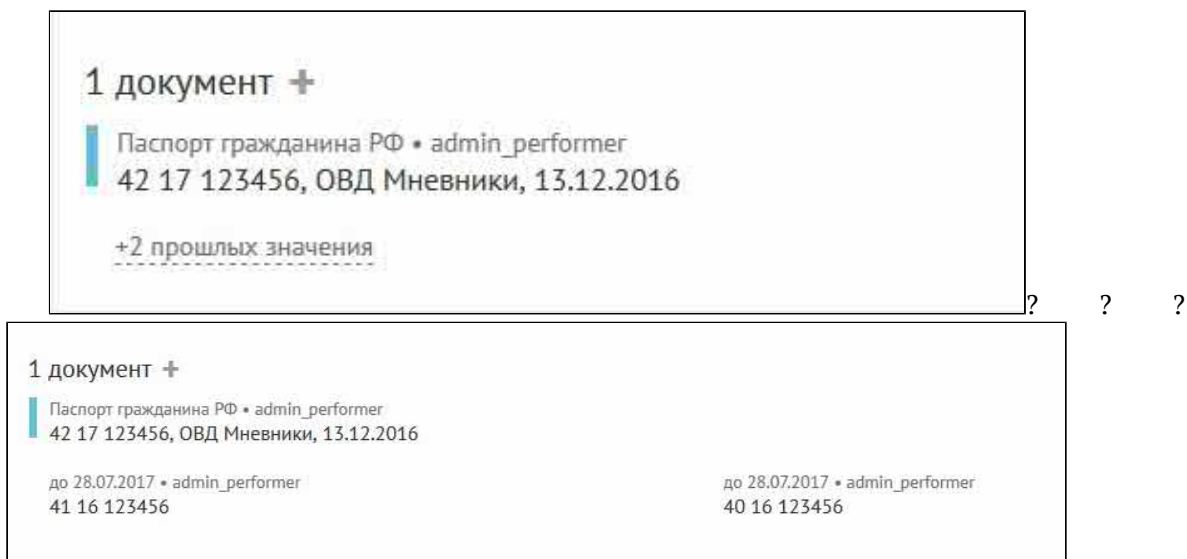
## Документы

Блок *Документы* содержит список документов контрагента. Для каждого документа доступны: тип, представление в гранулярной форме, исходная информация о документе (одной строкой), код качества.



## Прошлые документы

В блоке *Документы* под актуальным документом показывается ссылка с количеством прошлых значений (если они вообще есть). Если нажать на ссылку, показываются прошлые значения:



У них нет кода качества, показывается только дата окончания действия («до 28.07.2017»), исходная система («admin\_performer») и значение одной строкой («40 16 123456»). Прошлые значения упорядочены по дате окончания действия в обратном порядке (от новых к старым).

Прошрое значение удаляется без дополнительных подтверждений с помощью кнопки "удалить", которая появляется при наведении курсора на прошлое значение. Удалять прошлые значения могут только дата-стюарды.

1 документ +

Паспорт гражданина РФ • admin\_performer  
42 17 123456, ОВД Мневники, 13.12.2016

до 28.07.2017 • admin\_performer 41 16 123456

до 28.07.2017 • admin\_performer 40 16 123456

Прошлые документы не учитываются при подсчёте количества атрибутов. Один актуальный паспорт + 5 прошлых значений = 1 документ.

Если документ был закрыт, то в блоке *Документы* создаётся новое прошлое значение (= значению закрытого документа). Закрытые документы отображаются после актуальных документов. Работа с ними аналогична работе с прошлыми значениями актуальных документов.

1 документ +

Паспорт гражданина РФ • admin\_performer  
40 12 664645, ОВД Ненайдено, 13.12.2012

+4 прошлых значения

+3 прошлых значения (свидетельство о рождении)

# ПРОСМОТР КАРТОЧКИ. АДРЕСА

## *Адреса*

Блок *Адреса* содержит список адресов контрагента. Для каждого адреса доступны: тип, дата акт. ФИАС, представление в гранулярной форме, исходный адрес (одной строкой), коды качества и проверки.

1 адрес +

Иной • AL  
119034, РОССИЯ, Г МОСКВА, Г МОСКВА, ПЕР ТУРЧАНИНОВ, д. 6, стр. 2, кв. 17

# ПРОСМОТР КАРТОЧКИ. ТЕЛЕФОНЫ

## Телефоны

Блок *Телефоны* содержит список телефонов контрагента. Для каждого телефона доступны: тип, представление в гранулярной форме, исходный телефон, код качества.





## ПРОСМОТР КАРТОЧКИ. EMAIL

### Email

Блок *Email* содержит список адресов электронной почты контрагента. Для каждого адреса доступны: тип, распознанное значение, исходный адрес электронной почты, код качества, тип домена, дополнительная информация.

2 email +

Домашний • performer vasen`ka@mail.ru	Рабочий • performer ivanov@mail.com
--	--

1 email +

Домашний • admin\_performer • 25.11.2020 • HFLabs-CDI:891000  
firstemail13@mail.ru

Тип

Основной

Исходный


Email

Тип домена

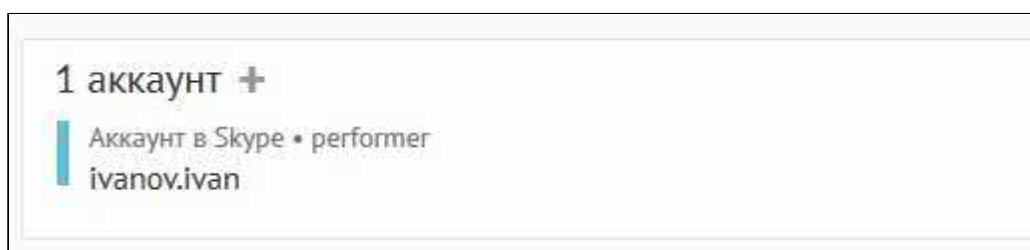
Доп.

# ПРОСМОТР КАРТОЧКИ. АККАУНТЫ

## *Аккаунты*


 Тип атрибута "Аккаунт" не входит в стандартную модель данных и подключается отдельно

Блок *Аккаунты* содержит список аккаунтов контрагента в социальных сетях и прочих интернет-сервисах. Для каждого аккаунта доступны: тип, распознанное значение, исходное значение, код качества.




# ПРОСМОТР КАРТОЧКИ. ДОКУМЕНТЫ ТС

## Документы ТС

 Тип атрибута "Документ ТС" не входит в стандартную модель данных и подключается отдельно

Блок *Документы ТС* содержит список документов транспортного средства контрагента. Для каждого документа доступны: тип, серия, номер, регистрационный знак, исходное значение, код качества.

1 документ ТС +

 Свидетельство о регистрации ТС • performer  
4501 12345, рег. знак 666

# ПРОСМОТР КАРТОЧКИ. СОГЛАСИЯ

## Согласия

Блок *Согласия* содержит список согласий данного контрагента и их значения.

12 согласий +

- Обработка персональных данных ..... предоставлено ▾
- Передача данных третьим лицам ..... предоставлено ▾
- Реклама товаров и услуг ..... предоставлено ▾
- Выдача карты ..... нет данных ▾

Контакт по

- SMS ..... предоставлено ▾
- почте ..... предоставлено ▾
- телефону ..... предоставлено ▾

Не беспокоить по

- домашнему телефону ..... нет данных ▾
- мобильному телефону ..... нет данных ▾
- рабочему телефону ..... нет данных ▾

[Все согласия](#)

Нажмите на кнопку «Все согласия» и увидите полную информацию: тип, значение, автора и дату актуальности.

Пустое поле "Автор" означает, что согласие является согласием по умолчанию.

Общая информация | Документы | 1 адрес | Телефоны | Email | Аккаунты | Документы ТС | 12 согласий ✕ | Дополнительные атрибуты

Согласие на обработку перс данных менял операционист, остальные --> значения по умолчанию, так как автор пустой

Автор	Тип	Согласие	Дата акт.	Автор
performer	Обработка персональных данных	Получен отказ ▾	15.01.2016	performer
	Реклама товаров и услуг	Предоставлено ▾		Тип
	Обработка персональных данных с целью продвижения ...	Предоставлено ▾		Все
	Передача данных третьим лицам	Предоставлено ▾		Выдача карты
	Контакт по телефону	Предоставлено ▾		Контакт по SMS
	Контакт по SMS	Предоставлено ▾		Контакт по почте
				Контакт по телефону
				Контакт по электронной почте
				Не беспокоить по домашнему телефону
				Не беспокоить по мобильному телефону

# ПРОСМОТР КАРТОЧКИ. ДОПОЛНИТЕЛЬНЫЕ АТТРИБУТЫ

## *Дополнительные атрибуты*

Блок «*Дополнительные атрибуты*» содержит атомарные (не составные) атрибуты контрагента. Если поле не заполнено, отображается название атрибута без автора. Если атрибут заполнен, то отображается его название, значение и автор.

## Доп. атрибуты

Идентификатор во внешней системе • ВТ  
AL:13

Количество иждивенцев • admin\_performer  
1

Место работы • ВТ  
ООО Рога и Копыта

Дата начала работы • admin\_performer  
01.01.2011

Отделение • admin\_performer  
отделение

+ Резидентство

Автор

Резидентство

Неизвестно ▾

Подтвердить

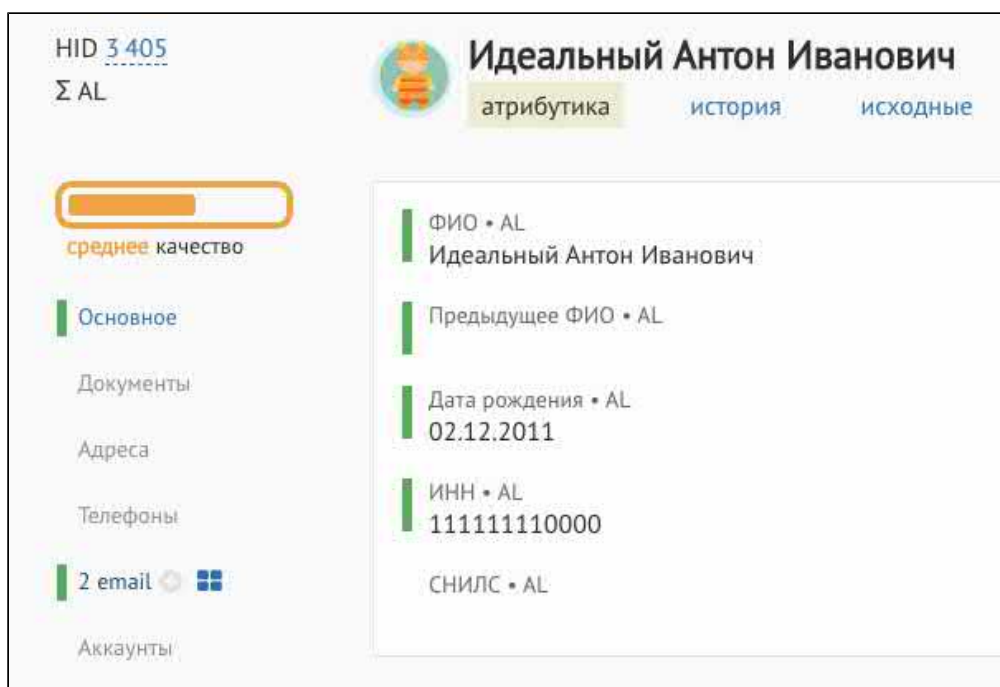
+ Полученное образование

# ПРОСМОТР КАРТОЧКИ. ТАБЛИЧНЫЙ РЕЖИМ ПРОСМОТРА АТТРИБУТОВ

## *Включение табличного режима отображения*

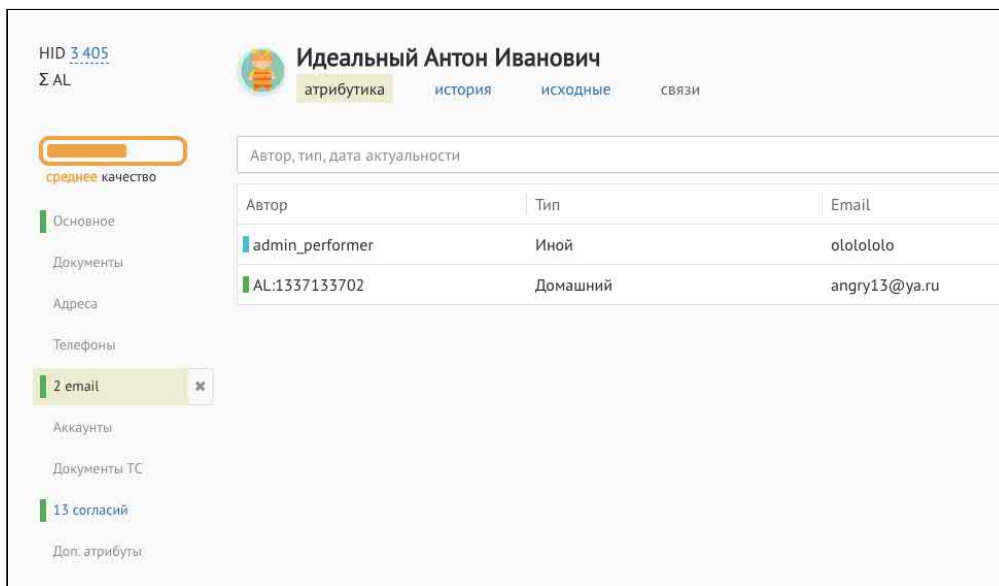
Если дата-стюарду требуется концентрированно разбирать телефоны, адреса или атрибуты другого типа, в веб-менеджере предусмотрен табличный режим работы.

Перейти в него можно, нажав на специальную иконку напротив нужного атрибута.



## *Управление карточкой в табличном режиме*

Находясь в табличном режиме можно видеть все атрибуты в виде таблицы, изменять ширину колонок таблицы, фильтровать в ней данные.



Если для табличного отображения доступны атрибуты другого типа, то переключиться на них можно, используя левое меню. При этом нет необходимости покидать табличный режим. При этом настройки таблицы сохраняются для каждого конкретного типа атрибутов по отдельности.

Переключаясь между контрагентами, отображение нужного табличного режима сохраняется.

Чтобы вернуться к стандартному отображению карточки контрагента, нужно нажать на «крестик» в левом меню.

## ***Управление карточкой в табличном режиме с клавиатуры***

Перемещение по элементам в табличном режиме возможно с помощью [горячих клавиш](#).



# ПРОСМОТР КАРТОЧКИ. СВЯЗИ

## Связи

Часто Единый клиент знает:

о договорах, которые заключил клиент;

взаимосвязях клиентов между собой (сотрудник-работодатель, страхователь-застрахованный, муж-жена).

Эта информация помогает дата-стюардам лучше разрешать потенциальные дубликаты, а маркетологам — строить аналитику по клиентской базе.

Дата-стюард переходит к связям по ссылке из карточки клиента:

**МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ** ID [2048](#) (07.07.2015)  
[история](#) [3 связи](#)

Общая информация | 2 документа | 6 адресов | 10 телефонов | 1 email | Аккаунты | Докум

ФИО • admin\_performer  
МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ

Дата рождения • performer  
26.11.1990

ИНН • performer  
743000228005

Список связей дата-стюард фильтрует по виду связи или связанному объекту:

**МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ** СВЯЗИ

3 связи: [застрахованный \(1\)](#) [созаемщик по отношению к \(1\)](#) [страхователь \(1\)](#)

Связаны: [физ. лица \(1\)](#) [договоры \(2\)](#)

Созаемщик по отношению к	РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ
Страхователь	Договор 130000025
Застрахованный	Договор 130000022

По клику на строку списка Единый клиент показывает подробности:

## МАРАКУЛИН АЛЕКСАНДР ВИКТОРОВИЧ СВЯЗИ

3 связи: [застрахованный \(1\)](#) [созаемщик по отношению к \(1\)](#) [страхователь \(1\)](#)

Связаны: [физ. лица \(1\)](#) [договоры \(2\)](#)

Созаемщик по отношению к

РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ



**РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ** ИД 2051

Источник: AL:11, AL:12, GM:2015-03-1554

ФИО  
РАГУЛИН АЛЕКСАНДР АЛЕКСАНДРОВИЧ

Дата рождения  
05.03.1987

ИНН  
753001686893

СНИЛС  
123-456-789 01

### Телефоны

Иной  
7 495 4886881 доб. 1

Для SMS-оповещений  
7 495 4887777 доб. 2

Телефон родственников/знак  
7 495 4886883 доб. 3

### Адреса

Иной  
309311, РОССИЯ, ОБЛ БЕЛГОРОДСКАЯ, Р-Н РАКИТЯН...

Регистрация  
309311, РОССИЯ, ОБЛ БЕЛГОРОДСКАЯ, Р-Н РАКИТЯН...

Прописка  
309311, РОССИЯ, ОБЛ БЕЛГОРОДСКАЯ, Р-Н РАКИТЯН...

### Документы

Загранпаспорт гражданина РФ  
0 7899877, ВОЗЛЕ ДОМА, 02.02.2002

# ПРОСМОТР КАРТОЧКИ. ИСТОРИЯ

## *История изменения клиентов*

Хороший клиент остается с компанией надолго. Пока идут годы, люди женятся и разводятся, переезжают, меняют номера телефонов, работу и документы, покупают новые продукты компании. Эти события попадают в информационные системы организации, а оттуда — в «Единый клиент».

С другой стороны, дата-стюарды выверяют контактные и паспортные данные, исправляют ошибки и опечатки, объединяют одинаковых клиентов. Эти события «Единый клиент» тоже запоминает.

Сводную историю изменений по человеку видно на карточке:

НID 2 203  
AL:2017071511071

Московский Юрий Олегович

атрибутика история исходные связи

среднее качество

Основное

1 документ

ФИО • AL  
Московский Юрий Олегович

Предыдущее ФИО • AL

«Единый клиент» увязывает все события в единую последовательность; показывает, что и когда изменилось:

НID 2 203  
AL:2017071511071

Московский Юрий Олегович

атрибутика история исходные связи

82 85 85 Alt

События

создание (2)

изменение (1)

удаление

слияние

разливание

восстановление

2017

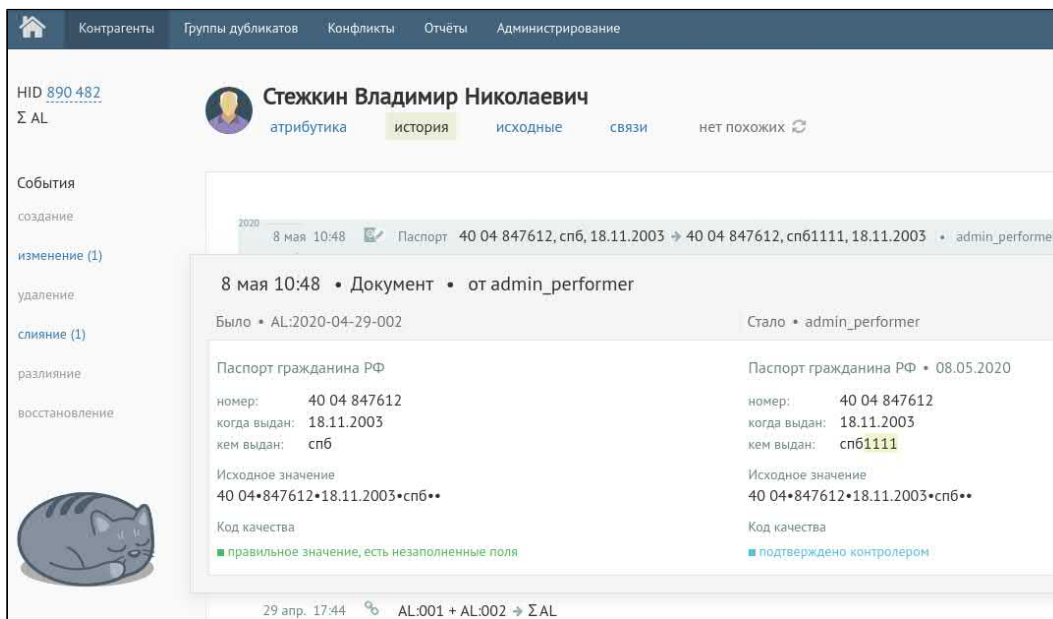
15 июля 13:34

ФИО Сельский Антон Олегович → Московский Юрий Олегович • AL:2017071511071

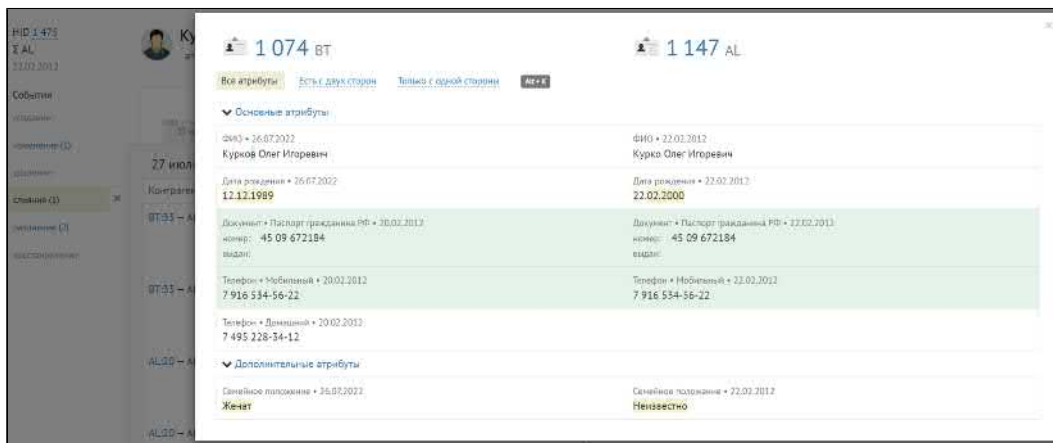
Паспорт 284158 • AL:2017071511071

AL:2017071511071

Если развернуть строку с событием, то показывает не только, что изменилось, но и кто автор изменения:



При клике на событии слияния открывается окно со сравнением дубликатов — исходных карточек именно на момент слияния:




## История изменения золотой карточки

В случаях, когда в золотую карточку подливается одна или несколько новых карточек, чьи атрибуты или реквизиты выигрывают, в истории ЗК для события слияния отображаются изменения в реквизитах и атрибутах золотой карточки.


При клике на событие слияния ЗК с новыми карточками, появляется выпадающее окно со сравнением золотой карточки — до и после слияния:


ID: 1\_026  
 Σ BT  
 01.02.2012


**Савченко Олег Игоревич**  
 атрибутыка история исходные связи нет похожих

**События**  
 создание: 22 сен. 08:30 Σ BT + BT.5 Σ BT - admin\_performer  
 изменение:  
 удаление:  
 слияние (2)  
 разделение:  
 восстановление:

12 сент. 08:30 • Слияние • подтверждено admin\_performer


**Было**  1 026 Σ BT

**Стало**  1 026 Σ BT

Все атрибуты:  Есть с двух сторон  Только с одной стороны  Актив

Основные атрибуты

ИМНО - 01.02.2012	ИМНО - 01.02.2012
Самец/кошка - Олег Игоревич	Самец/кошка - Олег Игоревич
Дата рождения - 01.02.2012	Дата рождения - 01.02.2012
12.05.2012	12.05.2012
ИМНО - 01.02.2012	ИМНО - 01.02.2012
Пустое значение (исходные: 419455136588)	441587423808
Документ - Паспорт гражданина РФ - 05.02.2012	Документ - Паспорт гражданина РФ - 05.02.2012
номер: 45 07 654543	номер: 45 07 654543
выдан:	выдан:
Адрес - Рязанский - 01.02.2012	Адрес - Рязанский - 01.02.2012

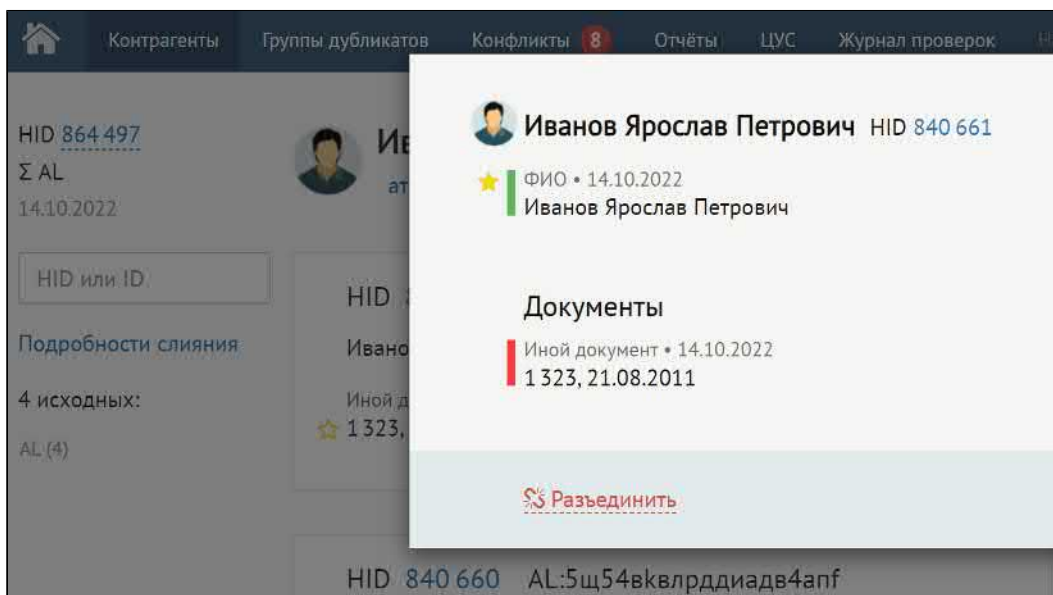


## ПРОСМОТР КАРТОЧКИ. ИСХОДНЫЕ

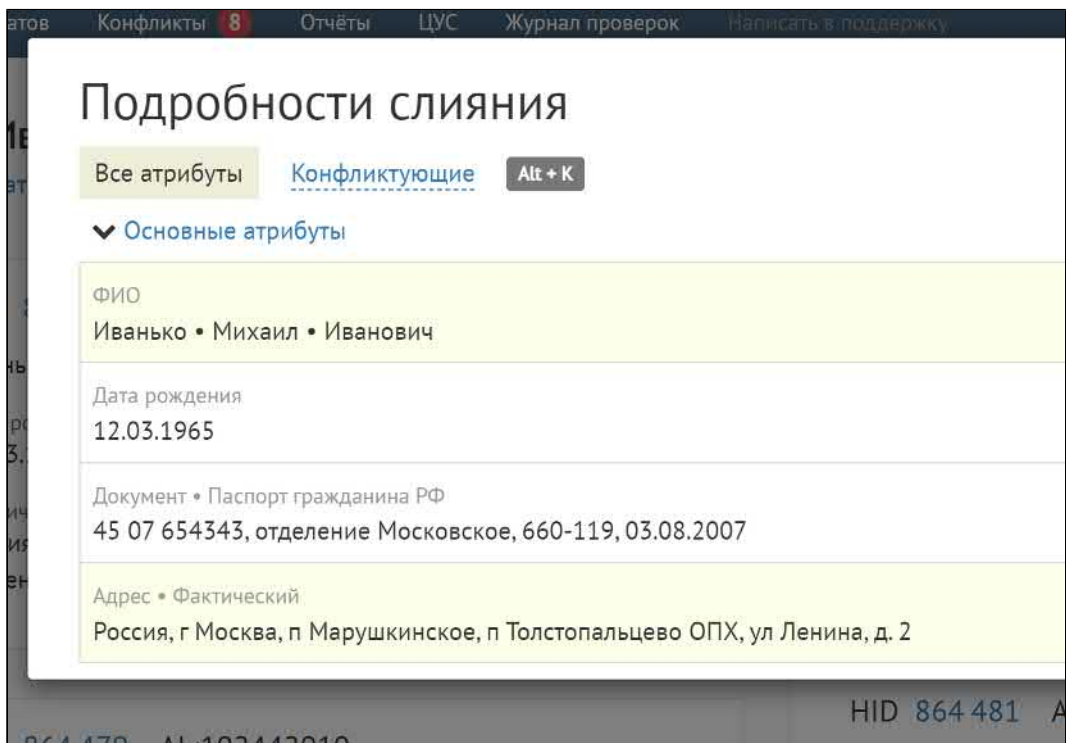
На вкладке можно увидеть все исходные карточки, входящие в золотую:



При необходимости влившую карточку можно вылить из итоговой — для этого нужно кликнуть по ней и нажать «Разъединить»:

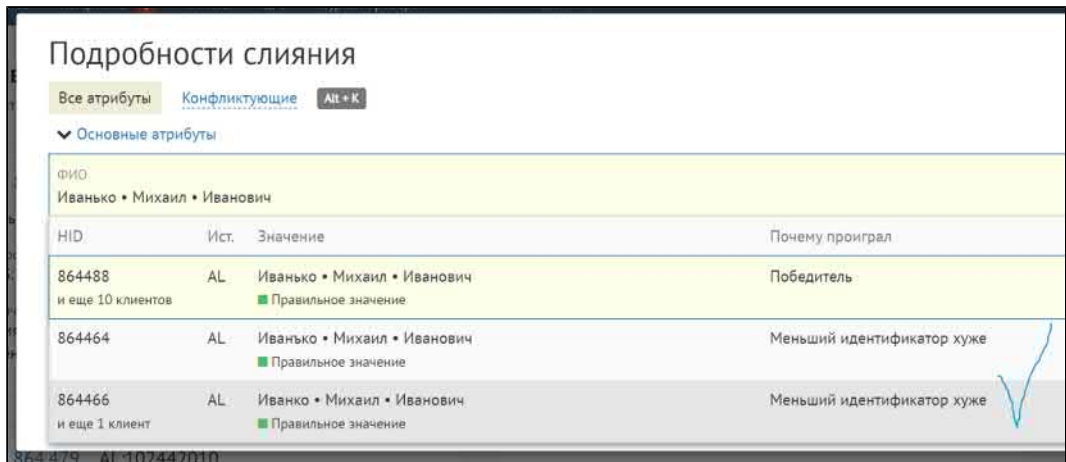


На вкладке «Исходные» также есть возможность посмотреть, почему золотая карточка выглядит так, а не иначе: прогноз её слияния из исходных сохраняется и доступен во всплывающем окне «Подробности слияния»:



По умолчанию показываются все реквизиты и атрибуты контрагента, желтым цветом выделены конфликтовавшие при слиянии.

Результат слияния можно поменять прямо на подробностях — нужно просто выбрать нужный вариант...



...и нажать «Переслать»:

### Подробности слияния

Переслать

Все атрибуты Конфликтующие Alt + K

▼ Основные атрибуты

ФИО  
Иванко • Михаил • Иванович

Дата рождения  
12.03.1965

Документ - Паспорт гражданина РФ  
45 07 654343, отделение Московское, 660-119, 03.08.2007

Адрес - Фактический  
Россия, г Москва, п Марушкинское, п Толстопальцево ОПХ, ул Ленина, д. 2

После ручного переслияния его факт также будет отражен на «Подробностях слияния»:

### Подробности слияния

Все атрибуты Конфликтующие Alt + K

▼ Основные атрибуты

ФИО  
Иванко • Михаил • Иванович

ID	Ист.	Значение	Почему проиграл
864466 и еще 1 клиент	AL	Иванко • Михаил • Иванович ■ Правильное значение	Выбрано вручную (admin_performer, 11.05.2023 17:11:17)
864438 и еще 10 клиентов	AL	Иванько • Михаил • Иванович ■ Правильное значение	Проиграл при ручном слиянии
864464	AL	Иванько • Михаил • Иванович ■ Правильное значение	Проиграл при ручном слиянии

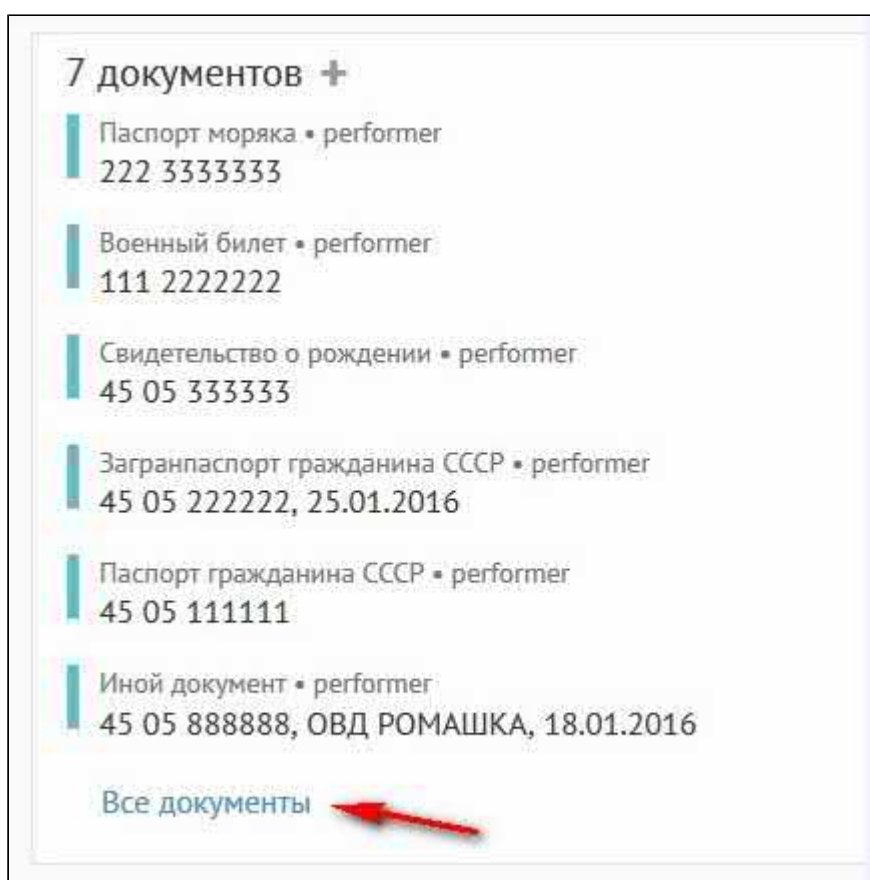


# ПРОСМОТР КАРТОЧКИ. ОТОБРАЖЕНИЕ АТТРИБУТОВ

## Отображение атрибутов

Показываем стандартизованное значение атрибута одной строкой, а если оно пустое, то исходное.

Адреса и документы показываем в один столбец (максимум 6 штук), телефоны, email и аккаунты — в три (максимум 11 штук). Остальное — по клику на «все атрибуты».



В режиме «Все атрибуты» отображаются только выбранные атрибуты клиента, общая информация и все остальные атрибуты скрыты.

Есть строка поиска. Справа расположены фильтры.

Чтобы вернуться к полной карточке клиента — нажмите крестик около атрибутов в верхней навигационной панели

ВИННИКОВ ОЛЕГ ИГОРЕВИЧ HID 28 338 (22.02.2012)

история исходные

Общая информация 7 документов 1 адрес

Документы ТС 12 согласий Дополнительные атрибуты

Автор, тип, код качества, дата актуальности

Автор	Тип	Документ	Дата акт.
performer	Паспорт моряка	222 3333333	14.01.2016
performer	Военный билет	111 2222222	14.01.2016
performer	Свидетельство о рождении	45 05 333333	14.01.2016
performer	Загранпаспорт гражданина СССР	45 05 222222, 25.01.2016	14.01.2016
performer	Паспорт гражданина СССР	45 05 111111	14.01.2016
performer	Иной документ	45 05 888888, ОВД РОМАШКА, 18.01.2016	14.01.2016
AL:20	Паспорт гражданина РФ	45 09 672184	22.02.2012

Автор: Все, performer

Тип: Все, Военный билет, Загранпаспорт гражданина СССР, Иной документ, Паспорт гражданина РФ, Паспорт гражданина СССР, Паспорт моряка, Свидетельство о рождении

Код качества: Все, Подтверждено контролером, Сомнительное значение

Нажмите крестик, чтобы закрыть вкладку документов и вернуться к общему списку

В строке поиска написано, по каким значениям можно искать

Справа в отдельной панели вынесены все фильтруемые параметры и их значения, можно фильтровать и комбинировать фильтры

# ПРОСМОТР КАРТОЧКИ. ОТКЛОНЕННЫЕ ДУБЛИКАТЫ

## Отклоненные дубликаты

Если «Единый клиент» находил похожие карточки, но дата-стюард сказал, что они «Не дубликаты» — такие пары больше не появятся в разделе «Группы дубликатов». Найти их можно в разделе «Отклоненные дубликаты» на карточке контрагента.

The screenshot shows a user profile for 'Исторический Иннокентий' (Historical Innocent). The profile includes a profile picture, name, and several tabs: 'атрибутика' (attributes), 'история' (history), 'исходные' (originals), 'связи' (connections), and 'отклоненные дубли' (rejected duplicates). The 'отклоненные дубли' tab is active, showing a list of rejected duplicates. A search bar labeled 'HID или ID' is present. The first entry in the list is for 'Исторический Иннокентий Варламович' with HID 2 754 and AL:2018062242424... and a birth date of 14.01.1980. On the left side, there are sections for 'Дубликаты:' (Duplicates: 1), 'Системы:' (Systems: 1), and 'Сценарии:' (Scenarios: 1).

Кроме того, в этом разделе показаны и ранее разъединенные карточки. В левой части есть удобный фильтр по типу пары, исходным системам и правилам.

Нажав на карточку, можно увидеть подробности: сравнение, информацию об истории статуса, а также возможность изменить решение.

2754 AL,BT,GM      2775 BT

▼ Основные атрибуты

ФИО (присвоенное значение) • 22.06.2018 Исторический Иннокентий	ФИО • 24.02.2017 Исторический Иннокентий
актуальное значение Исторический Иннокентий Варламович	
Дата рождения • 21.06.2018 14.01.1980	Дата рождения • 24.02.2017 15.01.1980
ИНН • 21.06.2018 374979431791	
СНИЛС • 21.06.2018 335-728-543 95	
Паспорт гражданина РФ • 20.06.2018 73 01 223697, ОВД района Зюзино, 01.01.2016	
Домашний • 23.02.2016 Россия, Москва, Черемушкинская, 28	Иной • 22.02.2016 Россия, Москва, Черемушкинская, 25
Иной • 10.10.2012 129090, Россия, г Москва, пр-кт Мира, д. 42, кв. 34	
Мобильный • 22.04.2015 7 916 226-30-44	Мобильный • 22.02.2012 7 916 226-30-44
Иной • 10.10.2012 7 495 234-64-53	
Мобильный • 22.06.2018 7 916 226-30-42	

Дубликат был найден по правилу: ФИО 70-100 + телефон  
Опсложен admin\_performer 22 июня 2018 года

[Принять дублиат](#)

Версия: 3.8.8-SNAPSHOT (5462794875eb) Ядро: 1.8.8-SNAPSHOT (655b4151a6ed) ФАКТОР (оса) 0.1.80.0-8.14 (97979)

## ПРОСМОТР КАРТОЧКИ. ОБНОВЛЕНИЕ ИНФОРМАЦИИ О ДУБЛИКАТАХ

### ⚠ Обратите внимание

Опция «Обновление информации о дубликатах» осуществляет поиск дубликатов только между карточками клиентов одного типа (например, ФЛ-ФЛ, ЮЛ-ЮЛ) и при условии, что для данного типа контрагентов есть правила дедупликации.

Кросс-сверки (такие как проверка по ЕГРЮЛ, проверка по ЧС, кросс-сверка ФЛ-ЮЛ и др.) выполнены не будут, они осуществляются только в рамках специальных задач.

Кнопка обновления информации о дубликатах находится рядом с количеством похожих.

Константинопольский Константин Константинович

атрибутика история исходные связи **2 похожих**

ФИО • AL  
Константинопольский Константин Константинович

Исходные идентификаторы  
AL:2017042313091  
AL:2017042313091

Предыдущее ФИО • AL  
Валерьянов Валерьян Владимирович

Дата рождения • AL  
13.11.1977

ИНН • AL  
201577814430

СНИЛС • AL

2 документа +


Паспорт гражданина РФ • admin\_performer  
45 08 565785

Паспорт гражданина СССР • AL  
4508565785

Если у карточки нет похожих, то кнопка показывается рядом с серой надписью «нет похожих».

Если после нажатия кнопки количество дубликатов изменилось, появляется зеленая точка.

Если нет — ничего не меняется.

 **Константинопольский Константин Константинович**

атрибутика   история   исходные   связи   [2 похожих](#)

ФИО • AL  
Константинопольский Константин Константинович

Предыдущее ФИО • AL  
Валерьянов Валерьян Владимирович

Дата рождения • AL  
13.11.1977

ИНН • AL  
201577814430

СНИЛС • AL

Исходные идентификаторы  
AL:2017042313091  
AL:2017042313091

**2 документа +**

Паспорт гражданина РФ • admin\_performer  
45 08 565785

Паспорт гражданина СССР • AL  
4508565785

Кнопка обновления показывается всегда для всех актуальных карточек.

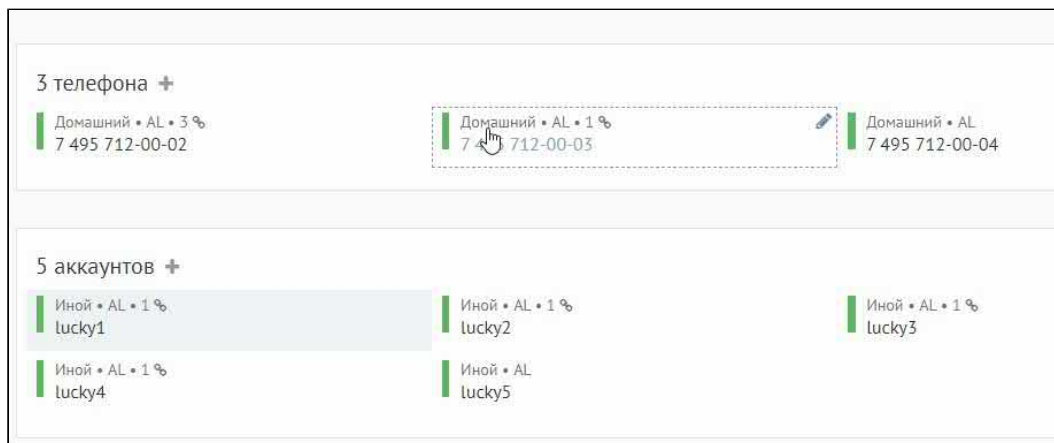
# ПРОСМОТР КАРТОЧКИ. СВЯЗАННЫЕ АТТРИБУТЫ

## Отображение связанных атрибутов на карточке

Рядом со связанным атрибутом в свернутом виде отображается пометка о количестве связей.

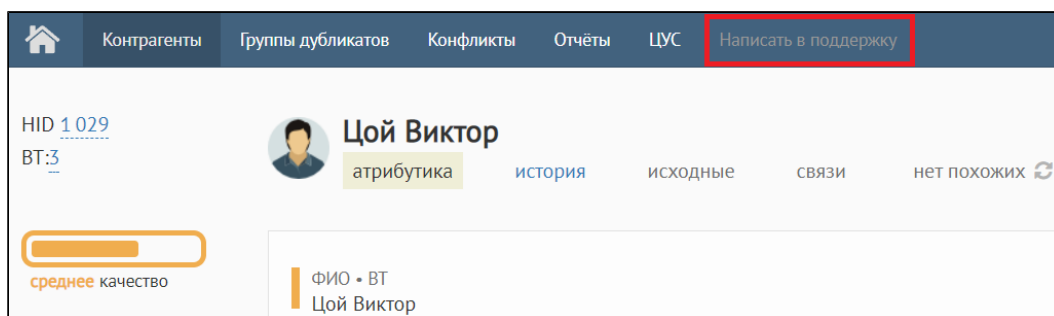


При наведении мышки на связанный атрибут, либо при раскрытии его формы, подсвечиваются остальные связанные с ним атрибуты:

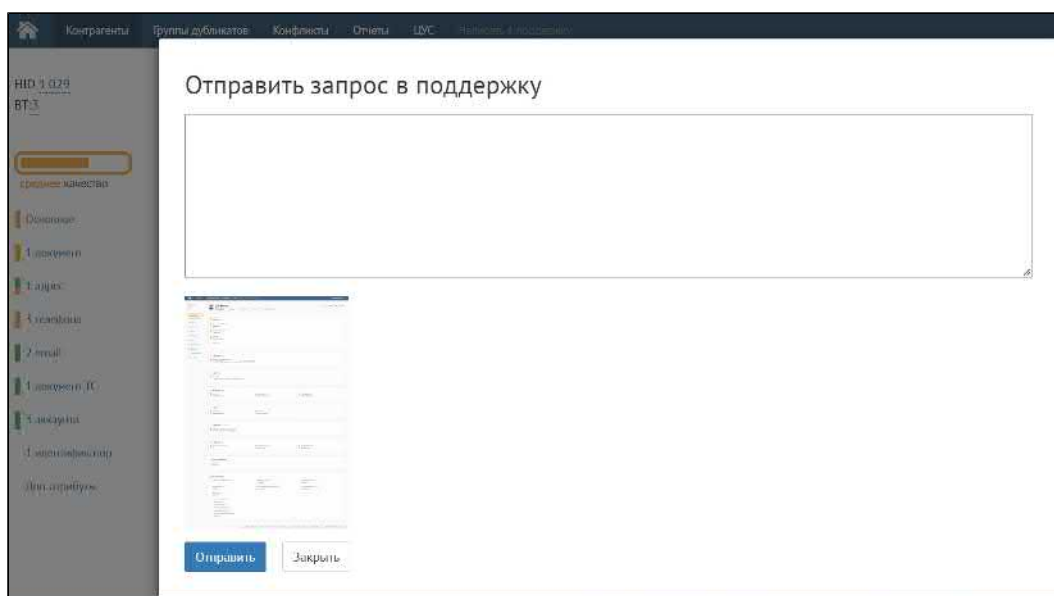


# ПРОСМОТР КАРТОЧКИ. ОБРАЩЕНИЕ В ПОДДЕРЖКУ

На панели с основными вкладками есть кнопка «Написать в поддержку».



При клике по ней открывается форма для комментария:



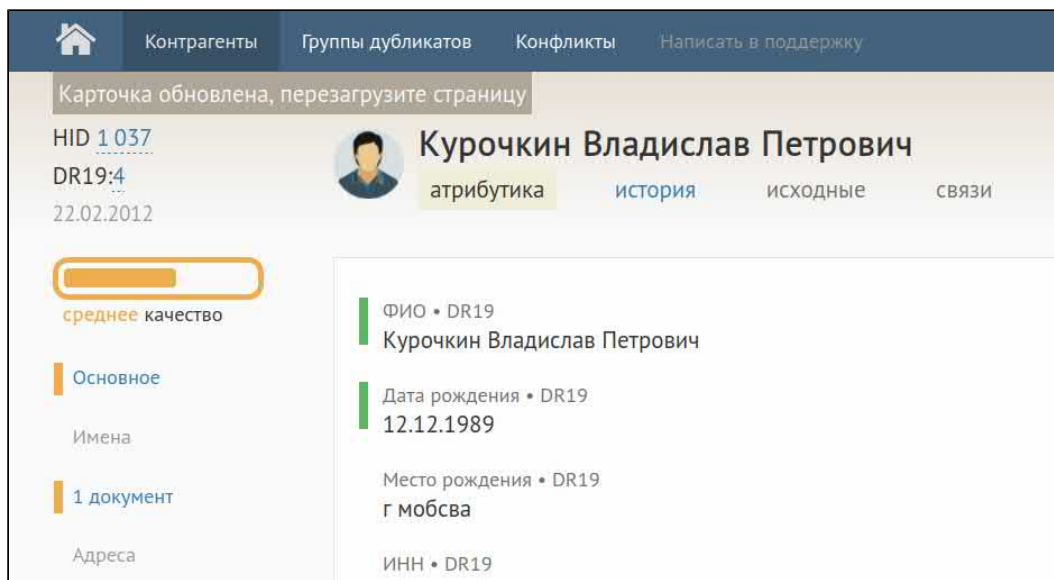
По нажатию на «Отправить» автоматически уйдет письмо в поддержку с комментарием, диагностикой по контрагенту, с карточки которого осуществлялась отправка, и скриншотом экрана.

**⚠** Кнопка «Написать в поддержку» доступна только пользователям с ролями ROLE\_ADMINISTRATOR и ROLE\_PERFORMER.



## ПРОСМОТР КАРТОЧКИ. УВЕДОМЛЕНИЯ ОБ ОБНОВЛЕНИИ КАРТОЧКИ

Если во время просмотра или редактирования карточки контрагента дата-стюардом карточка получила обновление, пришедшее, например, по SOAP-интерфейсу, то отобразится уведомление — «Карточка обновлена, перезагрузите страницу»:



The screenshot shows a user interface for a CRM card. At the top, a dark blue navigation bar contains a home icon and menu items: 'Контрагенты', 'Группы дубликатов', 'Конфликты', and 'Написать в поддержку'. Below this, a light blue notification banner reads 'Карточка обновлена, перезагрузите страницу'. The card header includes the ID 'NID 1 037', the code 'DR19:4', and the date '22.02.2012'. The name 'Куручкин Владислав Петрович' is displayed with a profile picture icon. Below the name are tabs: 'атрибутика', 'история', 'исходные', and 'связи'. On the left side, there is a progress indicator for 'среднее качество' (average quality) and a list of sections: 'Основное', 'Имена', '1 документ', and 'Адреса'. The main content area on the right lists attributes: 'ФИО • DR19' (Куручкин Владислав Петрович), 'Дата рождения • DR19' (12.12.1989), 'Место рождения • DR19' (г мобсва), and 'ИНН • DR19'.

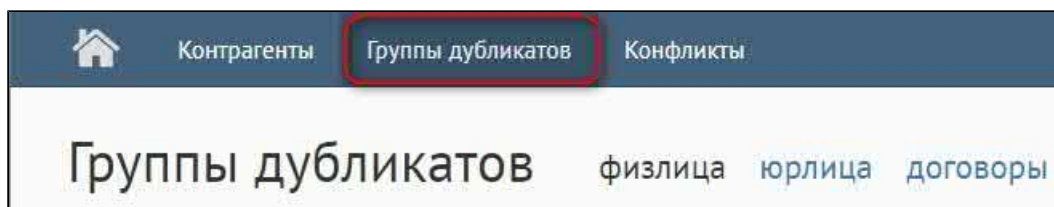
# РАБОТА СО СПИСКОМ ГРУПП ДУБЛИКАТОВ

- Просмотр списка
  - Работать со списком групп дублей
    - Обычный поиск
    - Расширенный фильтр
    - Синтаксис для поиска
- Переход на карточку группы

## ПРОСМОТР СПИСКА

Список дубликатов содержит информацию по найденным системой дубликатам, а также набор инструментов, позволяющий найти необходимую пару дублей.

Чтобы открыть список, нужно перейти на вкладку «Группы дубликатов»



### *Работать со списком групп дублей*

В Менеджере данных дата-стюард находит интересующие его дубликаты и принимает решение о слиянии.



### *Обычный поиск*

Доступные инструменты для управления списком:

- Поисковый запрос, может быть 2-х видов:
  - Обычный, поиск значения осуществляется по полям *Описание* и *ИД контрагента*.
  - Расширенный, поиск по реквизитам контрагента или группам дубликатов. При данном варианте полностью слитые группы не попадают в результирующий список.
- Фильтр по типу клиента.
- Фильтр по статусу обработки группы:

*Не обработаны* — отобразить все группы, в которых есть хоть один необработанный дубликат.

*Отложены* — отобразить только обработанные группы, в которых есть хоть один отложенный дубликат.

*Обработаны* — отобразить только полностью обработанные группы, по всем дубликатам которых принято решение *Дубликат / Не дубликат*.

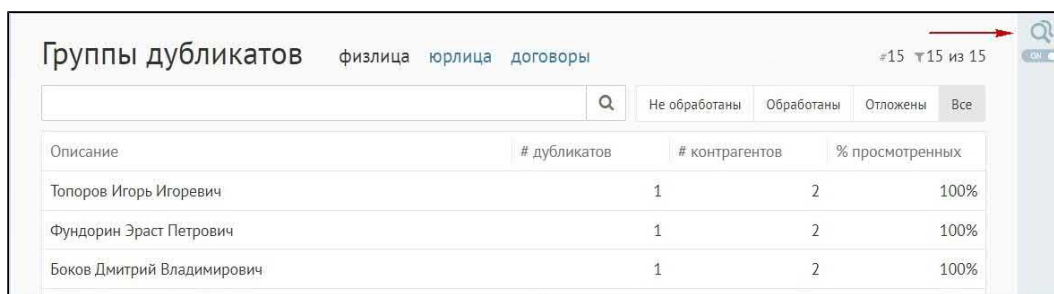
*Все* — отобразить все группы.

- Сортировка списка групп по всем полям

**i** Сортировка списка отключается, когда количество групп дубликатов превышает 1000.

### **Расширенный фильтр**

Находится в панели справа, как на списке контрагентов:



Описание	# дубликатов	# контрагентов	% просмотренных
Топоров Игорь Игоревич	1	2	100%
Фундорин Эраст Петрович	1	2	100%
Боков Дмитрий Владимирович	1	2	100%

Этот режим позволяет добавить дополнительные условия для фильтрации групп:

### Группы дубликатов

физлица юрлица договоры

Имя:  Поиск:  Не обработано:

Описание	# дубликатов	# конт.
Тихонравов Михаил Клавдиевич	1	
Николаенко Ирина Ивановна	1	
Злой Иван Олегович	1	
Шумная Полина Олеговна	1	
Прохладный Семен Олегович	1	
Крузенштерн Ирина Ивановка	1	
Тукай Габдула Мухамедгари	1	
Единорог Радугович Клиент	1	
Дедеде DDDDD И	3	
Петров Иван А	3	
Барабашки Анатолий Анатольевич	13	
Овес Полезен	28	

#### Расширенный фильтр

Описание:

Есть отклоненные

Реестр: (Не задано)

Сегмент: (Не задано)

НID контрагента:

Сценарий: (Не задано)

Дата создания дубликата: (Не задано)

Исходная система: (Не задано) - (Не задано)

только одиночные  только одиночные

Коэф. схожести: от  до

Количество контрагентов: от  до

- *Описание*
- *Наличие отклоненных дублей*
- *Реестр*
- *Сегмент*
- *НID контрагента*
- *Сценарий*
- *Дата создания дубликата*, поиск осуществляется только среди необработанных пар дубликатов
- *Исходная система*
- *Коэф. схожести*
- *Количество контрагентов*

Подробнее об использовании расширенного фильтра.

### **Синтаксис для поиска**

При вводе значений в поисковые поля (*Поисковый запрос*, *Описание*) можно указывать подстановочные символы:

- **\_** — один произвольный символ;
- **%** — произвольная последовательность символов (ноль или более символов).

## ПЕРЕХОД НА КАРТОЧКУ ГРУППЫ

Переход на **карточку группы** осуществляется:

- нажатием левой кнопки мыши по записи в списке;
- установкой курсора на нужную запись и нажатием клавиши **Enter**.

Вся фильтрация, кроме «Количества контрагентов» с экрана групп дубликатов доступна для применения и на карточке группы дубликатов.

**ДИВИЦКИЙ ДЕН О К ШАРИФБОВЕВИЧ** нет подтвержденных дубликатов

5 контрагентов, [5 дубликатов](#)

Показана вся группа Применить фильтры: сценарий: ФИО 90-100 + адрес 90-100 + ДР ×

Контрагенты	Коэф.	Сцен.	Правило
27 383 192 – 55 307 340	КП=89	31 :	ФИО 90-100 + документ
27 383 192 – 121 676 460	КП=89	31 :	ФИО 90-100 + документ
55 307 340 – 121 676 460	КП=89	31 :	ФИО 90-100 + документ
27 383 192 – 121 508 827	КП=88	32 :	ФИО 90-100 + ИНН
55 307 340 – 121 680 946	КП=87	34 :	ФИО 90-100 + адрес 90-100 + ДР

#1 ▾ 5 из 5 КП=89 31 : ФИО 90-100 + документ

Дубликат Не дубликат Отложить ▾ Следующая пара →

Alt + Y Alt + N Alt + L

Фильтрацию можно применить и отменить.

Перейти на следующую (предыдущую) группу можно:

- по клику на стрелочку вправо (влево) в верхнем правом углу экрана,
- по сочетанию Alt + ? (Alt + ?).

При переходе между группами фильтрация сохраняется.

# ОТОБРАЖЕНИЕ ДУБЛИКАТОВ ЮЛ И ЕГРЮЛ

- Отображение групп дублей ЮЛ - ЮЛ
- Отображение групп дублей ЮЛ - ЕГРЮЛ
- При выборе двух фильтров будет работать фильтр по ЕГРЮЛ

## Отображение групп дублей ЮЛ - ЮЛ

Чтобы разобрать дубликаты между двумя ЮЛ, выберите фильтр в интерфейсе "Проверка дубликатов ЮЛ":

The screenshot shows a web interface for checking duplicates. The main area is titled "Группы дубликатов" and has tabs for "физлица", "юрлица", and "договоры". Below the tabs is a search bar and a "Не обработано" button. A table displays the results of a duplicate check for "ЮЛ". The table has columns for "Описание", "# дубликатов", and "# конт". The data rows are as follows:

Описание	# дубликатов	# конт
ИП Мокрушина Елена Аркадьевна	1	
ИП Агузаров Марат Владимирович	1	
ИП Логачева Тамара Николаевна	1	
ИП Кириенко Вячеслав Николаевич	1	
ИП Уткин Артем Александрович	1	
ИП Печерская Алла Александровна	1	
ИП Варвашян Сируш Завеновна	1	
ИП Калачева Юлия Александровна	1	

On the right side, an "Расширенный фильтр" (Advanced filter) panel is open, showing various filter options:

- Описание: ФИО, название компании
- Есть отклоненные
- Проверка дубликатов ЮЛ
- Проверка по ЕГРЮЛ
- Реестр: Пока реестров нет
- Сегмент: (Не задано)
- ИД контрагента: Можно несколько через запятую
- Сценарий: (Не задано)

Так будут показаны все группы, которые содержат дубликаты ЮЛ друг к другу среди других.

## Отображение групп дублей ЮЛ - ЕГРЮЛ

Чтобы разобрать дубликаты между ЮЛ и ЕГРЮЛ, выберите фильтр в интерфейсе "Проверка по ЕГРЮЛ":

Группы дубликатов [физлица](#) [юрлица](#) [договоры](#)

Не обработано

проверка по ЕГРЮЛ x

Описание	# дубликатов	# конт
ООО РИА ДЕЛОВОЙ МИР-ИДМ	1	
ООО УНИТРЕЙД	1	
Учреждение УЧРЕЖДЕНИЕ СОЦИАЛЬНОЙ И ПРОФЕССИ...	1	
Общественное учреждение УЧРЕЖДЕНИЕ СОЦИАЛЬНО...	1	
ООО ФЛЕР ЭКСКЛЮЗИВ	1	
ООО ПЕТРОМИР	2	
ООО ПРОМЭНЕРГОСЕРВИС	1	
ООО ВИКТОРИ	1	
Общественный фонд ФОНД ИЗОБРЕТАТЕЛЬ И РАЦИОНА...	1	

Расширенный фильтр ON

Описание

ФИО, название компании

Есть отклоненные

Проверка дубликатов ЮЛ

Проверка по ЕГРЮЛ

Реестр

Пока реестров нет

Сегмент

(Не задано)

НID контрагента

Можно несколько через запятую

Сценарий

(Не задано)

Так будут показаны все группы, в которых есть хотя бы одна пара дубликатов ЮЛ к ЕГРЮЛ.

### При выборе двух фильтров будет работать фильтр по ЕГРЮЛ

Группы дубликатов [физлица](#) [юрлица](#) [договоры](#)

Не обработано

проверка дубликатов ЮЛ, проверка по ЕГРЮЛ x

Описание	# дубликатов	# конт
ООО РИА ДЕЛОВОЙ МИР-ИДМ	1	
ООО УНИТРЕЙД	1	
Учреждение УЧРЕЖДЕНИЕ СОЦИАЛЬНОЙ И ПРОФЕССИ...	1	
Общественное учреждение УЧРЕЖДЕНИЕ СОЦИАЛЬНО...	1	
ООО ФЛЕР ЭКСКЛЮЗИВ	1	
ООО ПЕТРОМИР	2	
ООО ПРОМЭНЕРГОСЕРВИС	1	
ООО ВИКТОРИ	1	
Общественный фонд ФОНД ИЗОБРЕТАТЕЛЬ И РАЦИОНА...	1	

Расширенный фильтр ON

Описание

ФИО, название компании

Есть отклоненные

Проверка дубликатов ЮЛ

Проверка по ЕГРЮЛ

Реестр

Пока реестров нет

Сегмент

(Не задано)

НID контрагента

Можно несколько через запятую

Сценарий

(Не задано)

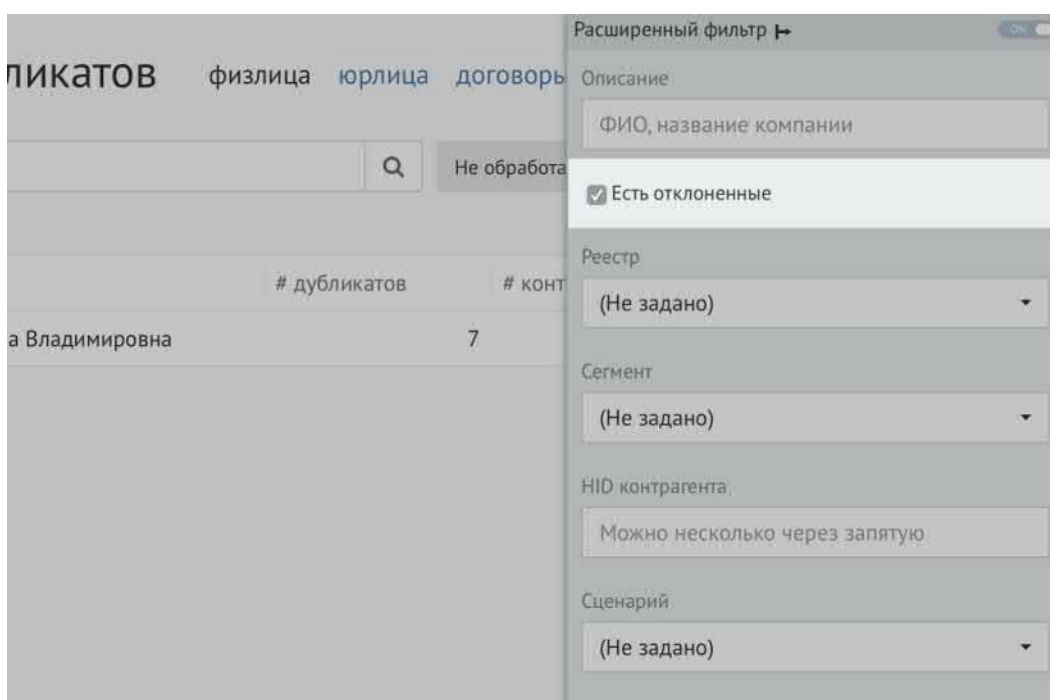


# ОТОБРАТЬ ГРУППЫ С ОТКЛОНЕННЫМИ ДУБЛИКАТАМИ

## Фильтрация групп с отклоненными дубликатами

«Единый клиент» проверяет перед слиянием все пары и, если в группе присутствуют отклоненные, разлитые или антидубликаты, то не проводит автоматическое слияние, а оставляет группу на разбор дата-стюарду.

Для отбора таких групп, внутри которых есть конфликты, нужно отметить чек-бокс «Есть отклоненные»:



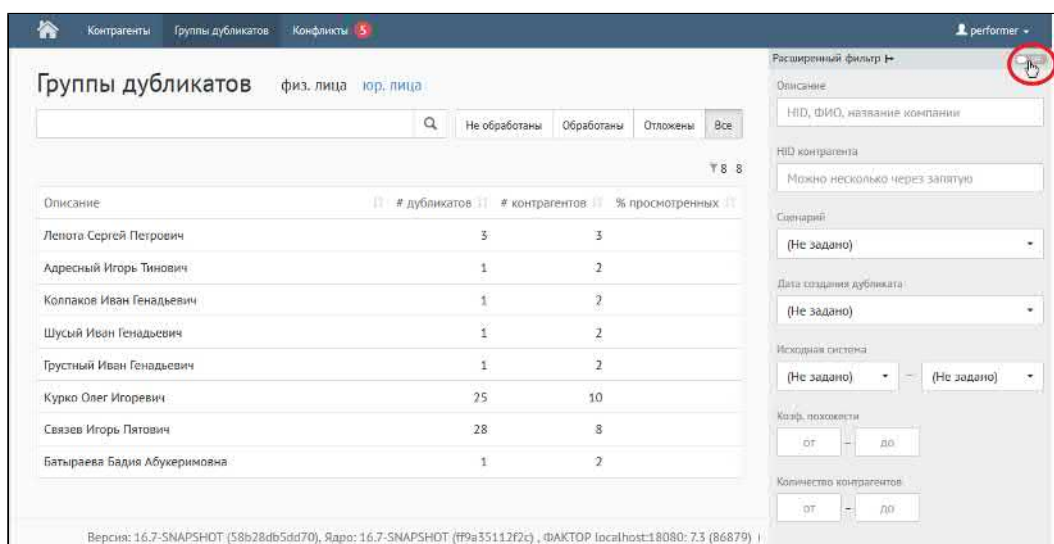
Будут показаны все группы, где есть отклоненные (в том числе автоматически отклоненные и ранее разлитые) дубликаты.

# ОТОБРАТЬ НЕДАВНО СОЗДАННЫЕ ДУБЛИ

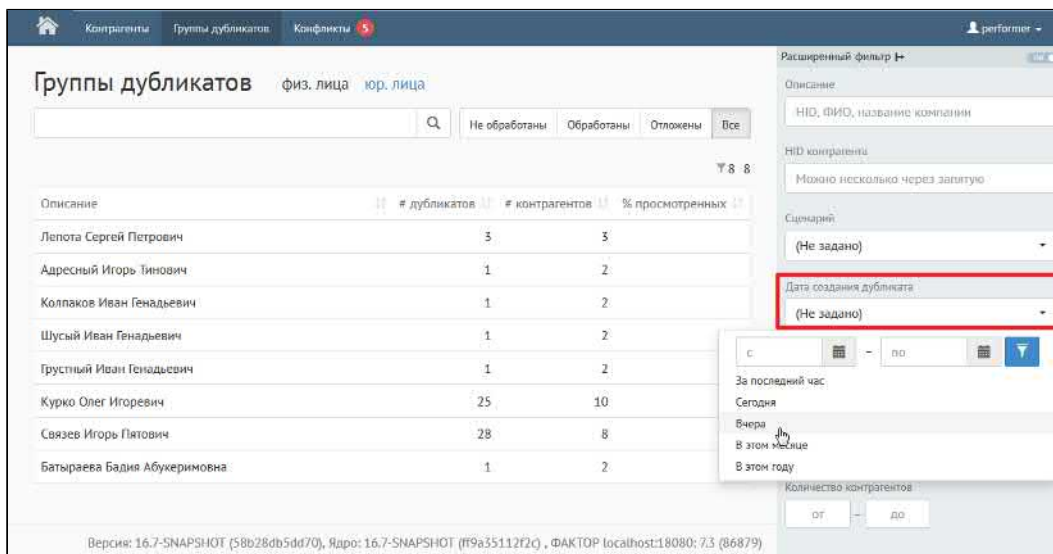
Открыть список дубликатов и развернуть панель фильтров



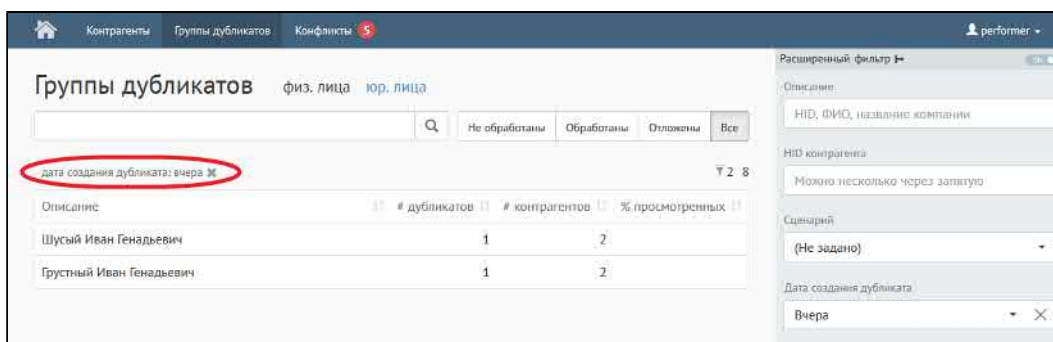
Включить фильтрацию.



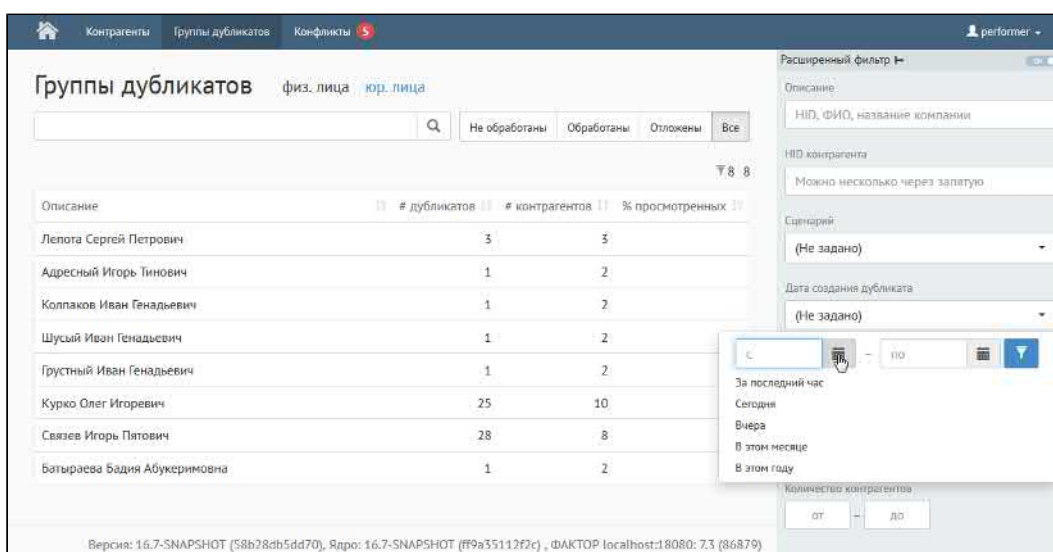
Открыть фильтр «Дата дубликата» и выбрать «Вчера».



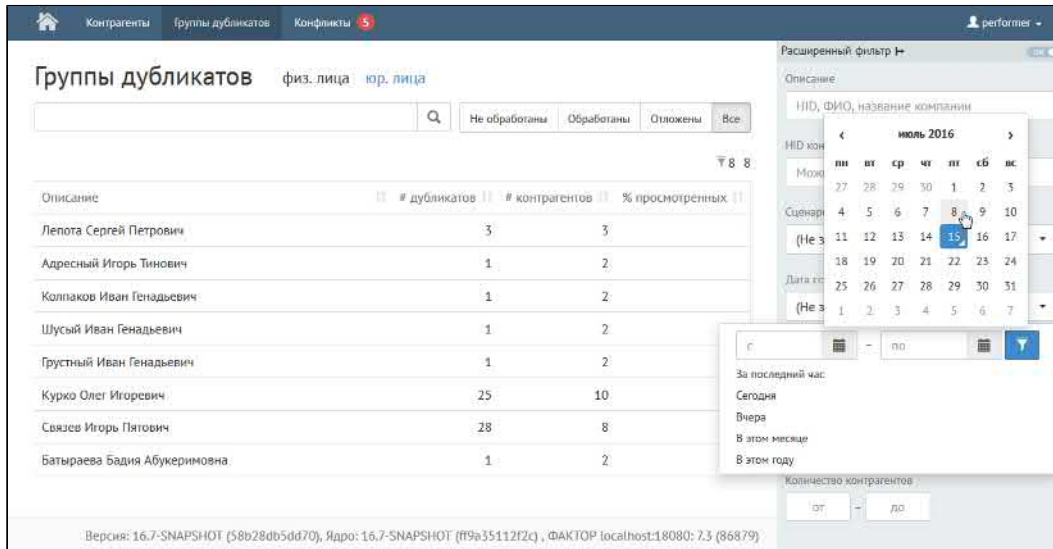
Фильтр будет сразу применен.



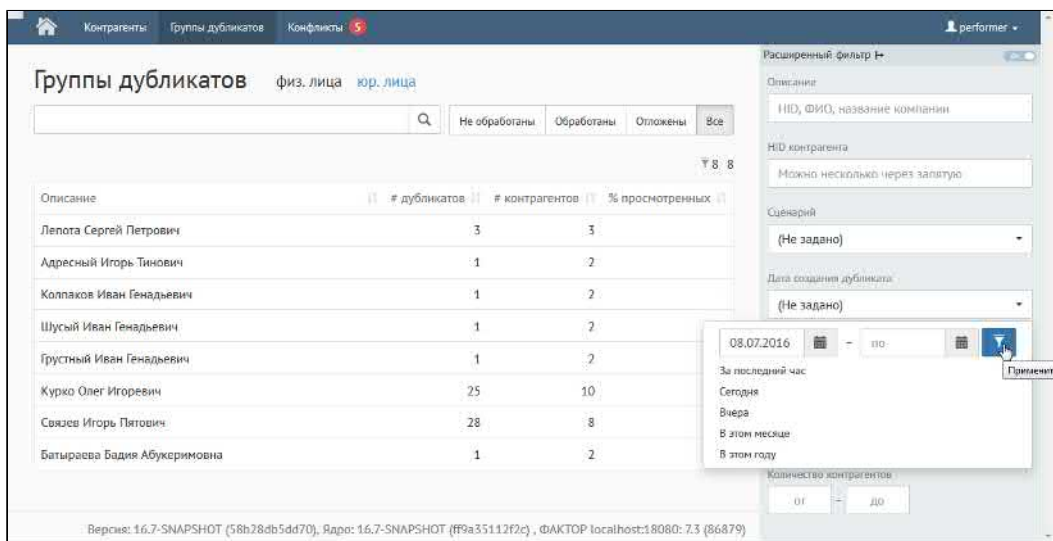
Если нужно отфильтровать дубли за неделю или любой другой промежуток, то открыть календарь.



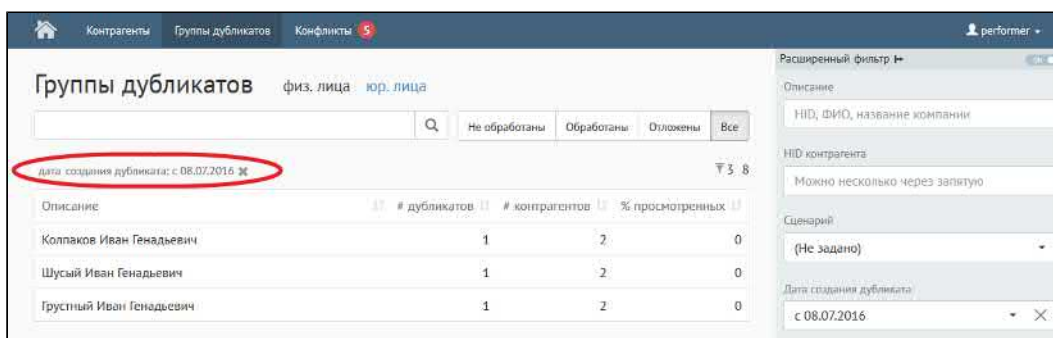
Выбрать нужную дату.



Нажать «Применить фильтр».

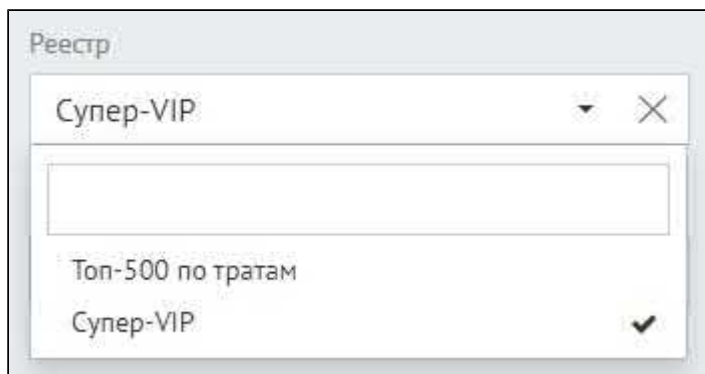


Будут показаны дубли, созданные в выбранном промежутке дат.



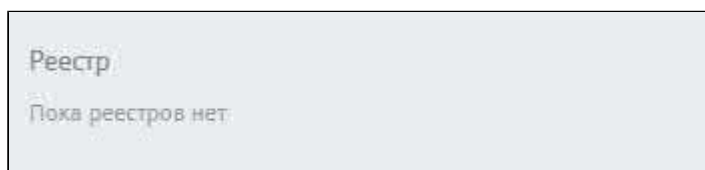
## ПОИСК ДУБЛИКАТОВ ПО ВХОЖДЕНИЮ В РЕЕСТР

Фильтрация групп дубликатов по внешнему реестру доступна в блоке «Реестр» расширенного фильтра:

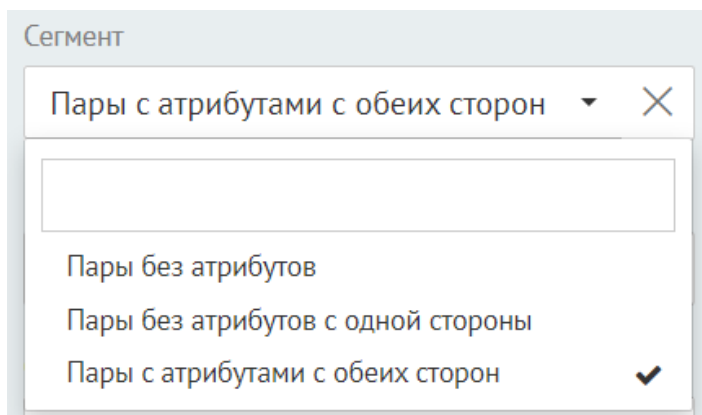


Чтобы список реестров появился в веб-менеджере данных, Администратор предварительно загружает их в специальную таблицу и запускает задачу [актуализации реестров](#).

При отсутствии реестров в Едином клиенте фильтрация недоступна:



## ПОИСК ДУБЛИКАТОВ ПО ВХОЖДЕНИЮ В СЕГМЕНТ



Фильтрация групп дубликатов по сегменту доступна в блоке «Сегмент» расширенного фильтра:

По умолчанию доступны сегменты:

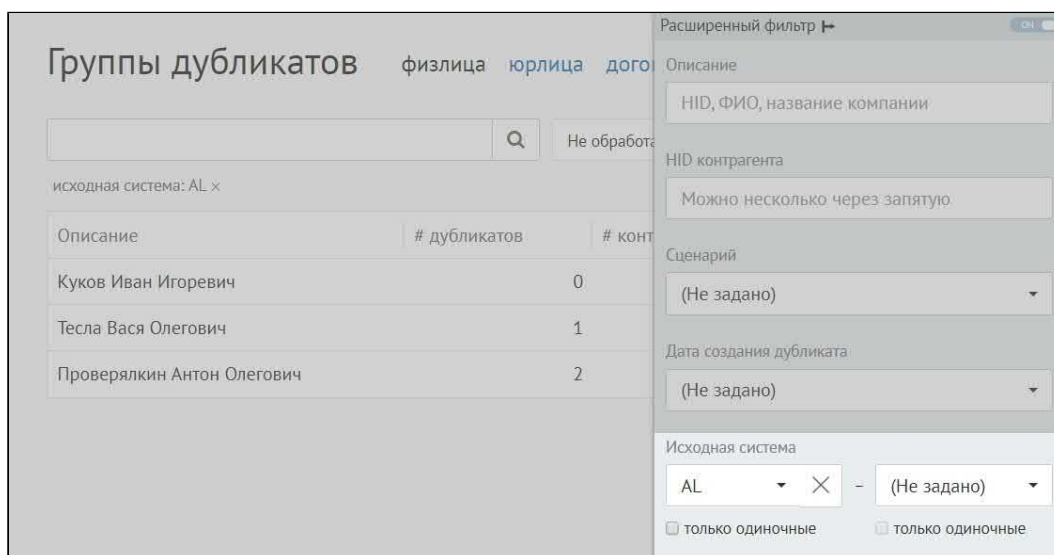
- Пары без атрибутов (у обоих КА отсутствуют любые атрибуты);
- Пары без атрибутов с одной стороны (у одного КА из пары есть минимум один любой атрибут);
- Пары с атрибутами со обеих сторон (у обоих КА из пары есть минимум один любой атрибут).

# ПОИСК ДУБЛИКАТОВ ПО ИСХОДНОЙ СИСТЕМЕ

- Фильтрация по исходной системе с одной стороны
- Фильтрация по дублиям одиночных карточек из исходной системы
- Фильтрация внутрисистемных дублей
- Фильтрация внутрисистемных дублей для одиночной карточки
- Фильтрация внутрисистемных дублей между одиночными карточками
- Фильтрация межсистемных дублей
- Фильтрация межсистемных дублей среди одиночных карточек из нескольких систем

## Фильтрация по исходной системе с одной стороны

Чтобы разобрать дубликаты из конкретной исходной системы, выберите ее в фильтре:



Так будут показаны все группы, в которых есть пара дублей с участием AL, например:

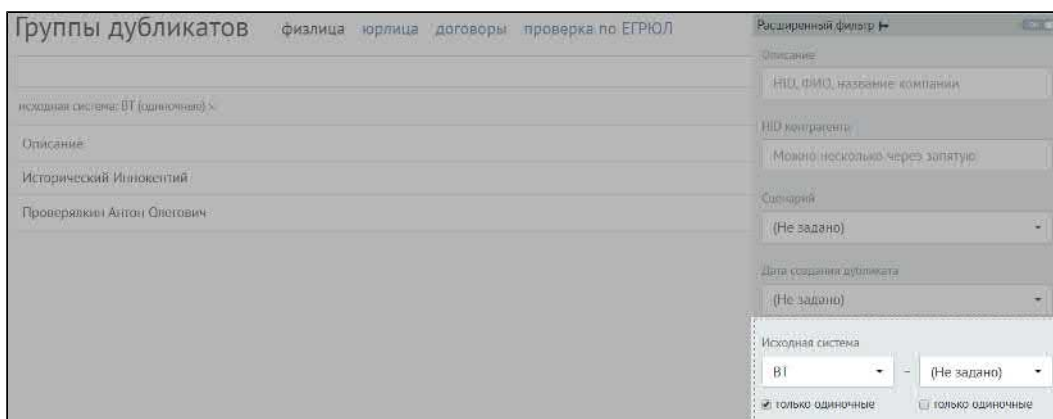
"AL:123" — "BT:456" — тут AL:123 — это одиночная карточка.

"AL:777, BT:888" — "GM:99, GM:222" — тут карточка из системы AL входит в состав золотой карточки "AL:777, BT:888".

Чтобы перейти на страницу группы, нужно выбрать ее из списка. При этом в группе также могут быть и пары дублей из других систем.

## Фильтрация по дублиям одиночных карточек из исходной системы

Чтобы оставить только те группы дубликатов, где дублем выступает одиночная карточка из исходной системы (например, BT), выберите ее в фильтре и поставьте галочку "только одиночные":

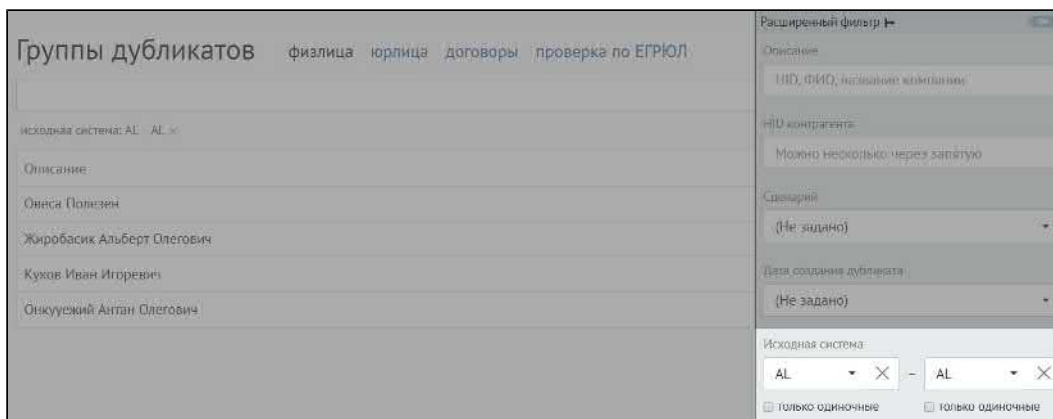


В результат попадут группы дубликатов, где есть пара дублей с участием одиночной карточки из исходной системы BT:

"AL:123" — "BT:456" найдется по этому условию, т.к. здесь BT:456 — одиночная карточка, а группа с парой дублей "AL:777, BT:888" — "GM:99, GM:222" не попадет в результат, т.к. карточка из BT здесь входит в состав золотой записи.

## Фильтрация внутрисистемных дублей

Чтобы найти группы дубликатов, где дублями являются карточки из одной и той же исходной системы, выберите ее в фильтре для обеих сторон дубля:



Останутся группы дубликатов, в которых есть пара дублей с участием карточек из AL с обеих сторон, например:

"AL:123" — "AL:456" найдется по этому условию, т.к. здесь обе одиночные карточки из исходной системы AL,

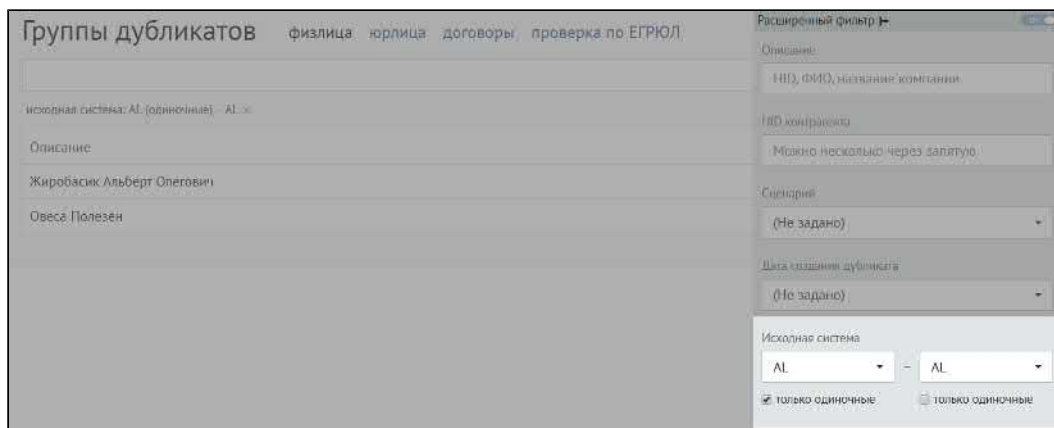
"AL:777, BT:888" — "AL:99, GM:222" — тут карточка из системы AL входит в состав золотых карточек "AL:777, BT:888" и "AL:99, GM:222",

"BT:23, AL:17" — "AL:98" найдется по этому условию, т.к. здесь, с одной стороны, "AL:123" — одиночная карточка, с другой стороны — "AL:17" входит в состав золотой карточки.



## Фильтрация внутрисистемных дублей для одиночной карточки

Чтобы оставить только те группы дубликатов, где дублем выступает одиночная карточка из исходной системы на одном конце, и карточкой любого типа из этой же системы на другом конце, выберите исходные системы в фильтре и поставьте галочку "только одиночные" для одной из сторон:



Останутся группы, где есть дубликаты с участием одиночной карточки из исходной системы AL с одной стороны, а на другом конце — золотая карточка, среди источников которой есть карточка из AL, или одиночная из системы AL.

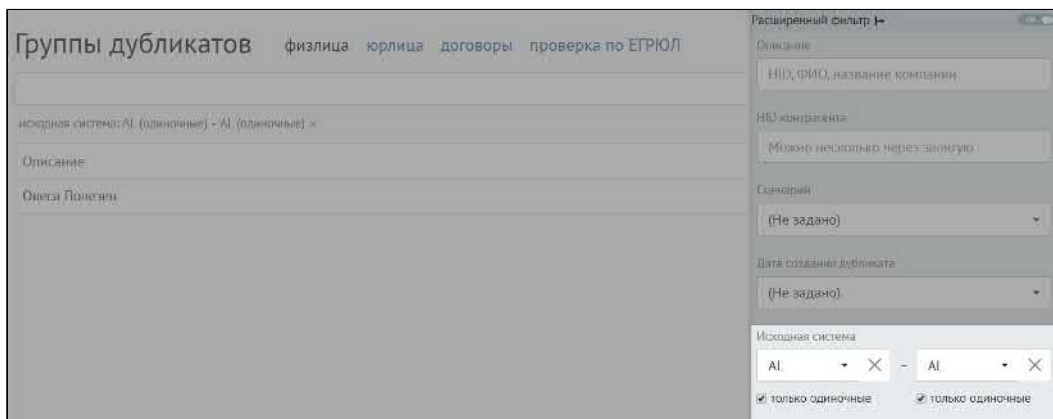
Из предыдущего примера в результат попадут только:

"AL:123" — "AL:456", т.к. здесь обе одиночные карточки из исходной системы AL,  
"BT:23, AL:17" — "AL:98" — здесь, с одной стороны, "AL:123" — одиночная карточка, с другой стороны — "AL:17" входит в состав золотой карточки,

А пара "AL:777, BT:888" — "AL:99, GM:222" не будет показана, т.к. тут с обеих сторон дубликата золотые карточки, хотя они обе содержат исходные записи из системы AL.

## Фильтрация внутрисистемных дублей между одиночными карточками

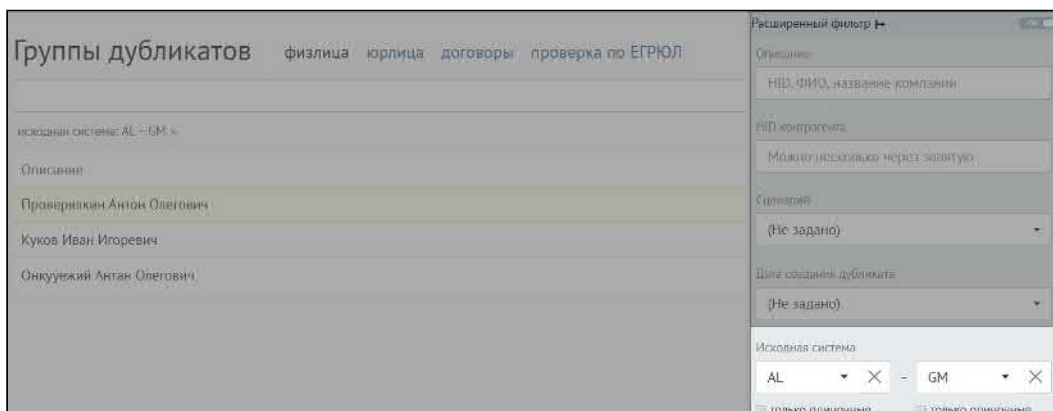
Для поиска дублей между одиночными карточками из одной и той же исходной системы, выберите исходные системы в фильтре и поставьте галочки "только одиночные" для обеих сторон:



В результате останутся только группы дубликатов, где есть пара с двумя одиночными карточками из системы AL, например, "AL:123" — "AL:456" из предыдущего примера.

### Фильтрация межсистемных дублей

Чтобы оставить только те группы дубликатов, где дублями выступают карточки из одной системы (например, AL) к карточкам из другой системы (например, GM), нужно выбрать разные исходные системы:



Так будут показаны все группы дубликатов, в которых есть пара дублей с участием карточек разного типа из исходной системы AL с одной стороны и карточек разного типа из исходной системы GM с другой стороны.

Например, будут найдены группы с парами:

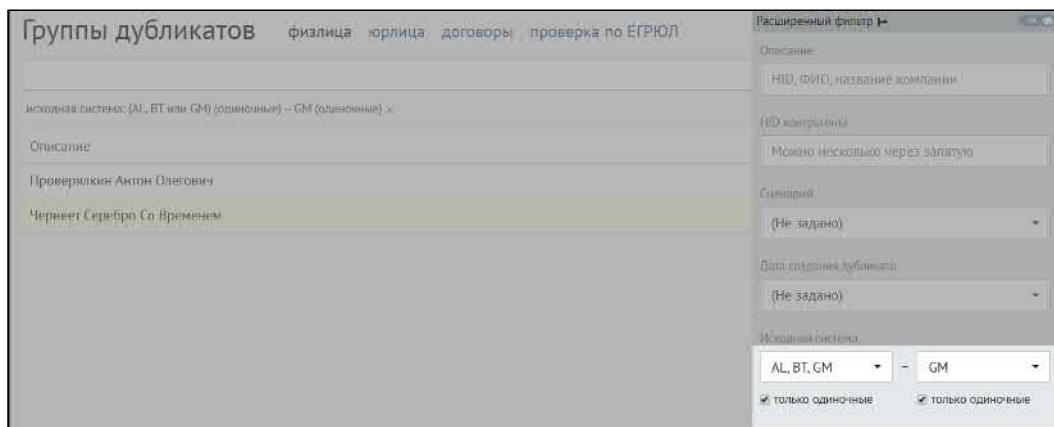
"AL:123" — "GM:333" — одиночные карточки, одна из AL, вторая из GM.

"AL:777, BT:888" — "AL:99, GM:222" — тут карточка из системы AL входит в состав золотой карточки "AL:777, BT:888", а GM входит в состав золотой карточки "AL:99, GM:222".

При этом группа с парой "AL:9090" — "AL:456" не найдется, т.к. здесь обе одиночные карточки из исходной системы AL.

## Фильтрация межсистемных дублей среди одиночных карточек из нескольких систем

Если хотим отобразить дубликаты для одиночных карточек из одной системы к одиночным карточкам сразу из нескольких исходных систем (например, дубли из CRM ко всем бэк-офисным системам), то нужно выбрать несколько исходных систем на одном конце дубля и поставить две галочки "только одиночные":



Так будут показаны все группы, в которых есть пара дублей с участием одиночных карточек из исходных систем AL, VT, GM с одной стороны и одиночной карточки из исходной системы GM с другой стороны, например:

"AL:123" — "GM:456" найдется по этому условию, т.к. здесь одиночная карточка из исходной системы AL с одной стороны и одиночная карточка из исходной системы GM с другой стороны.

"VT:7" — "GM:890" найдется по этому условию, т.к. здесь одиночная карточка из исходной системы VT с одной стороны и одиночная карточка из исходной системы GM с другой стороны.

# ПРОСМОТР ГРУППЫ ДУБЛИКАТОВ

Страница сравнения группы дубликатов содержит 2 блока:

- Список всех пар, входящих в группу;
- Панель сравнения выбранной пары дубликатов.

Список всех пар, входящих в группу:

Аомонович Пётр прогноз слияния ← Alt →

13 контрагентов, [42 дубликата](#)

Контрагенты	Коэф.	Сцен.	Правило	Статус
891 090 – 891 091	КП=76	47	: ФИ, ДР и email, отчество пустое	<span>ДУБЛИКАТ</span>
891 092 – 891 093	КП=76	47	: ФИ, ДР и email, отчество пустое	<span>Не просмотрен</span>
891 124 – 891 125	КП=76	47	: ФИ, ДР и email, отчество пустое	<span>Не просмотрен</span>
891 682 – 891 075	КП=76	47	: ФИ, ДР и email, отчество пустое	<span>Не просмотрен</span>
891 682 – 891 103	КП=76	47	: ФИ, ДР и email, отчество пустое	<span>Не просмотрен</span>
891 682 – 891 085	КП=76	47	: ФИ, ДР и email, отчество пустое	<span>Не просмотрен</span>
891 075 – 891 085	КП=60	46	: ФИО 70-100 + ДР	<span>Не просмотрен</span>
891 075 – 891 103	КП=60	46	: ФИО 70-100 + ДР	<span>Не просмотрен</span>
891 075 – 891 093	КП=60	46	: ФИО 70-100 + ДР	<span>Не просмотрен</span>
891 075 – 891 091	КП=60	46	: ФИО 70-100 + ДР	<span>Не просмотрен</span>

#2 42 из 42 | КП=76 | 47 : ФИ, ДР и email, отчество пустое

Дубликат Не дубликат Отложить Следующая пара →

Alt+Y Alt+N Alt+L

Панель сравнения выбранной пары дубликатов:

891 092 AL	891 093 BT
Источник первой коммуникации Неизвестно	Источник первой коммуникации Неизвестно
▼ Основные атрибуты	
ФИО Йомонович Пётр Семенович	ФИО Йомонович Пётр
Дата рождения 01.01.2000	Дата рождения 01.01.2000
Документ • Удостоверение беженца в РФ • 08.02.2021 номер: 4508 777222 выдан: ОВД Тверской, 13.06.2008	Документ • Паспорт гражданина РФ • 08.02.2021 номер: 45 08 777222 выдан: ОВД Тверской, 028-77, 13.06.2006
Email • Иной Email: first10@gmail.com	Email • Иной Email: first10@gmail.com

Для каждой пары дубликатов на панели сравнения наглядно показано, какие атрибуты у контрагентов совпадают, а какие отличаются (и в чём именно отличие). Синим выделяются недостающие части информации.

Сверху карточки отображён уникальный идентификатор записи в системе «Единый клиент»? и код исходной системы записи.

# РАБОТА С ЧЕРНЫМИ СПИСКАМИ

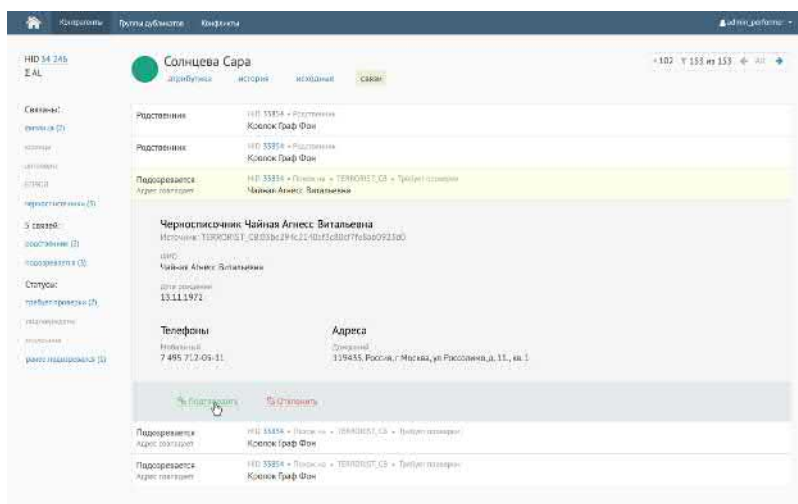
Пользователь работает в стандартном интерфейсе пользователя.

Для работы с нужными клиентами он использует стандартный фильтр "Найдены в черном списке".

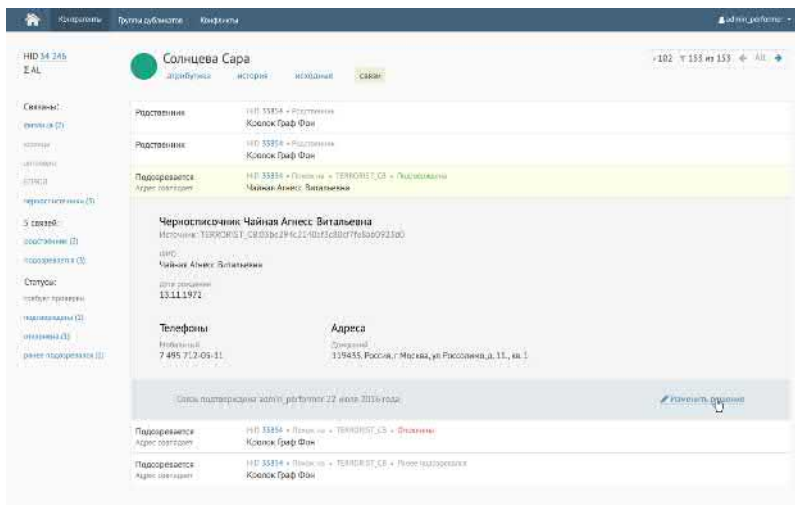
Пользователь открывает карточку и переходит на вкладку «Связи». В левой колонке есть раздел «Статусы», с активными ссылками на все варианты:

- требуют проверки;
- подтверждены;
- отклонены;
- неактуальные.

Пользователь может раскрыть пункт списка, чтобы посмотреть детали карточки связи. Если необходимо, подтвердить или отклонить связи с черными списками.

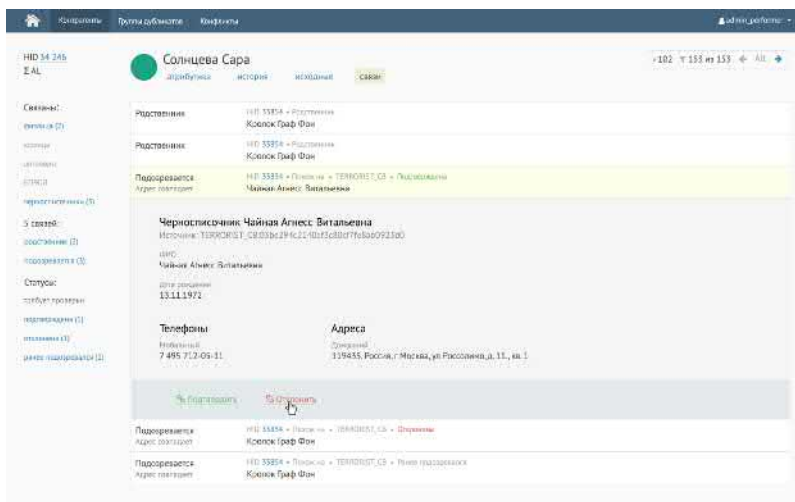


После подтверждения или отклонения связи статус изменяется. В статусе описано кто и когда изменил статус связи в последний раз.



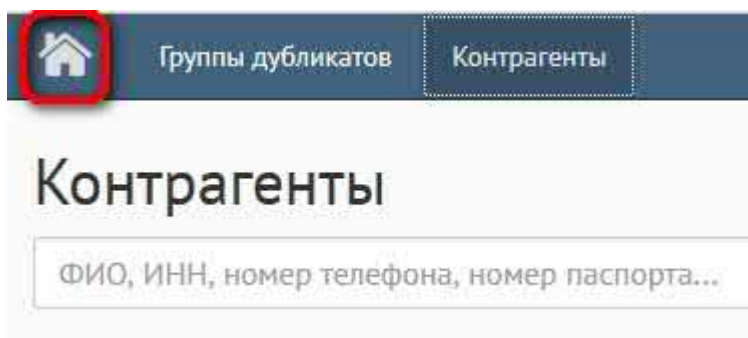
## Изменение решения

Пользователь может изменить решение, нажав на соответствующую ссылку на связи.



## ВЫХОД В АРМ АДМИНИСТРАТОРА

Вернуться обратно в АРМ Администратора из нового интерфейса можно, нажав на иконку в левом верхнем углу



Данная возможность есть только у пользователей, имеющих роль *Администратор*.

Пользователь с ролью *Менеджер данных* при нажатии на эту иконку попадет на страницу списка групп дубликатов.



# ПОДТВЕРЖДЕНИЕ И ОТКЛОНЕНИЕ СВЯЗЕЙ

Связи любого типа можно сделать отклоняемыми. Обратитесь в поддержку ХФЛабс для подключения функционала. Например, включили отклонение связей по домохозяйствам:

Пользователь работает в стандартном интерфейсе пользователя.

Пользователь открывает карточку и переходит на вкладку «Связи». В левой колонке есть раздел «Статусы», с активными ссылками на все варианты:

- требуют проверки;
- подтверждены;
- отклонены;
- неактуальные.

Пользователь может раскрыть пункт списка, чтобы посмотреть детали карточки связи. Если необходимо, подтвердить или отклонить связь.

НІД 1 136  
1:10020

**Куменко Елена Анатольевна**  
атрибутика история исходные **связи**

Связаны:  
физлица (1)  
юрлица  
Автомобили

1 связь:  
домохозяйство (фамилия и телефон) (1)

Статусы:  
требуется проверки (1)  
подтверждена  
отклонена  
неактуальна

Домохозяйство (Фамилия и телефон) НІД 1 137 • Домохозяйство (Фамилия и телефон) • Требуется проверки и телефон  
Куменко Антон Анатольевич

**Физлицо Куменко Антон Анатольевич**  
Источник: 1:20020

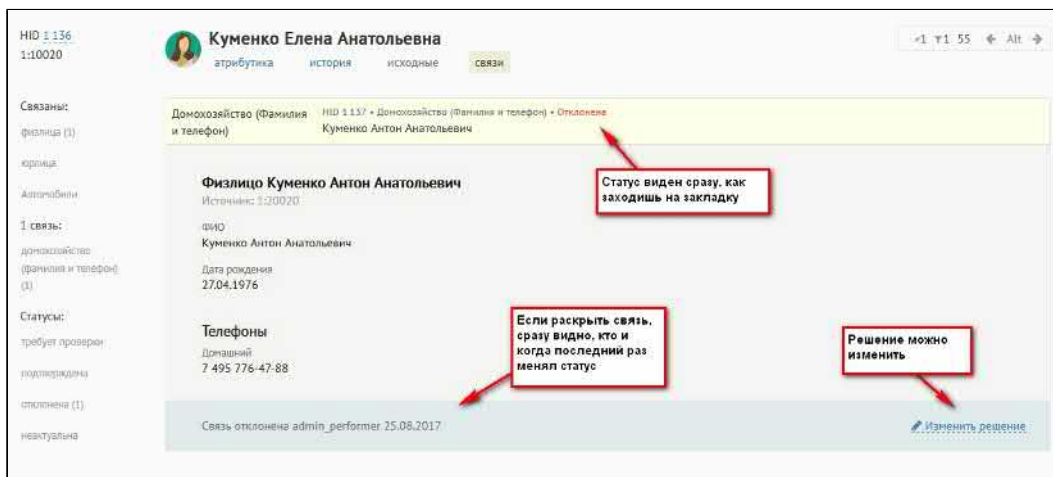
ФИО  
Куменко Антон Анатольевич

Дата рождения  
27.04.1976

**Телефоны**  
Домашний  
7 495 776-47-88

[Подтвердить](#) [Отклонить](#)

После подтверждения или отклонения связи статус изменяется. В статусе описано кто и когда изменил статус связи в последний раз.



## ***Изменение решения***

Пользователь может изменить решение, нажав на соответствующую ссылку на связи.

Для этого нужно раскрыть связь и нажать справа снизу кнопку «изменить решение»

# СЦЕНАРИИ РАБОТЫ МЕНЕДЖЕРА ДАННЫХ

- Проверка информации о контрагентах
  - Копирование адресов по образцу
- Работа с группами дубликатов
  - Подтверждение или отклонение дубликатов
  - Прогноз результатов слияния
- Работа с конфликтующими изменениями
- Разъединение дубликатов контрагентов

# ПРОВЕРКА ИНФОРМАЦИИ О КОНТРАГЕНТАХ

- Увидеть всю информацию по золотой карточке ? проверить и исправить контактные данные
- Разрешить конфликт
- Проверка на дублирование атрибутов
- Создание атрибутов
- Редактирование атрибутов
- Удаление атрибутов
- Сохранение изменений

## Увидеть всю информацию по золотой карточке ? проверить и исправить контактные данные

На карточке клиента дата-стюард видит все атрибуты единым списком. Проблемные данные отмечены желтыми и красными кодами качества. Дата-стюард проверяет их, исправляет и подтверждает.

The screenshot displays a web interface for a customer card. The browser address bar shows the URL: trunk-demo.intranet.hffabs.ru:8080/cdi/ui/parties/PHYSICAL/34528?state=N4lgbglpg7iBcoAuBPADIBIAKAJAmgMoCSA. The page title is 'Контрагенты' and 'Группы дубликатов'. The user 'performer' is logged in. The card details for 'Винников Олег Игоревич' (ID 34528) are as follows:

- Quality: хорошее качество (green bar)
- Basic information: 1 document, 2 addresses, 2 phone numbers, 2 emails, accounts, documents, 12 agreements, additional attributes.
- Attributes:
  - ФИО • ВТ: Винников Олег Игоревич
  - Предыдущее ФИО • ВТ: (empty)
  - Дата рождения • ВТ: 12.12.1989
  - ИНН • ВТ: 1234567890
  - СНИЛС • ВТ: (empty)
- Source identifiers: AL:2017052509116, BT:2017052509111
- Documents: 1 document (Passport of a citizen of the Russian Federation, 45 09 672184)
- Addresses: 2 addresses (Home: 119021, Russia, Moscow, Rossolimo St, 1, apt. 911; Work: 119021, Russia, Moscow, Rossolimo St, 11, apt. 13)

Данные контрагента (реквизиты, адреса, телефоны, email, документы) считаются помеченными на ручную обработку, если хотя бы один из них имеет "сомнительный" код качества. Чтобы проверить корректность таких данных, требуется участие человека. В верхней навигационной панели атрибутов они отображаются желтым или красным цветом.

После внесения изменений (если требуется) в карточку, менеджер данных должен:

- либо сразу подтвердить корректность новых данных, пропуская стандартизацию:  
нажать на кнопку *Подтвердить*.
- либо же инициировать стандартизацию изменённого атрибута и уже после этого подтвердить корректность стандартизованного значения:  
нажать на кнопку *Стандартизовать*;  
нажать на кнопку *Подтвердить*.
- **отложить** принятие решения по атрибуту, запросив дополнительную информацию у внешних систем.

На стандартизацию отправляется только изменённый реквизит или атрибут.

После подтверждения в системе будет создана новая версия соответствующего объекта, а произошедшие изменения будут зафиксированы в истории объекта контрагента (вкладка **История**).

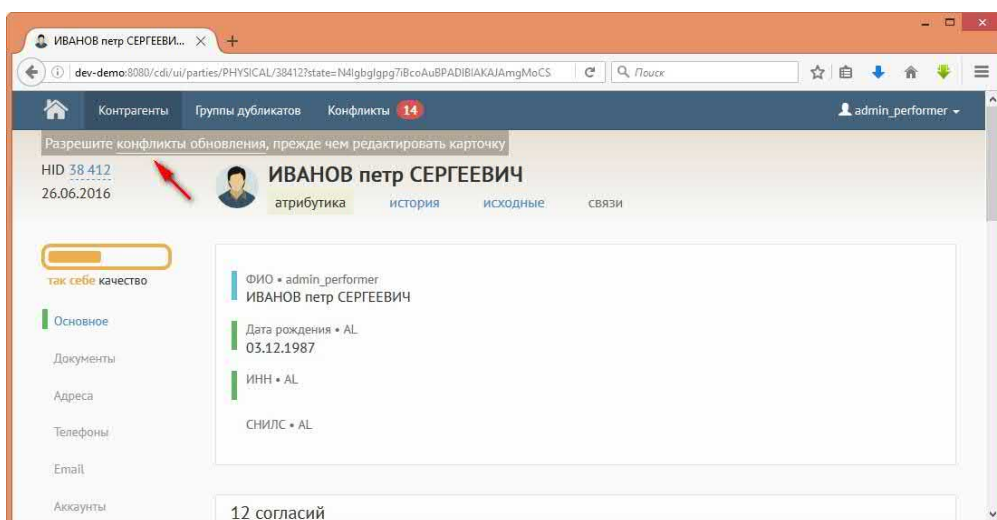
Сложнее всего работать с "сомнительными" адресами, поэтому для их проверки предусмотрена дополнительная функциональность (подсказки по ФИАС). Рекомендуем посмотреть обучающие ролики:

- [Разбор адресов - введение](#)
- [Разбор адресов - исправление ошибок распознавания](#)

## Разрешить конфликт

Если у клиента есть **конфликтное обновление**, редактирование будет запрещено. Менеджер данных должен:

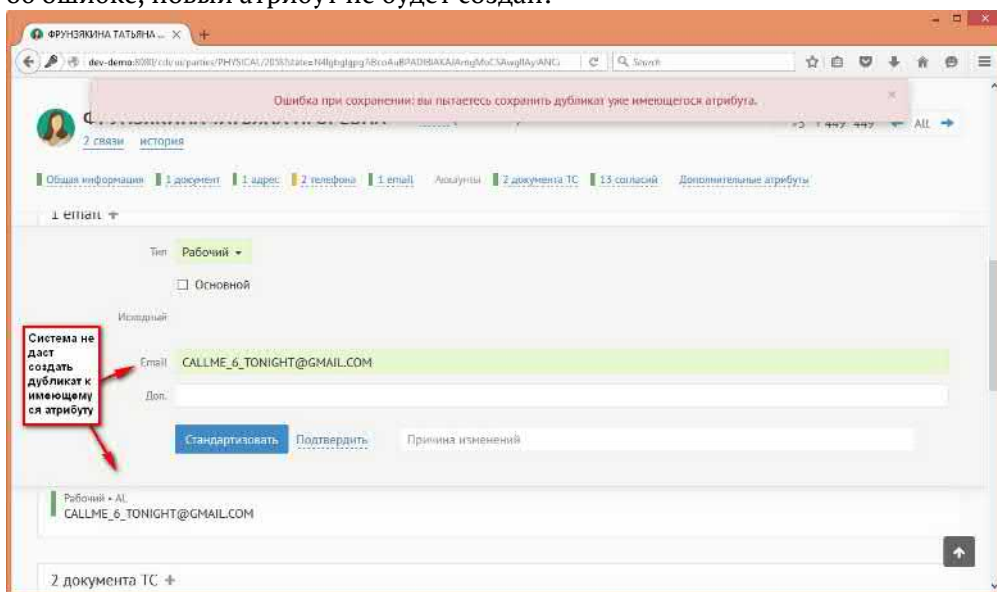
- Нажать на ссылку «разрешите конфликты обновления».
- **Принять решение** по конфликту.
- Вернуться к карточке и продолжить редактирование



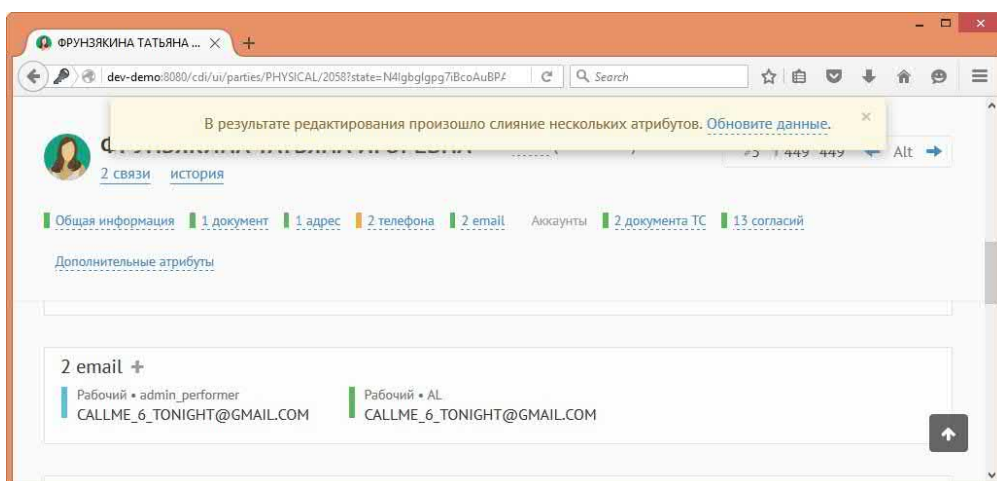
## Проверка на дублирование атрибутов

При создании и редактировании атрибута производится проверка на то, что он не является дубликатом одного из существующих атрибутов.

В случае, когда созданный атрибут — дубль существующего, возникнет следующее сообщение об ошибке, новый атрибут не будет создан:



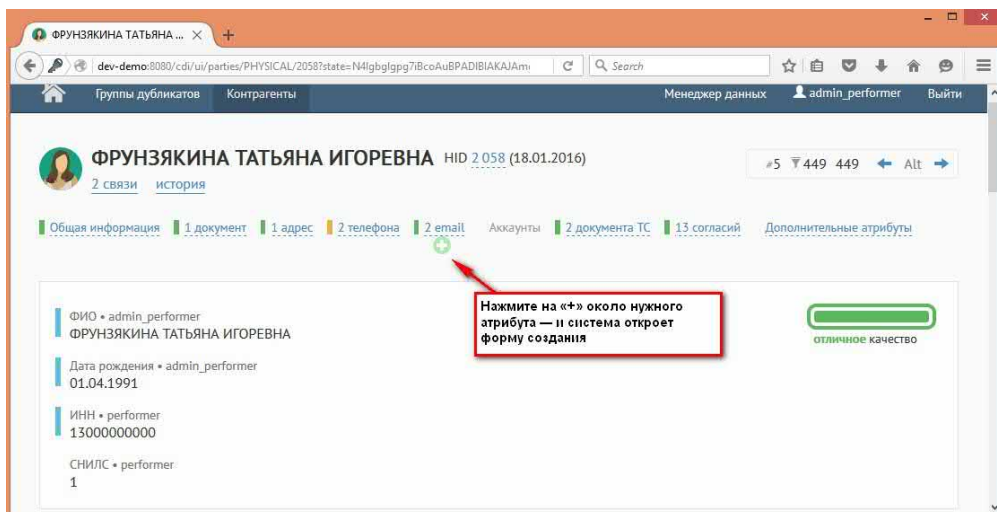
В случае, когда редактирование приводит к дублированию атрибутов, возникнет следующее сообщение, за которым последует слияние атрибутки:



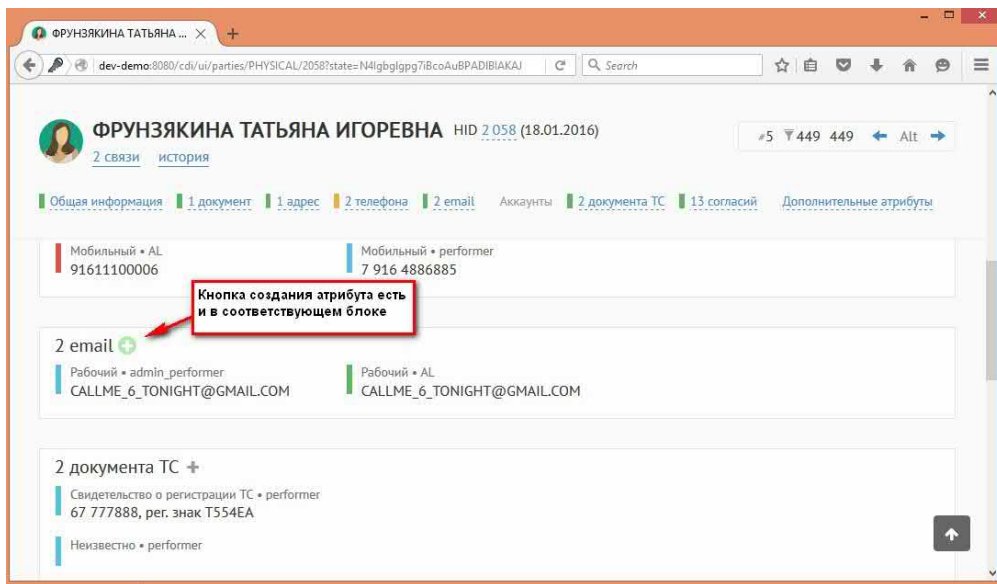
Единый клиент предоставляет возможность слить существующие дубли атрибутов. Для этого на закладке атрибута нужно нажать кнопку *Объединить имеющиеся дубликаты*. Слияние производится только для выбранного типа атрибутов — нажатие кнопки на вкладке с адресами приведет к слиянию адресов, но не телефонов.

## Создание атрибутов

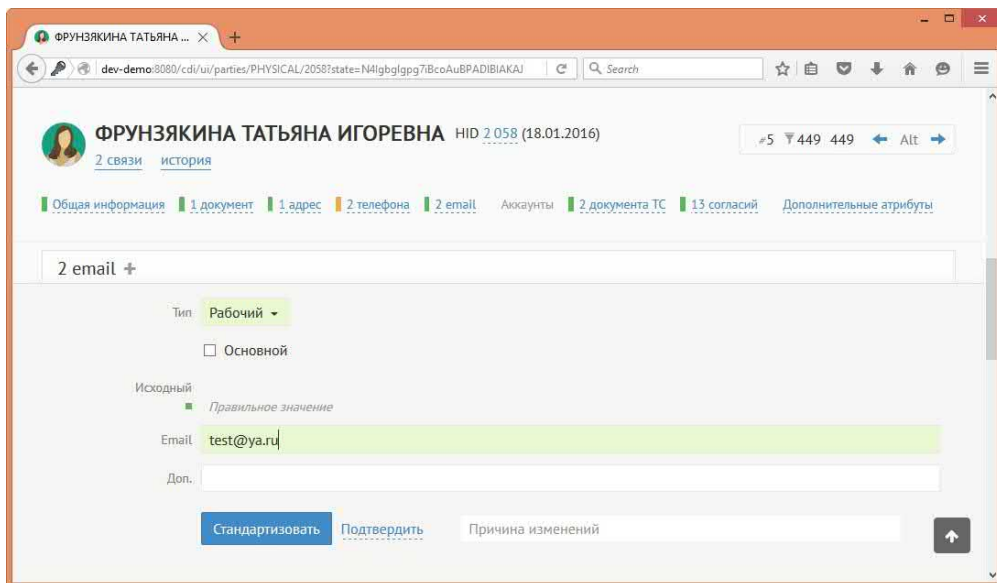
Чтобы создать новый атрибут контрагента (адрес, телефон, email, документ), необходимо нажать на кнопку «+» (*Создать новый атрибут*) в верхней навигационной панели:



Или в самом блоке с атрибутами

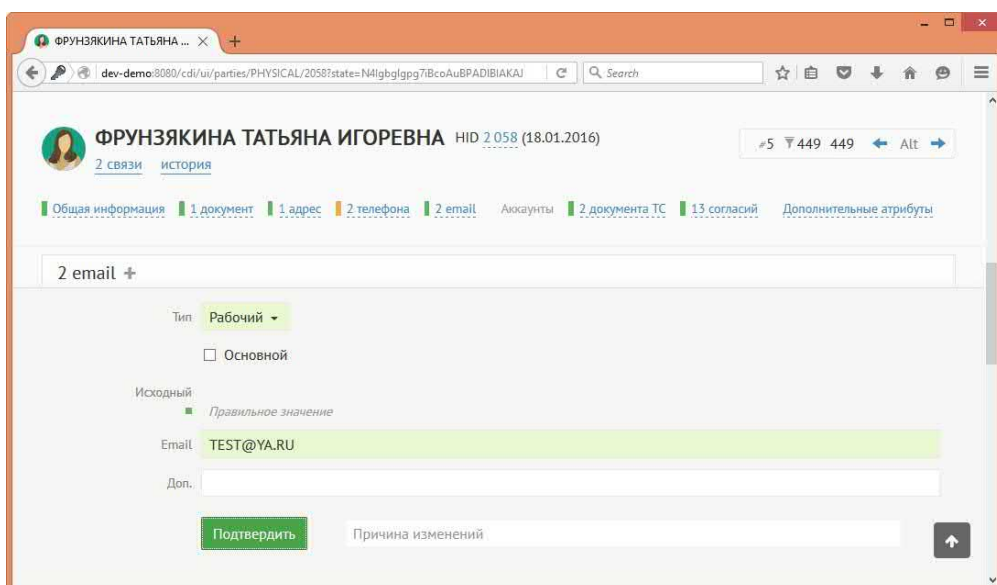


Далее необходимо заполнить поля атрибута и стандартизовать его, нажав на кнопку *Стандартизовать*:



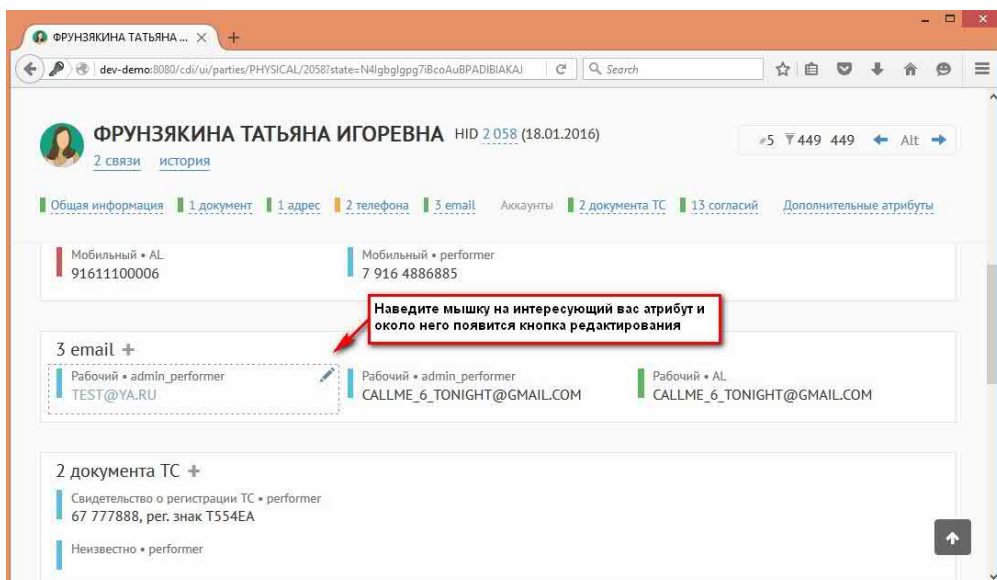
Затем необходимо убедиться в корректности стандартизованных данных и сохранить атрибут, нажав на кнопку *Подтвердить*:





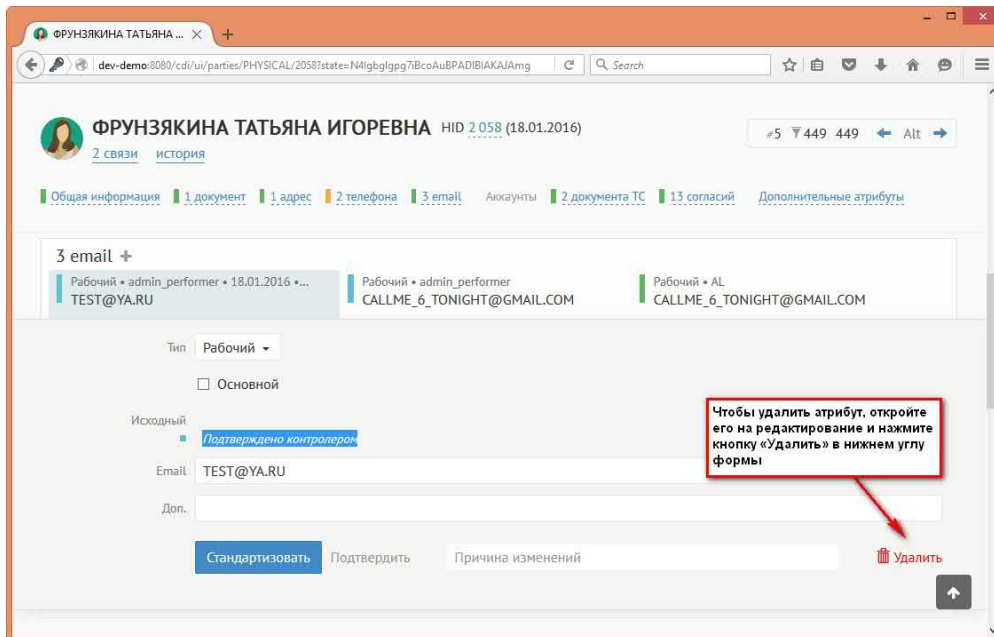
## Редактирование атрибутов

Чтобы отредактировать атрибут контрагента (адрес, телефон, email, документ), необходимо перейти в соответствующий блок, выбрать в списке интересующий вас атрибут и навести на него мышку — атрибут будет выделен, в правом верхнем углу появится кнопка редактирования в виде карандашика:



## Удаление атрибутов

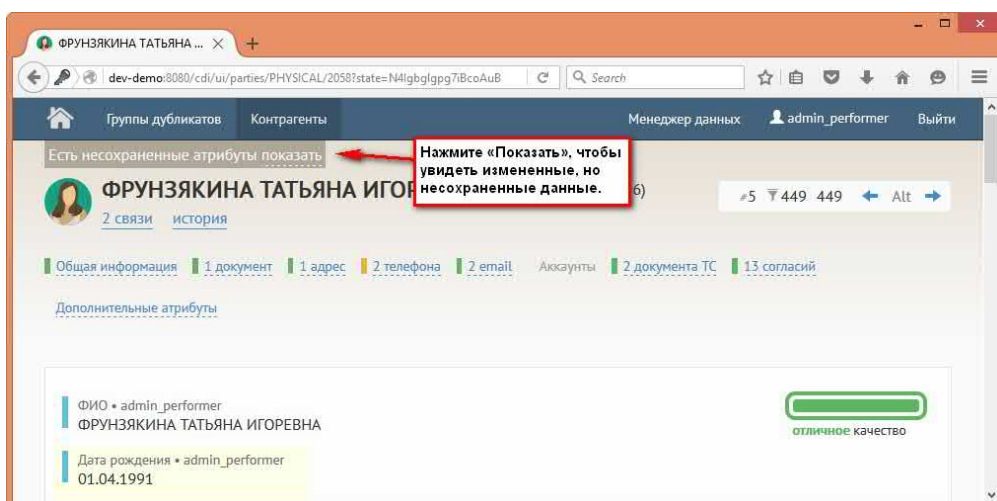
Чтобы удалить атрибут контрагента (адрес, телефон, email, документ), необходимо перейти в соответствующий блок, выбрать в списке интересующий вас атрибут, открыть его на редактирование и нажать на кнопку *Удалить*:



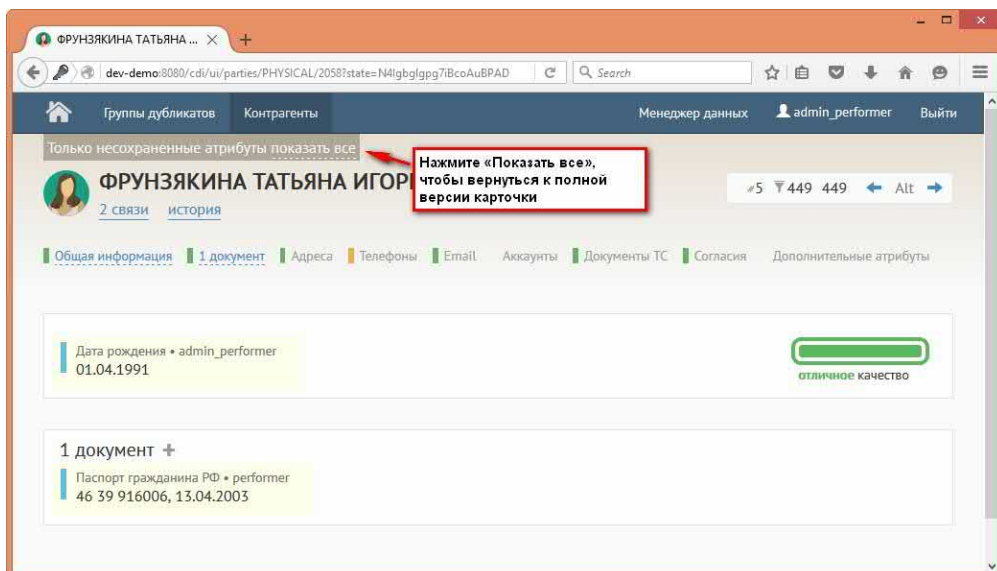
Если вы удалили атрибут ошибочно, и хотите его вернуть, то это можно сделать, восстановившись на предыдущую версию (написал заявку администратору, а он уже выполнит [Задачу восстановления атрибутов по списку НИДов](#)).

## Сохранение изменений

Если какой-либо реквизит, был изменен, но при этом пользователь не отметил его как подтвержденный, система выдаст предупреждение:



Если нажать «Показать», на карточке останутся только измененные, но неподтвержденные атрибуты и кнопка возвращения назад «Показать все»:



Если перейти на карточку другого клиента, несохраненные изменения пропадут.

На добавление нового атрибута без подтверждения система предупреждений не выдает, только на редактирование.

# КОПИРОВАНИЕ АДРЕСОВ ПО ОБРАЗЦУ

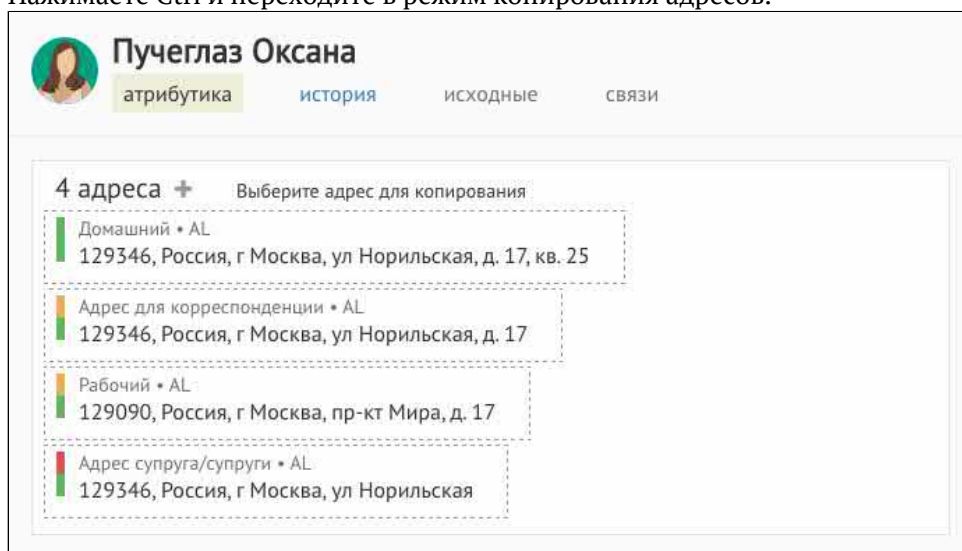
## Копирование адресов по образцу

Бывает, что у клиента несколько похожих «сомнительных» адресов:

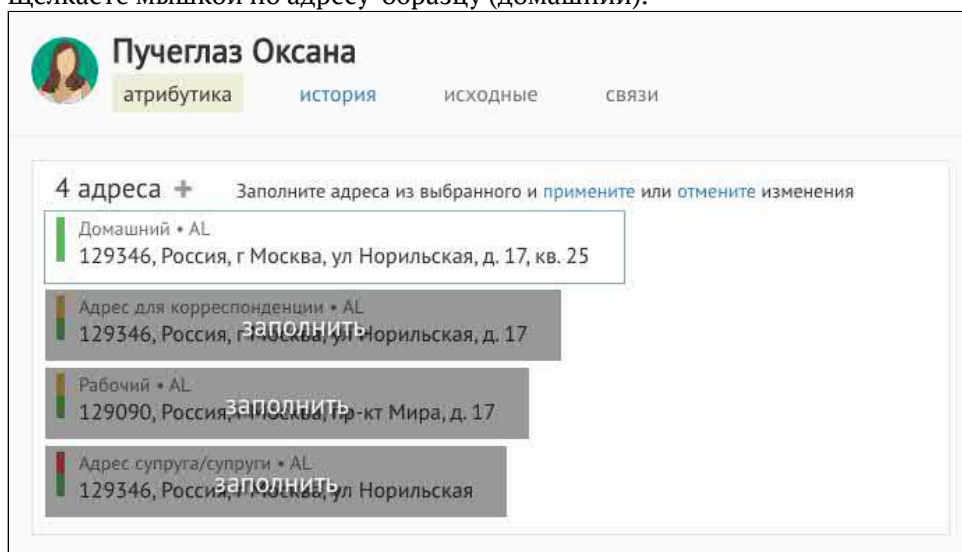
- Домашний: г Москва, ул Норильская, д. 17, кв. 25 (корректный)
- Для писем: г Москва, ул Норильская, д. 17 (сомнительный)
- Супруга: г Москва, ул Норильская (сомнительный)

Чтобы исправить их по образцу одним махом:

1. Нажимаете Ctrl и переходите в режим копирования адресов.



2. Щелкаете мышкой по адресу-образцу (домашний).



3. Выбираете адреса, в которые следует скопировать образец.

Пучеглаз Оксана  
атрибутика история исходные связи

4 адреса + Заполните адреса из выбранного и примените или отмените изменения

- Домашний • AL  
129346, Россия, г Москва, ул Норильская, д. 17, кв. 25
- Адрес для корреспонденции • AL  
129346, Россия, г Москва, ул Норильская, д. 17, кв. 25 ✓ **заполнен**
- Рабочий • AL  
129090, Россия, г Москва, пр-кт Мира, д. 17 **заполнить**
- Адрес супруга/супруги • AL  
129346, Россия, г Москва, ул Норильская, д. 17, кв. 25 ✓ **заполнен**

4. Нажимаете «применить». Готово!

Пучеглаз Оксана  
атрибутика история исходные связи

4 адреса +

- Домашний • AL  
129346, Россия, г Москва, ул Норильская, д. 17, кв. 25
- Адрес для корреспонденции • admin\_performer  
129346, Россия, г Москва, ул Норильская, д. 17, кв. 25
- Рабочий • AL  
129090, Россия, г Москва, пр-кт Мира, д. 17
- Адрес супруга/супруги • admin\_performer  
129346, Россия, г Москва, ул Норильская, д. 17, кв. 25

## РАБОТА С ГРУППАМИ ДУБЛИКАТОВ

При выверке дубликатов всегда приходится решать две разных задачи для каждой конкретной группы дубликатов:

1. Отсеять "ложные" дубли и подтвердить "настоящие".
2. Удостовериться, что в результирующей карточке победят "правильные" значения.

В состав Единого клиента входит специальное рабочее место, которое позволяет решить обе эти задачи. Рабочее место доступно через веб-интерфейс для пользователей, имеющих права *Оператора* или *Менеджера данных*. Подтверждение дубликатов и слияние — только для *Менеджера данных*.

- Список групп дубликатов
- Просмотр группы
- Подтверждение или отклонение дубликатов
- Прогноз результатов слияния

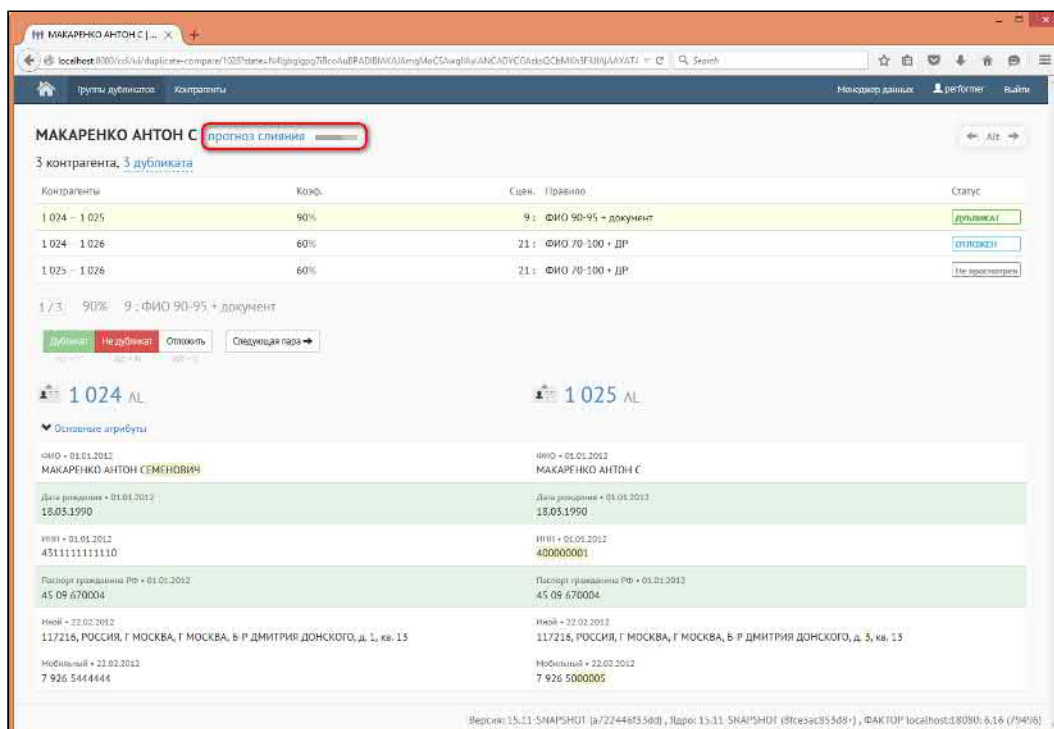
# ПОДТВЕРЖДЕНИЕ ИЛИ ОТКЛОНЕНИЕ ДУБЛИКАТОВ

На странице сравнения группы дубликатов для каждой конкретной пары менеджер данных может:

- Подтвердить факт дублирования, нажав на кнопку *Дубликат*.
- Опровергнуть дублирование, нажав на кнопку *Не дубликат*.
- Отложить принятие решения, нажав на кнопку *Отложить ?* пара дубликатов будет считаться обработанной. Единый клиент не предложит ее дата-стюарду, пока по клиенту не поступит новая информация.
- Принять решение по Конфликту слияния, нажав на кнопку *Дубликат* либо *Не дубликат*.
- Перейти к следующей паре, нажав на кнопку *Следующая пара*. Система переходит к следующей необработанной паре группы по списку, статус текущей пары остается без изменений. Если фокус был на последней паре группы, то совершается переход на следующую группу.

После подтверждения хотя бы одной пары из группы, становится доступным ручное слияние, переход по ссылке [Прогноз слияния](#).

При попытке слияния группа проверяется на конфликт.



Контрагенты | Группы дубликатов | Отчёты | performer

**Кареев Александр Витальевич** прогноз слияния ← Alt →

5 контрагентов, [10 дубликатов](#)

Контрагенты	Коэф.	Сценарий	Статус	Конфликт
623 916 706 – 952 974 919	60	46 : ФИО 70-100 + ДР	<span style="color: blue;">ОТЛОЖЕН</span>	<span style="color: red;">КОНФЛИКТ слияния</span> -71: Разные ДР
623 916 706 – 1 021 933 077	60	46 : ФИО 70-100 + ДР	<span style="color: green;">ДУБЛИКАТ</span>	
623 916 706 – 1 021 812 719	60	46 : ФИО 70-100 + ДР	Не просмотрен	<span style="color: red;">КОНФЛИКТ слияния</span> -74: Разные ДР (зеркальные)
623 916 706 – 1 021 816 439	60	46 : ФИО 70-100 + ДР	Не просмотрен	
952 974 919 – 1 021 933 077	60	46 : ФИО 70-100 + ДР	Не просмотрен	
952 974 919 – 1 021 816 439	60	46 : ФИО 70-100 + ДР	Не просмотрен	
952 974 919 – 1 021 812 719	60	46 : ФИО 70-100 + ДР	Не просмотрен	
1 021 812 719 – 1 021 933 077	60	46 : ФИО 70-100 + ДР	Не просмотрен	
1 021 812 719 – 1 021 816 439	60	46 : ФИО 70-100 + ДР	Не просмотрен	

#1 ↓ 10 из 10 КП=60 46 : ФИО 70-100 + ДР Отложил performer 25.05.2021 (Недостаточно данных – нет счетов)

Конфликт слияния -71: Разные ДР

Дубликат
Не дубликат
Изменить причину ▾
Следующая пара →

Alt + Y Alt + N Alt + L

**623 916 706** ВТ  **952 974 919** ВТ

Если пользователь объединил группу, где подтверждена лишь часть дубликатов, то она автоматически актуализируется, после чего с ней можно продолжать работать дальше.



# ПРОГНОЗ РЕЗУЛЬТАТОВ СЛИЯНИЯ

На странице отображается прогноз золотой карточки контрагента (какой она получится после объединения всех подтвержденных пар дубликатов).

Показываются все атрибуты контрагента. Если по атрибуту есть конфликт (т.е. значения этого атрибута у исходных контрагентов отличаются между собой), то он подсвечен желтым цветом.

Доступна фильтрация атрибутов *Все атрибуты / Только конфликтующие*.

← Сравнение дубликатов

## Прогноз слияния

Выполнить слияние

Все атрибуты Конфликтующие Alt + K

▼ Основные атрибуты

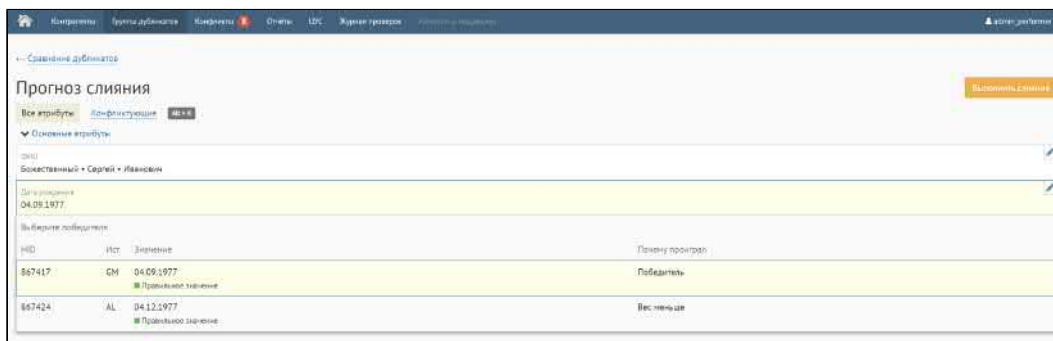
ФИО	Макаренко • Антон • Семенович
Дата рождения	02.04.1989
Паспорт гражданина РФ	46 39 916222
Домашний	108841, Россия, г Москва, г Троицк, ул Садовая, д. 10, кв. 111
Рабочий	454000, Россия, обл Челябинская, г Челябинск

▼ Дополнительные атрибуты

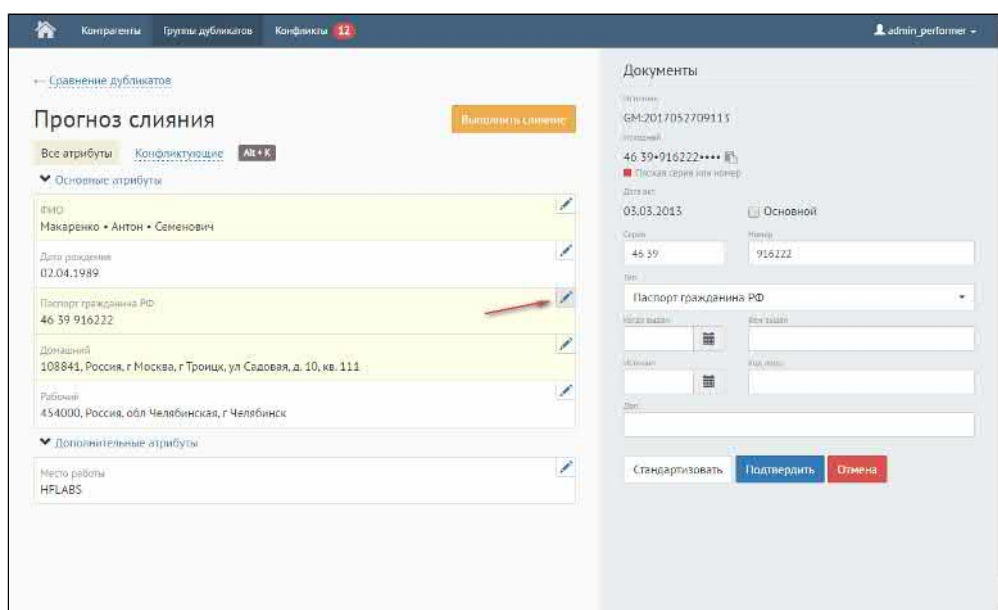
Место работы	HFLABS
--------------	--------

Версия: 17.5-SNAPSHOT (69da2f65e732), Ядро: 17.5-SNAPSHOT (8c8e3b2a0381), ФАКТОР localhost:18080: 8....

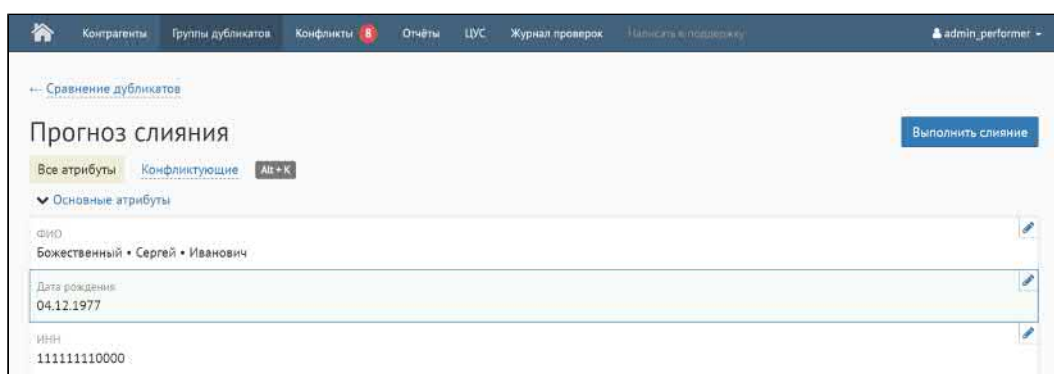
При выборе конфликтного атрибута открывается панель выбора победителя. В ней можно увидеть, какое из исходных значений должно победить при слиянии и почему проиграла остальные:



Все атрибуты можно отредактировать, нажав на иконку — справа появится форма для редактирования, где можно внести изменения по атрибуту.

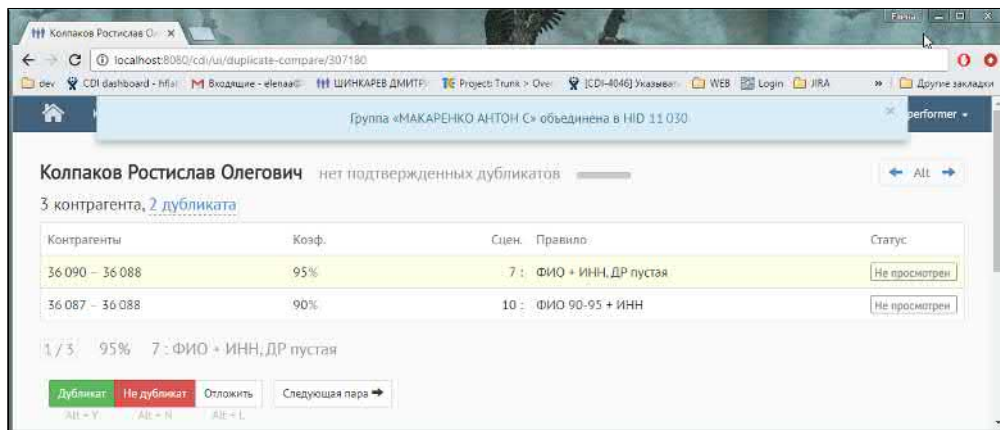


Результат слияния можно поменять, выбрав победителя вручную и нажав «Выполнить слияние»:



После этого можно переходить к следующей группе дубликатов.

Ссылка на хид "золотого" контрагента будет отображена на информационной плашке, которая автоматически скрывается через 10 секунд.



При ручном слиянии в системе сохраняется информация о том:

- кто провел слияние;
- был ли изменен победитель по конфликтному атрибуту.

Посмотреть можно через АРМ Менеджера данных (desktop) на вкладке *История*.

# РАБОТА С КОНФЛИКТУЮЩИМИ ИЗМЕНЕНИЯМИ

## Проблема

Единый клиент классно объединяет клиентские карточки в одну, но не защищает объединенную карточку от «сомнительных» изменений.

### ***Пример 1: «повторное использование» идентификаторов в исходных системах***

Есть объединенный клиент:

Орлангуров Карл Давыдович  
адрес: Москва, проспект Мира, 27  
паспорт: 4507 622332

И вдруг на него приходит обновление:

Рюриков Михаил Аркадьевич  
адрес: Москва, Турчанинов переулок, д 6  
паспорт: 4500 623452

Конечно, Карл мог настолько радикально изменить свою жизнь. Но верится с трудом. Скорее всего, произошла ошибка в исходной системе: завели нового клиента (Мишу) с идентификатором старого (Карла).

### ***Пример 2: фрод***

Есть клиент:

Рюриков Михаил Аркадьевич  
адрес: Москва, Турчанинов переулок, д 6  
паспорт: 4500 623452  
счастливый обладатель последнего выпущенного Вейрона, часов Вашерон Константин и 100 500 000 рублей на счете

И есть злоумышленник Карл, тоже клиент:

Орлангуров Карл Давыдович  
адрес: Москва, проспект Мира, 27  
паспорт: 4507 622332

Карл зарегистрирован в интернет-банке и (о нет!) у него есть доступ к одной из исходных систем. В понедельник Карл меняет в исходной системе адрес:

Москва, проспект Мира, 27 ? Москва, Турчанинов переулок, д 6

Во вторник Карл меняет в исходной системе паспорт:

4507 622332 ? 4500 623452

В среду Карл потирает потные ладони и меняет в исходной системе ФИО:

Орлангуров Карл Давыдович ? Рюриков Михаил Аркадьевич

В этот самый момент оба клиента становятся 100% дублями, Единый клиент их стремительно объединяет. Карл бежит в интернет-банк, видит там счета Миши, уводит с них все деньги, а Вашерон Константин продает на авито по фотографии. Миша грустит, пьет горькую и разбивает свой Вейрон вдребезги.

## Решение

1. Отслеживать подозрительные изменения при обновлении клиентов.
2. Не применять их автоматически.
3. Показывать дата стюарду, чтобы тот принял решение: применять или нет.

Конфликты отслеживаются как для объединенных карточек, так и для «обычных» (которые ни с кем не объединялись). По обычному ведь тоже может быть ошибочное обновление.

ЕК отслеживает конфликты при обновлении клиента через буферную таблицу, JMS либо онлайн-сервис (SOAP, REST). При редактировании в Менеджере данных конфликты не проверяет — дата-стюард знает, что делает.

## В интерфейсе

В главном меню — пункт «Конфликты» с маркером, если есть конфликты. Если конфликтов больше 1000, пишет «1000+».



Отдельный список клиентов, по которым есть подозрительные изменения:

Конфликты обновления				
Правило, HID, контрагент				
Дата обновления	Автор обновления	Правило	HID	Контрагент
23 нояб. 14:52	AL:465985	ФИО	770417	Курко Олег Игоревич
23 нояб. 14:08	AL:1574	ОГРН, ИНН	801456	МАГНИТОГОРСКИЕ ЭЛЕКТРОТЕПЛОВЫЕ СЕТИ
24 июня 23:08	AL:25555	ФИО	797426	Достоевский Федор Михайлович
13 янв. 12:57	AL:59433	ФИО	780416	Документов Юрий

На списке есть строка поиска — можно искать конфликты по правилу, хиду КА (и исходного, и слитого), наименованию КА.

Список фильтруется:

- по дате обновления;
- исходной системе, из которой пришло обновление;
- атрибуту, по которому обнаружен конфликт (фамилия, имя, дата рождения);
- идентификатору контрагента (в том числе итогового).

Если наложен фильтр, отфильтрованные конфликты можно массово «пропустить» на обновление или отклонить. Если в фильтрах ничего не выбрано, кнопки действий скрыты.

На странице конкретного конфликта — сравнение обновления и существующей карточки. Для объединенной карточки дополнительно показываем исходную:

Курков Олег Андреевич конфликт обновления: ФИО 15 авг. 09:55		
<input type="button" value="Разрешить обновление"/> <input type="button" value="Различь и разрешить"/> <input type="button" value="Отклонить"/>		
Исходный 9 833 AL:224565	Итоговый 9 835 AL:BT	Обновление AL:224565
ФИО • 22.02.2022 Курков • Олег • Андреевич	ФИО • 22.02.2022 Курков • Олег • Андреевич	ФИО • 22.02.2022 Смирнов • Алексей • Андреевич
Пол • 22.02.2022 Мужской	Пол • 22.02.2022 Мужской	Пол • 22.02.2022 Мужской
Дата рождения • 22.02.2022	Дата рождения • 22.02.2022 12.12.1989	Дата рождения • 22.02.2022
Документ • Паспорт гражданина РФ • 22.02.2022 номер: 45 09 672184	Документ • Паспорт гражданина РФ • 22.02.2022 номер: 45 09 672184	Документ • Паспорт гражданина РФ • 22.02.2022 номер: 65 09 672184
Телефон • Мобильный • 22.02.2022 7 916 534-56-22	Телефон • Мобильный • 22.02.2022 7 916 534-56-22	Телефон • Мобильный • 22.02.2022 7 916 534-56-22

Подсвечиваем отличия:

- между обновлением и исходной карточкой;
- между обновлением и объединенной карточкой.

Отличия между исходной и объединенной карточкой не подсвечиваем. Если в дополнительных атрибутах отличий нет, их не показываем.

## Действия дата-стюарда

1. Разрешить обновление («сыграть» его по обычным правилам обновления). Обновление применяется целиком: нельзя разрешить ФИО, а ИНН отклонить.
2. Вылить исходную карточку, на которое пришло обновление, и применить на нее обновление.
3. Отклонить обновление и пойти разбираться в исходную систему.

Действие записывается в журнал аудита:

Дата	Пользователь	Тип действия	Комментарий
12.10.2015	valera	resolveConflict	Обновить контрагента HID '9345445', ID стейджинга '200133454'
12.10.2015	valera	resolveConflict	Разлить и обновить контрагента HID '8345446', ID стейджинга '100133455'
12.10.2015	valera	resolveConflict	Отклонить обновление контрагента HID '6345447', ID стейджинга '900133456'

Если пользователь выбрал вариант «Отклонить», то отклоненное обновление сохраняется в БД. В других случаях — не сохраняется.

## Через SOAP/REST

Если обновление конфликтующее, то запрос сохранения (save, saveAndMerge) возвращает текущее (неизмененное) состояние контрагента. Как будто обновление проиграло.

## Правила обнаружения конфликтов (16.6+)

### *Что делать с обновлениями после конфликтов*

Если по клиенту уже есть конфликтное обновление, то с последующими обновлениями ЕК поступает по-разному — в зависимости от того, полное обновление или частичное.

полное обновление;

частичное обновление (16.8+).

# РАЗЪЕДИНЕНИЕ ДУБЛИКАТОВ КОНТРАГЕНТОВ

## *Разъединение исходных карточек*

Иногда из-за ошибок в исходных данных разные люди объединяются в одну карточку. Тогда дата-стюард разъединяет («разливает») их.

Чтобы отделить одного из исходных контрагентов от объединенного контрагента, необходимо:

1. Перейти на карточку объединенного контрагента.
2. На вкладке *Исходные записи* выбрать исходного контрагента, которого необходимо отделить.
3. Нажать на кнопку *Разъединить*.

В результате разъединения система создаст новую версию отчуждаемого контрагента, в которой будут учтены все обновления и редактирования, произошедшие с момента слияния до момента разъединения. Система также создаст новую версию объединенного контрагента, исключив из нее значения атрибутов отчуждаемого контрагента.

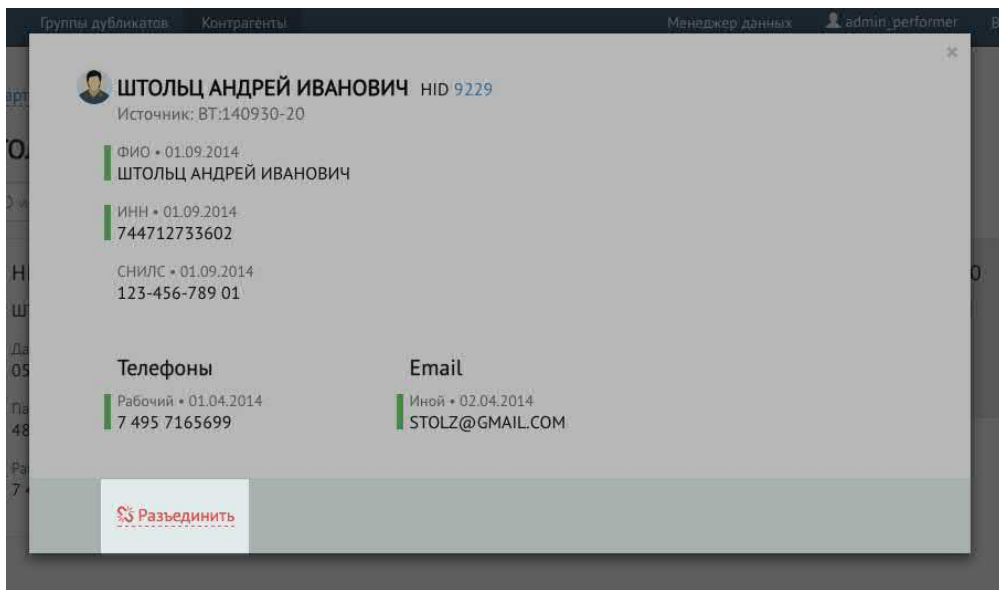
**ШТОЛЬЦ АНДРЕЙ ИВАНОВИЧ** исходные контрагенты

НID или исходный идентификатор  ✕ 🔍 AL (1) BT (2) всего 3

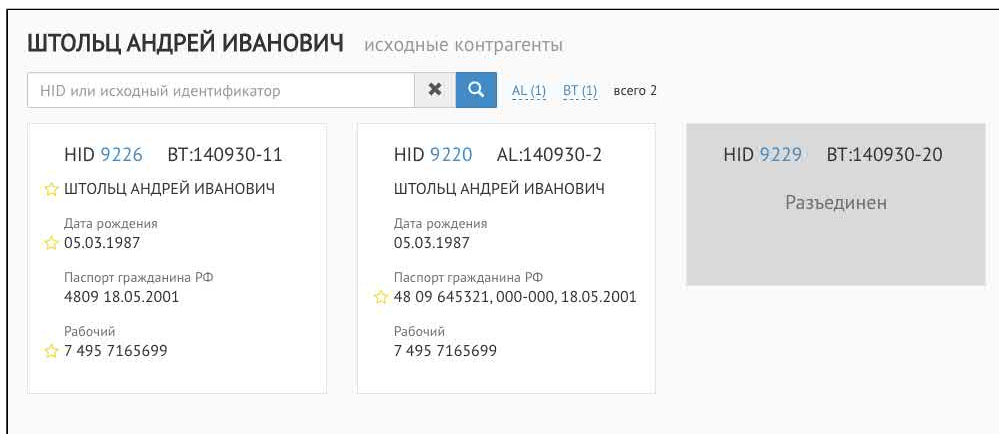
<p>НID 9226 BT:140930-11</p> <p>★ ШТОЛЬЦ АНДРЕЙ ИВАНОВИЧ</p> <p>Дата рождения ★ 05.03.1987</p> <p>Паспорт гражданина РФ 4809 18.05.2001</p> <p>Рабочий ★ 7 495 7165699</p>	<p>НID 9220 AL:140930-2</p> <p>ШТОЛЬЦ АНДРЕЙ ИВАНОВИЧ</p> <p>Дата рождения 05.03.1987</p> <p>Паспорт гражданина РФ ★ 48 09 645321, 000-000, 18.05.2001</p> <p>Рабочий 7 495 7165699</p>	<p>НID 9229 BT:140930-20</p> <p>ШТОЛЬЦ АНДРЕЙ ИВАНОВИЧ</p> <p>Рабочий 7 495 7165699</p>
--	---	---

?






?



## ***Изменение решения***

Если решение о разъединении нужно изменить, то ранее разъединенные карточки можно найти на экране «Отклоненные дубликаты» карточки контрагента.

HID 2 775  
BT:20180622424246  
24.02.2017

 **Исторический Иннокентий**  
атрибутика история исходные связи **отклоненные дубли**

HID или ID

**Дубликаты:**  
Не дубликат (1)  
Разлит

**Системы:**  
Объединенные (1)

**Сценарии:**  
ФИО 70-100 + телефон (1)

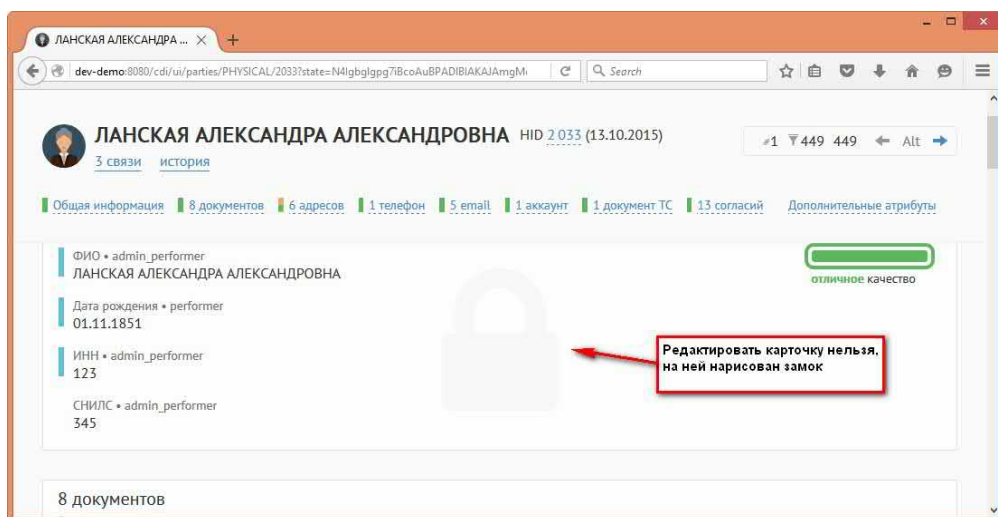
HID 2 754 AL:2018062242424...  
Исторический Иннокентий Варламович  
Дата рождения  
14.01.1980

Нажав на карточку, можно увидеть сравнение данных, история статуса, а также изменить решение по разъединению.

# СЦЕНАРИИ РАБОТЫ ОПЕРАТОРА

Пользователи в роли *Оператор* могут работать со **списком контрагентов**, а также просматривать **карточки контрагентов** и **карточки групп дублей**.

Для данной роли карточки показываются в режиме read-only (изменять данные контрагента запрещено).



# РАСШИРЕННЫЙ ПОИСК

- Бизнес-смысл
- Описание
- Синтаксис

Равенство

Неточный поиск

Комбинация условий

Исключающий фильтр

Перечисления

Неравенства и диапазоны

Поиск по разным типам клиентов

## Бизнес-смысл

Расширенный поиск — это возможность искать контрагентов, не ограничиваясь стандартными рамками, строить гибкие отчеты и составлять различные комбинации условий.

В системе обнаружилось 100 000 дубликатов? А сколько из них найдено по конкретному сценарию? Например, сценарий № 11 — насколько он важен, много ли дубликатов было найдено именно по нему?

Сколько есть дубликатов со 100% коэффициентом соответствия? Сколько из них имеют адрес без указания дома? Имеют кредитную карточку вашего банка? Или старше 30 лет? Или... Варианты возможны любые.

## Описание

Расширенный поиск можно применять:

- В АРМ Менеджера данных при работе со списком контрагентов.
- При вызове [Party.search](#).

Расширенный поисковый запрос всегда должен начинаться с вопросительного знака:

**? Passport.documentNumber = 638431**

Название нотаций в поиске регистрозависимое.

Перечень реквизитов и атрибутов, по которым возможен расширенный поиск, доступен в описании модели данных в столбце **«Расширенный поиск»** в пространстве каждого заказчика.

Для упрощения работы с расширенным поиском в веб-интерфейсе реализованы подсказки на каждую часть запроса — начните свой запрос с вопросительного знака и следуйте подсказкам.

Примеры наименований полей расширенного поиска по некоторым реквизитам и атрибутам:

- Поиск по реквизитам контрагента
- Поиск по реквизитам юрлица
- Поиск по адресам
- Поиск по телефонам
- Поиск по email
- Поиск по документам
- Поиск по дубликатам
- Поиск по группам дубликатов
- Поиск по аккаунтам
- Поиск по реквизитам любого контрагента

## Синтаксис

### *Равенство*

Чтобы обозначить равенство, можно использовать символ **=**.

Контрагенты с фамилией *Иванов*:

```
? PhysicalParty.surname = Иванов
```

Контрагенты, родившиеся *18.03.1975*:

```
? PhysicalParty.birthdate = 18.03.1975
```

Пробел интерпретируется как разделитель, составные значения нужно передавать в кавычках:

```
? PhysicalParty.patronymic = "Бахтырай Оглы"
```

«Е» и «Ё» воспринимаются, как одно и то же. По запросу «СЕЛЕЗНЕВА» найдется и «селезнЕва», и «селезнЁва»:

```
? PhysicalParty.surname = селезнева
```

## Неточный поиск

В строковых полях можно искать по части слова, используя символ \*, который означает "любое количество любых символов"

Контрагенты с фамилией *Иванов*:

```
? PhysicalParty.surname = Ива*
```

Пробел экранируем через слеш

```
? LegalParty.name = *часовой\ пояс*\ Омск*
```

Двоеточие тоже экранируем через слеш

```
? PhysicalParty.mergeAuthor = AL:RADAR*
```

Если нужно искать именно кавычки, их тоже экранируем

```
? LegalParty.shortName = *\"Яблочко*\"  
? LegalParty.name = *Студия\ \"Яблочко\"* --- экранируем сначала пробел, потом  
кавычку
```

Примеры правил поиска с пробелами и кавычками:

```
- Поискать ЮЛ с пробелами в названии: [? LegalParty.name = *\ *]  
- Поискать ЮЛ с двумя пробелами в названии [? LegalParty.name = *\ * \ *]  
- Поискать ЮЛ у которого есть часть в кавычках, а внутри неё пробел [? LegalParty.name  
= *\"*\ *\"*]  
(первая и последняя звёздочки для "любое начало и конец"; кавычки экранированы, т.  
к. спец.символ; между кавычками стоит "что угодно, пробел, что угодно")
```

Просто написать

```
? LegalParty.name = *часовой пояс Омск*
```

Нельзя ? пробел означает, что поиск по данной сущности закончен и дальше или новый запрос, или простой поиск, не расширенный. То есть мы получили три условия поиска:

- ? LegalParty.name = \*часовой
- Пояс
- Омск\*

Из них «\*часовой» относится к наименованию, а остальное — к любым индексируемым полям.

Если брать в кавычки, не будет работать символ \*

## Комбинация условий

Чтобы выполнялись несколько условий поиска одновременно, необходимо использовать выражение **AND (И)**.

Например, *Ивановых*, родившихся *18.03.1975*, можно найти таким запросом:

```
? PhysicalParty.surname = Иванов and PhysicalParty.birthdate = 18.03.1975
```

Чтобы выполнялось хотя бы одно условие поиска, необходимо использовать выражение **OR (ИЛИ)**.

Например, поиск контрагентов с фамилией *Иванов* или *Кутузов*:

```
? PhysicalParty.surname = Иванов or PhysicalParty.surname = Кутузов
```

Можно использовать и простой поиск, и расширенный ? если не экранировать пробел символом «\», то пробел в расширенном поиске означает, что дальше мы ищем полнотекстово.

Поэтому можно совместить — сначала все условия по атрибутам, потом пробел и ФИО одной строкой

```
? Phone.aggregateNumber = 4953332243 Иванов Иван Иванович
```

## Исключающий фильтр

Исключающее условие **NOT(не)** работает только совместно с каким-либо позитивным фильтром.

Например, выбираем карточки с датой рождения *18.03.1975*, источником которых не является система AL:

```
? PhysicalParty.birthdate = 18.03.1975 and not PhysicalParty.mergeAuthor = *AL*
```

Если фильтрация нужна только по одному столбцу, сначала явно проводим выборку всех данных по этому столбцу, а затем через **AND** добавляем исключение.

Например, чтобы выбрать все карточки, источником которых не является система AL, указываем составное условие:

```
? PhysicalParty.mergeAuthor = * and not PhysicalParty.mergeAuthor = *AL*
```

❗ Так не работает!!!

Фильтр с отрицанием без позитивного условия работать не будет:

```
? not PhysicalParty.mergeAuthor = *AL*
```

## Перечисления

Поиск по вхождению значения в множество значений (перечисление) выполняется с помощью выражения **OR (ИЛИ)**.

Например, для поиска всех контрагентов с указанными идентификаторами:

```
? PhysicalParty.historicalId = (1024 or 1025 or 1026 or 2048)
```

## Неравенства и диапазоны

Для числовых полей и дат можно делать выборки по условиям на неравенство:

- >
- >=
- <
- <=

Например, для поиска всех контрагентов старше 20 лет:

```
? PhysicalParty.birthdate < 01.01.1993
```

Если дата может включать время, то для поиска по конкретному дню необходимо использовать диапазон дат. Например, чтобы найти контрагентов с датой актуальности 15.01.2013:

```
? PhysicalParty.latestActualityDate >= 15.01.2013 and PhysicalParty.latestActualityDate < 16.01.2013
```

Условие на диапазон также можно указать с помощью интервала:

```
? PhysicalParty.birthdate : [18.03.1975 TO 18.03.1995]
```

Квадратные скобки **[]** указывают, что поиск включает границы интервала, фигурные скобки **{}** — что не включает.

## Поиск по разным типам клиентов

В Менеджере данных можно искать только по той вкладке, на которой вы находитесь. Но через SOAP/REST запрос [Party.search](#) можно искать сразу по разным типам клиентов

Например, поиск по ИНН среди ФЛ и ЮЛ:

```
? PhysicalParty.inn = 7705974943 OR LegalParty.inn = 7705974943
```



# ПОИСК ПО РЕКВИЗИТАМ КОНТРАГЕНТА

Пример запроса (контрагенты, объединенные за 30.01.2013):

```
? PhysicalParty.mergedStatus = RESULT and PhysicalParty.startDate >= 30.01.2013 and PhysicalParty.startDate < 31.01.2013
```

Расширенный поиск работает для следующих полей, общих для всех заказчиков:

Название поля	Формат	Описание	Пример
historicalId	число	ИД контрагента	? PhysicalParty.historicalId = 2051
actualityDate	дд.мм.гггг	Дата актуальности реквизитов (число, месяц, год полностью)	? PhysicalParty.actualityDate = 17.12.2019
startDate	дд.мм.гггг	Дата обновления (число, месяц, год полностью)	? PhysicalParty.startDate = 22.02.2023
mergedStatus	Статус по слиянию (литера)	Статус контрагента по слиянию	? PhysicalParty.mergedStatus = ORIGINAL
mergeAuthor	строка	Перечисление записей исходных систем, на основе которых был сформирован данный контрагент	? PhysicalParty.mergeAuthor = "AL:13" ? PhysicalParty.mergeAuthor = "AL:138, AL:20150330, BT:138, BT:20150330"  Ищет только по полному соответствию, если написать не все системы — уже не найдёт.
latestActualityDate	дд.мм.гггг	Последняя (максимальная) дата актуальности контрагента (включая все его атрибуты)	? PhysicalParty.latestActualityDate = 07.07.2015
manualRequisite	Логический тип (литера)	Признак наличия реквизитов, отмеченных на ручную обработку	? PhysicalParty.manualRequisite = true
manualAddress	Логический тип (литера)	Признак наличия адресов, отмеченных на ручную обработку	? PhysicalParty.manualAddress = true
manualPhone	Логический тип (литера)	Признак наличия телефонов, отмеченных на ручную обработку	? PhysicalParty.manualPhone = false
manualEmail	Логический тип (литера)	Признак наличия email, отмеченных на ручную обработку	? PhysicalParty.manualEmail = true
manualDoc	Логический тип (литера)	Признак наличия документов, отмеченных на ручную обработку	? PhysicalParty.manualDoc = false
birthdate	дд.мм.гггг	Дата рождения (число, месяц, год полностью)	? PhysicalParty.birthdate = 25.07.1997
surname	строка	Фамилия	? PhysicalParty.surname = дубровина
name	строка	Имя	? PhysicalParty.name = Анна
patronymic	строка	Отчество	? PhysicalParty.patronymic = Николаевна
fullName	строка	ФИО одной строкой — работает как простой поиск, но только по полям ФИО  Указываются компоненты через пробел в скобках — в любом порядке	Аслан кызы • Назила • Викторовна  ? PhysicalParty.fullName = (Аслан Кызы Назила Викторовна)

			? PhysicalParty.fullName = (Аслан Назила Кызы Викторовна)
			? PhysicalParty.fullName = (Аслан)
			? PhysicalParty.fullName = (Аслан Кызы)
			? PhysicalParty.fullName = (Назила Кызы)
			? PhysicalParty.fullName = (Викторовна)
			? PhysicalParty.fullName = (Викторовна кызы)
sourceId	строка	Код исходной системы и идентификатор в исходной системе	? PhysicalParty.sourceId = "SIBL:1-E45GZ-33"
snils	строка	СНИЛС	718-906-760 36
			? PhysicalParty.snils = "718-906-760 36"
			Также возможен поиск в произвольном виде
			? PhysicalParty.snils = 71890676036
			? PhysicalParty.snils = "718-906-760 36"
			? PhysicalParty.snils = "718 906 760 36"
			? PhysicalParty.snils = 718*906*760*36
			Требует дополнительной настройки.
sourceSystem	строка	Код исходной системы	? PhysicalParty.sourceSystem = "SIBL"

# ПОИСК ПО РЕКВИЗИТАМ ЮРЛИЦА

Расширенный поиск работает для следующих полей, общих для всех заказчиков:

Название поля	Формат	Описание	Пример
historicalId	число	НID контрагента	? LegalParty.historicalId = 841167
actualityDate	дд.мм.гггг	Дата актуальности реквизитов (число, месяц, год полностью)	? LegalParty.actualityDate = 11.10.2021
startDate	дд.мм.гггг	Дата обновления (число, месяц, год полностью)	? LegalParty.startDate = 21.10.2022
mergedStatus	Статус по слиянию (литера)	Статус контрагента по слиянию	? LegalParty.mergedStatus = RESULT
mergeAuthor	строка	Перечисление записей исходных систем, на основе которых был сформирован данный контрагент	? LegalParty.mergeAuthor = "AL:447" ? LegalParty.mergeAuthor = "AL:140930-1314, AL:140930-1315, AL:3.17-2"  Ищет по полному соответствию, если оставить только 2 системы уже не найдет.
latestActualityDate	дд.мм.гггг	Последняя (максимальная) дата актуальности контрагента, включая все его атрибуты (актуальность данных)	? LegalParty.latestActualityDate = 01.01.2012
manualRequisite	Логический тип (литера)	Признак наличия реквизитов, отмеченных на ручную обработку	? LegalParty.manualRequisite = true
manualAddress	Логический тип (литера)	Признак наличия адресов, отмеченных на ручную обработку	? LegalParty.manualAddress = false
manualPhone	Логический тип (литера)	Признак наличия телефонов, отмеченных на ручную обработку	? LegalParty.manualPhone = true
manualEmail	Логический тип (литера)	Признак наличия email, отмеченных на ручную обработку	? LegalParty.manualEmail = true
manualDoc	Логический тип (литера)	Признак наличия документов, отмеченных на ручную обработку	? LegalParty.manualDoc = false
sourceId	строка	Код исходной системы и идентификатор в исходной системе	? LegalParty.sourceId= "SIBL:1-317X5H4A"
sourceSystem	строка	Код исходной системы	? LegalParty.sourceSystem = "SIBL"

# ПОИСК ПО АДРЕСАМ

*Пример 1. Клиенты, у которых есть домашний адрес:*

? Address.type = HOME



Регистр в названиях полей очень важен.

Запрос "? Address.type=WORK" будет отлично работать, а вот "? address.Type=WORK" будет возвращать пустой результат.

*Пример 2. Клиенты с адресом, где не указана квартира:*

? Address.qualityCode = UNDEF\_05



**Важное ограничение**

Задать два условия сразу на один атрибут невозможно.

Запрос

```
? Address.type = HOME AND Address.qualityCode = UNDEF_05
```

найдет всех клиентов, у которых есть домашний адрес и есть адрес без квартиры. При этом совсем необязательно, что это будет один и тот же адрес.

Т.е. по этому запросу найдутся те, у кого есть любой домашний адрес и, например, рабочий адрес без квартиры.

Расширенный поиск работает для следующих полей адреса:

Название поля	Формат	Описание	Пример
historicalId	число	HID адреса	? Address.historicalId = 782363
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор контрагента)	? Address.mergeAuthor = "AL:229"
actualityDate	дд.мм.гггг	Дата актуальности (найдет КА, если хоть у одной исходной карточки в группе будет такая дата актуальности адреса)	? Address.actualityDate = 03.03.2014
qualityCode	Код полноты адреса (литера)	Код полноты адреса	? Address.qualityCode = GOOD
validationCode	Код валидации адреса (литера)	Код проверки адреса	? Address.validationCode = NOT_VALIDATED_HAS_AMBI
type	Тип адреса (литера)	Тип адреса	? Address.type = WORK

primary	Логический тип (литера)	Признак основного адреса	? Address.primary = true
postalCode	строка	Почтовый индекс	? Address.postalCode = 664035
country	строка	Наименование страны	? Address.country = "Беларусь"
district	строка	Федеральный округ	? Address.district = "Сибирский"
region	строка	Наименование региона	? Address.region = "Брянская"
rayon	строка	Наименование района	? Address.rayon = "Ракитянский"
city	строка	Наименование города	? Address.city = "Москва"
settlement	строка	Наименование населенного пункта	? Address.settlement = "Глинки"
street	строка	Наименование улицы	? Address.street = "Ленина"
houseNumber	число	Номер дома	? Address.houseNumber = 111
flat	число	Номер квартиры	? Address.flat = 111
flat2	число	Номер второй квартиры	? Address.flat2 = 2
popularity	Популярность атрибута (литера)	Популярность	? Address.popularity = POPULAR
fiasLevel	Уровень ФИАС (литера)	Код соответствия ФИАС	? Address.fiasLevel = FIAS_CITY
postponed	Признак отложенности (литера)	Признак отложенного атрибута	? Address.postponed = POSTPONED
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута (списки причин отложения и редактирования настраиваются индивидуально)	? Address.postponeReason = UNKNOWN Отложены без причины

# ПОИСК ПО ТЕЛЕФОНАМ

## Пример 1. Клиенты с мобильным телефоном

? Phone.cityCode = 916



Регистр в названиях полей очень важен.

Запрос "? Phone.cityCode = 916" будет отлично работать, а вот "? phone.citycode = 916" будет возвращать пустой результат.

## Пример 2. Клиенты с телефоном, помеченным, как "основной"

? Phone.primary = TRUE

## Пример 3. Клиенты с телефоном, где код "916".

? Phone.cityCode = 916



### Важное ограничение

Задать несколько условий сразу на один атрибут невозможно.  
Запрос

```
? Phone.type = HOME and Phone.primary = TRUE and Phone.cityCode = 916
```

найдет всех клиентов, у которых есть домашний телефон, есть телефон с признаком "Основной" и есть телефон с кодом "916". И это могут быть три **разных** телефона одного клиента, а не один и тот же.

По запросу выше найдется как Иванов Петр Петрович с телефоном "HOME, primary : +7-916-098-09-08"

Так и Семенов Игорь Васильевич, у которого есть следующие три телефона:

- HOME: +7-499-126-02-02
- MOBILE, primary: +7-986-109-00-99
- MOBILE: +7-916-123-45-67

Расширенный поиск работает для следующих полей телефона:

Название поля (регистр важен)	Формат	Описание	Пример
historicalId	число	ИД телефона	? Phone.historicalId = 53820
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)	? Phone.mergeAuthor = "AL:1801202"

actualityDate	дд.мм.гггг	Дата актуальности	? Phone.actualityDate = 22.02.2012
primary	Логический тип (литера)	Признак основного телефона	? Phone.primary = true
qualityCode	Код качества телефона (литера)	Код качества телефона	? Phone.qualityCode = GOOD_CITY_CONFLICT Конфликт по городу
type	Тип телефона (литера)	Тип телефона	? Phone.type = mobile
countryCode	строка	Код страны	? Phone.countryCode = 375
cityCode	строка	Код города	? Phone.cityCode = 495
number	строка	Номер телефона без кода города и страны	? Phone.number = 7821976
numberProfile	Характеристика номера телефона (литера)	Характеристика номера телефона	? Phone.numberProfile = BUSINESS Колл-центр
aggregateNumber	строка	Агрегированный номер (набор форматов телефона), состоит из полей : <ul style="list-style-type: none"> <li>• Тип телефона (опционально)</li> <li>• Код страны (опционально)</li> <li>• Код города + номер</li> <li>• Код страны (код города, тип телефона) + номер + добавочный номер</li> </ul> <b>aggregateNumber</b> является частью <b>fullPhone</b>	? Phone.aggregateNumber = 4951850968 ? Phone.aggregateNumber = 84951850968 ? Phone.aggregateNumber = 74951850968 ? Phone.aggregateNumber = HOME.74954113315 ? Phone.aggregateNumber = HOME.84954113315 ? Phone.aggregateNumber = HOME.4954113315 ? Phone.aggregateNumber = WORK.74959374477#4528 ? Phone.aggregateNumber = WORK.4959374477#4528
fullPhone	строка	Может состоять из: <ul style="list-style-type: none"> <li>• Полный номер одной строкой (плюс добавочный)</li> </ul>	? Phone.fullPhone = 9374477#4528

	<ul style="list-style-type: none"> <li>Только номер (код города или страны)</li> <li>Код страны (код города) + номер + добавочный номер</li> <li>Тип телефона + номер + код города (код страны). Отдельно тип телефона+ номер или тип телефона+код города не работает.</li> </ul>	<p>? Phone.fullPhone = 74959374477#4528</p> <p>? Phone.fullPhone = OTHER.74955544333</p> <p>? Phone.fullPhone = OTHER.4955544333</p>	
	<p>Отличие от <b>aggregateNumber</b>:</p> <p>Можно искать отдельно</p> <ul style="list-style-type: none"> <li>по коду города</li> <li>коду страны</li> <li>номеру без кода страны и города</li> <li>код страны + код города (через пробел)</li> <li>код страны + код города + телефон (через пробел)</li> </ul>	<p>? Phone.fullPhone = WORK. 74959374477#4528</p> <p>? Phone.fullPhone = 74955544333</p> <p>? Phone.fullPhone = 4955544333</p>	
		<p>Отличие от <b>aggregate Number</b>:</p> <p>? Phone.fullPhone = 495</p> <p>? Phone.fullPhone = 7</p> <p>? Phone.fullPhone = 7 495</p> <p>? Phone.fullPhone = 5544333</p>	
popularity	Код популярности (литера)	Популярность	? Phone.popularity = NONE
postponed	Признак отложенности (литера)	Признак отложенного атрибута	? Phone.postponed = POSTPONED
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута (списки причин отложения и редактирования настраиваются индивидуально)	? Phone.postponeReason = NEED_DOCS  Причина: нужны документы

#### Пример 4. Будут найдены все телефоны с кодом города 495

-- Запросы вернут одинаковый результат  
 ? Phone.fullPhone = 495  
 ? Phone.cityCode = 495

#### Пример 5. Телефон без кода города

? Phone.fullPhone = 7821976  
 ? Phone.number = 7821976

#### Пример 6. Все телефоны с кодом страны 375 (код Беларуси)

? Phone.fullPhone = 375  
 ? Phone.countryCode = 375



В **fullPhone** можно вводить:

- Только номер, или код города, или код страны
- Код страны (опционально) + код города + номер + добавочный номер (через # или \*)
- Тип телефона + код страны (опционально) + код города + номер

Тогда как **aggregateNumber** это составной номер. Содержит в себе два варианта написания телефона: с кодом страны (73439420592) и без него (3439420592).

Можно искать отдельно по типу, коду города и номеру через разные поля:

### Пример 7

-- Будут найдены контрагенты, у которых среди всех его телефонов есть хоть один мобильный, хоть один телефон с кодом города "495" и хоть один телефон с номером "5383254"  
? Phone.type = MOBILE and Phone.cityCode = 495 and Phone.number=5383254

А можно использовать **fullPhone** или **aggregateNumber** для нахождения конкретного телефона нужного типа:

### Пример 8

-- Запросы вернут одинаковый результат  
? Phone.fullPhone = MOBILE.9557752550  
? Phone.aggregateNumber = MOBILE.9557752550

### Пример 9

? Phone.number=5544333  
? Phone.fullPhone=5544333

# ПОИСК ПО EMAIL

## Пример 1. Клиенты, у которых есть рабочая электронная почта

? Email.type = WORK



Регистр в названиях полей очень важен.

Запрос "? Email.type=WORK" будет отлично работать, а вот "? email.Type=WORK" будет возвращать пустой результат.

## Пример 2. Клиенты, у которых есть электронная почта на Яндексе.

? Email.email= \*@yandex.ru



### Важное ограничение

Задать несколько условий сразу на один атрибут невозможно.

Запрос

```
? Email.email= *@yandex.ru and Email.type = WORK
```

найдет всех клиентов, у которых есть рабочий адрес электронной почты и есть электронная почта на Яндексе. И это могут быть **два разных** адреса одного клиента, а не один и тот же.

По запросу выше найдется как Иванов Петр Петрович с email "**WORK: petr@yandex.ru**"

Так и Семенов Игорь Васильевич, у которого есть электронные адреса:

- **WORK: igor@hflabs.ru**
- **HOME: igorechek@yandex.ru**

Расширенный поиск работает для следующих полей email:

Название поля (регистр важен)	Формат	Описание	Пример
historicalId	число	HID email	? Email.historicalId = 112374
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)	? Email.mergeAuthor = "AL:1412546425"
actualityDate	дд.мм.гггг	Дата актуальности	? Email.actualityDate = 04.04.2014
primary	Логический тип (литера)	Признак основного адреса электронной почты	? Email.primary = true
qualityCode	Код качества email (литера)	Код качества адреса электронной почты	? Email.qualityCode = CONFIRMED_MANUALLY

			Подтверждено контролером
type	Тип email	Тип адреса электронной почты	? Email.type = WORK
email	строка	Адрес электронной почты	? Email.email = test@test.ru
popularity	Код популярности (литера)	Популярность	? Email.popularity = NONE
postponed	Признак отложенности (литера)	Признак отложенного атрибута	? Email.postponed = normal
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута (списки причин отложения и редактирования настраиваются индивидуально)	? Email.postponeReason = UNKNOWN  Отложены без причины

# ПОИСК ПО ДОКУМЕНТАМ

- Документ, удостоверяющий личность (Passport)
  - Пример 1. Клиенты, у которых есть водительское удостоверение
  - Пример 2. Клиенты, у которых есть документ с серией "50 05"
  - Пример 3. Клиенты, у которых дата выдачи документа больше или равна 01.04.2017
  - Пример 4. Клиенты, у которых есть паспорт РФ с серией "45 01" и номером "910786"
  - Пример 5. Клиенты, у которых есть паспорт РФ с серией "45 01" / номером "910786"
  - Пример 6. Клиенты, у которых есть документ с серией "45 01" и номером "910786"
  - Пример 7. Клиенты, у которых есть документ с серией "45 01" / номером "910786"
- Документ индивидуального предпринимателя (IndividualPartyDocument)
  - Пример 8. ИП, у которых есть документ о дате регистрации до 01.01.2010
- Документ юридического лица (LegalPartyDocument)
  - Пример 9. ЮЛ, у которых есть документы с плохим кодом качества
- Лицензия (License)
  - Пример 10. ЮЛ, у которых есть лицензия с указанным номером
- Документ транспортного средства (VehicleDocument)
  - Пример 11. ТС с указанной серией

## Документ, удостоверяющий личность (Passport)

### Пример 1. Клиенты, у которых есть водительское удостоверение

```
? Passport.qualifierType = DRIVING_LICENSE
```



Регистр в названиях полей очень важен.

Запрос `? Passport.qualifierType = DRIVING_LICENSE` будет отлично работать, а вот `? passport.qualifierType = DRIVING_LICENSE` будет возвращать пустой результат.

### Пример 2. Клиенты, у которых есть документ с серией "50 05"

```
? Passport.documentSeries = "50 05"
```



#### Важное ограничение

Задать два условия сразу на один атрибут невозможно.

Запрос

```
? Passport.qualifierType = DRIVING_LICENSE and Passport.documentSeries = "50 05"
```

найдет всех клиентов, у которых есть права и документ с серией 50 05. При этом совсем необязательно, что серия "50 05" будет у водительского удостоверения. Т.е. по этому запросу найдутся те, у кого есть права (с любой серией) и одновременно есть документ с серией "50 05" (могут быть те же права, а может и паспорт)

### Пример 3. Клиенты, у которых дата выдачи документа больше или равна 01.04.2017

**? Passport.issueDate >= 01.04.2017**

Расширенный поиск работает для следующих полей документов:

Название поля (регистр важен)	Формат	Описание	Пример
historicalId	число	НID документа	? Passport.historicalId = 9317136
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)	? Passport.mergeAuthor = "AL:1111"
actualityDate	дд.мм.гггг	Дата актуальности	? Passport.actualityDate > 21.01.2017
primary	Логический тип (литера)	Признак основного документа	? Passport.primary = true
qualifierType	Тип документа ФЛ (лите ра)	Тип документа	? Passport.qualifierType = PASSPORT_RU
qualityCode	Коды качества полей документа ФЛ (ДУЛ) (литера)	Код качества документа	? Passport.qualityCode = EDITED
documentSeries	строка	Серия документа	? Passport.documentSeries = 45*01 ? Passport.documentSeries = "45 01"
documentNumber	строка	Номер документа	? Passport.documentNumber = 910786
fullDocument	строка	Поле для поиска документа, состоит из полей: <ul style="list-style-type: none"> <li>• место выдачи</li> <li>• тип документа (опционально)</li> <li>• серия и/или номер</li> <li>• код подразделения</li> </ul>	? Passport.fullDocument = "ОВД КИРОВСКОГО РАЙИСПОЛКОМА Г. КУЙБЫШЕВА" ? Passport.fullDocument = PASSPORT_RU.4511271447 ? Passport.fullDocument = PASSPORT_RU.271447 ? Passport.fullDocument = PASSPORT_RU.4511 ? Passport.fullDocument = 4511271447 ? Passport.fullDocument = 271447 ? Passport.fullDocument = 4511 ? Passport.fullDocument = "770-111"

issueDate	дд.мм.гггг	Дата выдачи документа	? Passport.issueDate > 05.02.2015
expiryDate	дд.мм.гггг	Дата истечения срока действия	? Passport.expiryDate > 05.02.2015
invalidStatus	Статус недействительности для документа (литера)	Статус недействительности для документа	? Passport.invalidStatus = ACTUAL
popularity	Код популярности (литера)	Популярность	? Passport.popularity = POPULAR
postponed	Признак отложенности (литера)	Признак отложенного атрибута	? Passport.postponed = NORMAL
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута	? Passport.postponeReason = DOCS_IN_ARCHIVE

Для поиска КА с определенным типом документа используется **Passport.qualifierType**.

Искать можно как отдельно по серии и номеру (**documentSeries** и **documentNumber**), так и через **fullDocument** — это поле сделано для удобства, чтобы можно было вводить тип (опционально), серию и/или номер единой строкой.

В поле **fullDocument** можно искать:

- Только серию, или номер, или код подразделения, или кем выдан. Поиск идет по полному совпадению с базой — в кавычках и с пробелами
- Номер и/или серию без пробелов
- Типу документа + его серия и/или номер без пробелов. Между типом и номером/серией разделитель точка.

Примеры поиска по одному и тому же документу:

#### *Пример 4. Клиенты, у которых есть паспорт РФ с серией "45 01" и номером "910786"*

-- Поиск по полному документу: (тип + номер + серия)  
? Passport.fullDocument = PASSPORT\_RU.4501910786

#### *Пример 5. Клиенты, у которых есть паспорт РФ с серией "45 01" / номером "910786"*

-- Поиск по полному документу: тип + номер/серия (без пробелов)  
? Passport.fullDocument = PASSPORT\_RU.4501  
? Passport.fullDocument = PASSPORT\_RU.910786

#### *Пример 6. Клиенты, у которых есть документ с серией "45 01" и номером "910786"*

-- Поиск по полному документу: номер + серия (без пробелов)  
? Passport.fullDocument = 4501910786

## Пример 7. Клиенты, у которых есть документ с серией "45 01" / номером "910786"

-- Поиск по полному документу: номер/серия  
? Passport.fullDocument = "45 01"  
? Passport.fullDocument = 4501  
? Passport.fullDocument = 910786

Другие варианты поиска:

-- Поиск по "кем выдан" (орган, выдавший документ)  
? Passport.fullDocument = "ОВД КИРОВСКОГО РАЙИСПОЛКОМА Г. КУЙБЫШЕВА"  
  
-- как НЕ НАДО делать: Поиск отдельно по серии / номеру  
? Passport.documentSeries = "45 01" and Passport.documentNumber = 910786  
т.к. в этом случае не гарантируется, что серия и номер будут относиться к одному документу.

## Документ индивидуального предпринимателя (IndividualPartyDocument)

### Пример 8. ИП, у которых есть документ о дате регистрации до 01.01.2010

? IndividualPartyDocument.registrationDate < 01.01.2010

Расширенный поиск работает для следующих полей документов ИП:

Название поля	Формат	Описание
historicalId	число	ИД документа
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)
actualityDate	дд.мм.гггг	Дата актуальности
primary	Логический тип (литера)	Признак основного документа
qualifierType	Тип документа ИП (литера)	Тип документа
qualityCode	Код качества документа (литера)	Код качества документа
documentSeries	строка	Серия документа
documentNumber	строка	Номер документа
registrationDate	дд.мм.гггг	Дата регистрации
postponed	Признак отложенности (литера)	Признак отложенного атрибута
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута

## Документ юридического лица (LegalPartyDocument)

### Пример 9. ЮЛ, у которых есть документы с плохим кодом качества

? LegalPartyDocument.qualityCode = NOT\_SURE

Расширенный поиск работает для следующих полей документов:

Название поля	Формат	Описание
historicalId	число	ИД документа
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)
actualityDate	дд.мм.гггг	Дата актуальности
primary	Логический тип (литера)	Признак основного документа
qualifierType	Тип документа ЮЛ (литера)	Тип документа
qualityCode	Код качества документа (литера)	Код качества документа
documentSeries	строка	Серия документа
documentNumber	строка	Номер документа
registrationDate	дд.мм.гггг	Дата регистрации
postponed	Признак отложенности (литера)	Признак отложенного атрибута
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута

## Лицензия (License)

### Пример 10. ЮЛ, у которых есть лицензия с указанным номером

? License.documentNumber = 000151

Расширенный поиск работает для следующих полей лицензии:

Название поля	Формат	Описание
historicalId	число	ИД документа
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)
actualityDate	дд.мм.гггг	Дата актуальности
primary	Логический тип (литера)	Признак основного документа
qualifierType	Тип лицензии (литера)	Тип документа



qualityCode	Код качества документа (литера)	Код качества документа
documentSeries	строка	Серия документа
documentNumber	строка	Номер документа
validityPeriod	строка	Срок действия
issueDate	дд.мм.гггг	Дата выдачи документа
postponed	Признак отложенности (литера)	Признак отложенного атрибута
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута

## Документ транспортного средства (VehicleDocument)

### Пример 11. ТС с указанной серией

? VehicleDocument.documentSeries = CC

Расширенный поиск работает для следующих полей документов транспортного средства:

Название поля	Формат	Описание
historicalId	число	ИД документа
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор клиента)
actualityDate	дд.мм.гггг	Дата актуальности
primary	Логический тип (литера)	Признак основного документа
qualifierType	Тип документа ТС (литера)	Тип документа
qualityCode	Код качества документа (литера)	Код качества документа
documentSeries	строка	Серия документа
documentNumber	строка	Номер документа
licensePlate	строка	Государственный регистрационный знак ТС
postponed	Признак отложенности (литера)	Признак отложенного атрибута
postponeReason	Причина отложения атрибута (литера)	Причина отложения атрибута

# ПОИСК ПО ДУБЛИКАТАМ

Пример запроса (все дубликаты с коэффициентом соответствия между 80 и 90):

```
? Duplicate.matchScope >= 80 and Duplicate.matchScope <= 90
```

Расширенный поиск работает для следующих полей дубликатов:

Название поля	Формат	Описание	Пример
historicalId	число	НID дубликата	? Duplicate.historicalId = 42220
rule	строка	Идентификатор сценария	? Duplicate.rule = 1
matchScope	число	Коэффициент соответствия	? Duplicate.matchScope = 90
status	Статус пары дубликатов (литера)	Статус пары дубликатов	? Duplicate.status = REJECT Дубликаты отклонены
mergedStatus	Результат слияния (литера)	Результат слияния пары дубликатов	? Duplicate.mergedStatus = UNPROCESSED Дубликаты найдены, но требуется ручное рассмотрение
postponeReason	Причина отложения дубликата (литера)	Причина отложения для пары дубликатов	? Duplicate.postponeReason = UNKNOWN Отложены без причины
minPartyHid	число	Минимальный НID из пары дубликатов	? Duplicate.minPartyHid = 30519

# ПОИСК ПО ГРУППАМ ДУБЛИКАТОВ

Пример запроса (все группы дубликатов, в которых 5 и более контрагентов, причем не все пары дублей подтверждены):

```
? ManualDuplicateGroup.partyCount >= 5 and ManualDuplicateGroup.progress < 100
```

Найти группу дубликатов по исходной системе с указанием того, что на каждом конце связи должна быть одиночная карточка:

```
? Duplicate.sourceSystemPair = "BT!-GM!"  
Кавычки важны! Без них не работает.
```

Найти группы дубликатов для одиночных карточек из определенной системы:

```
? Duplicate.sourceSystem = "BT!"  
Кавычки важны! Без них не работает.
```

Расширенный поиск работает для следующих полей групп дубликатов:

Название поля	Формат	Описание	Пример
merged	true / false	Признак того, что группа уже объединена	? ManualDuplicateGroup.merged = false
partyCount	число	Количество контрагентов в группе	? ManualDuplicateGroup.partyCount >= 5
partyType	строка	Тип контрагента	? ManualDuplicateGroup.partyType = LEGAL
progress	число	Прогресс обработки группы (значение в пределах от 0 до 100)	? ManualDuplicateGroup.progress < 100
sourceSystem	строка	Исходная система	? Duplicate.sourceSystem = "BT!"
sourceSystemPair	строка	Исходная система для группы дубликатов	? Duplicate.sourceSystemPair = "BT!-GM!"

# ПОИСК ПО АККАУНТАМ

## Пример 1. Клиенты, у которых есть скайп

? Account.type = SKYPE

Расширенный поиск работает для следующих полей аккаунта:

Название поля	Формат	Описание	Пример
historicalId	число	ИД аккаунта	? Account.historicalId = 3041
mergeAuthor	строка	Первоисточник (код системы + исходный идентификатор контрагента)	? Account.mergeAuthor = "AL:136"
actualityDate	дд.мм.гггг	Дата актуальности	? Account.actualityDate = 01.09.2011
qualityCode	Код качества реквизита(литера)	Код качества аккаунта	? Account.qualityCode = EDITED
type	Тип аккаунта (литера)	Тип аккаунта	? Account.type = SKYPE
primary	Логический тип (литера)	Признак основного адреса	? Account.primary = true
account	строка	Значение аккаунта	? Account.account =test_account (имя аккаунта)

# ПОИСК ПО РЕКВИЗИТАМ ЛЮБОГО КОНТРАГЕНТА

Пример запроса (контрагенты, объединенные за 30.01.2013, показано на примере **PhysicalParty**, но работает для любого типа контрагентов):

```
? PhysicalParty.mergedStatus = RESULT and PhysicalParty.startDate >= 30.01.2013 and PhysicalParty.startDate < 31.01.2013
```

Расширенный поиск работает для следующих полей, общих для всех заказчиков:

Название поля	Формат	Описание	Пример
historicalId	число	ИД контрагента	? PhysicalParty.historicalId = 2051
actualityDate	дд.мм.гггг	Дата актуальности реквизитов	? PhysicalParty.actualityDate = 17.12.2019
startDate	дд.мм.гггг	Дата обновления	? PhysicalParty.startDate = 22.02.2023
mergedStatus	Статус по слиянию (литера)	Статус контрагента по слиянию	? PhysicalParty.mergedStatus = ORIGINAL
mergeAuthor	строка	Перечисление записей исходных систем, на основе которых был сформирован данный контрагент	? PhysicalParty.mergeAuthor = "AL:13"
latestActualityDate	дд.мм.гггг	Последняя (максимальная) дата актуальности контрагента (включая все его атрибуты)	? PhysicalParty.latestActualityDate = 07.07.2015
manualRequisite	Логический тип (литера)	Признак наличия реквизитов, отмеченных на ручную обработку	? PhysicalParty.manualRequisite = true
manualAddress	Логический тип (литера)	Признак наличия адресов, отмеченных на ручную обработку	? PhysicalParty.manualAddress = true
manualPhone	Логический тип (литера)	Признак наличия телефонов, отмеченных на ручную обработку	? PhysicalParty.manualPhone = false
manualEmail	Логический тип (литера)	Признак наличия email, отмеченных на ручную обработку	? PhysicalParty.manualEmail = true
manualDoc	Логический тип (литера)	Признак наличия документов, отмеченных на ручную обработку	? PhysicalParty.manualDoc = false
sourceId	строка	Код исходной системы и идентификатор в исходной системе	? PhysicalParty.sourceId = "SIBL:1-E45GZ-33"

# ОПИСАНИЕ КОДОВ КАЧЕСТВА

- Коды качества полей документа ФЛ (ДУЛ)
  - Код качества документа ФЛ
  - Маркеры проверок документа
  - Статус актуальности документа
  - Отображение гранулярных кодов качества и маркеров ДУЛ
  - Статус недействительности для документа
  - Скоринг полноты ДУЛ
- Коды качества email
  - Код качества адреса электронной почты
  - Маркеры проверок и изменений email
  - Причины признания email плохим
- Коды качества адреса
  - Код валидации адреса
  - Код качества координат
  - Код полноты адреса
  - Маркер изменения адреса
  - Маркер невалидности адреса
  - Нижний уровень иерархии адреса
  - Соответствие уровню иерархии в ФИАС
- Коды качества телефона
  - Код качества телефона
  - Маркеры проверок и изменений телефона
  - Причины, почему телефон не распознан
- Коды качества ФИО и пола
  - Общий код качества ФИО
  - Статус распознавания ФИО
  - Причины отправки ФИО на ручной разбор
  - Маркеры проверки и изменения ФИО
  - Маркеры проверки и изменения фамилии
  - Маркеры проверки и изменения имени
  - Маркеры проверки и изменения отчества
  - Код качества пола
- Прочие коды качества полей
  - Код качества VIN-номера
  - Код качества ИНН
  - Код качества КИО

Код качества КПП  
Код качества ОГРН  
Код качества ОКВЭД  
Код качества ОКПО  
Код качества СНИЛС  
Код качества даты рождения  
Подробный код качества даты рождения  
Код качества наименования ЮЛ  
Популярность атрибута  
Признак террориста

# КОДЫ КАЧЕСТВА ПОЛЕЙ ДОКУМЕНТА ФЛ (ДУЛ)

- Код качества документа ФЛ
- Маркеры проверок документа
- Статус актуальности документа
- Отображение гранулярных кодов качества и маркеров ДУЛ
- Статус недействительности для документа
- Скоринг полноты ДУЛ



# КОД КАЧЕСТВА ДОКУМЕНТА ФЛ

Коды качества по убыванию приоритета (код качества в начале списка «лучше» кода качества в конце списка)

№	Литера	Локализация	Описание	Исходно	Стандартизовано	Описание примера стандартизации
1	<b>Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную внес изменения о документе ФЛ и/или подтвердил данные	тип: паспорт РФ серия: 4014 номер: 78543	тип: паспорт РФ серия: 4014 номер: 785436 дата выдачи: 27.08.2014	Любые данные, подтвержденные контролером, имеют статус «CONFIRMED_MANUALLY». Даже если у них не заполнены обязательные поля
3	EDITED	Правильное значение	Данные о документе прошли все проверки, документ корректный	тип: паспорт РФ номер: 4014785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 4014 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Документ корректный. Серию и номер паспорта можно указать в одном поле, система сама разделит их на отдельные составляющие
4	EDITED_NOT_CHECK	Правильное значение, без сверки с ДР	Правильное значение, не получилось сверить с датой рождения из-за пустой или некорректной даты рождения	тип: паспорт РФ номер: 4014785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-	тип: паспорт РФ серия: 4014 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Правильные данные паспорта гражданина РФ, но у лица отсутствует или некорректная дата рождения, указанная в личной карточке

				Петербургу Дата рождения клиента не заполнена		
5	EDITED_PARTIAL_REQ	Правильное значение, есть незаполненные поля	Данные о документе корректные, но имеются незаполненные поля «кем выдан», «код подразделения»	тип: паспорт РФ номер: 4014785436 дата выдачи: 27.08.2014 код подразделения: (пустое поле) кем выдан: (пустое поле) выдан: (пустое поле)	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 27.08.2014 код подразделения: (пустое поле) кем выдан: (пустое поле)	Правильные данные паспорта гражданина РФ, но есть незаполненные поля «кем выдан», «код подразделения»
<b>6 На ручную обработку</b>						
7	NOT_SURE	Некорректный документ	Некорректный документ (без специфичного кода качества). Литера проставляется в случае, если причина некорректности документа не покрыта описанными ниже проверками			
8	BAD_TYPE	Неизвестный тип документа	Указанный тип документа отсутствует в справочнике типов документов, удостоверяющих личность	тип: паспорт гражданина Азербайджана номер: 4014785436	тип: паспорт гражданина Азербайджана номер: 4014785436	Тип документа «паспорт гражданина Азербайджана» отсутствует в справочнике типов ДУЛ. Для иностранных паспортов необходимо указывать тип «Паспорт иностранного гражданина»
9	BAD_REQUIRED	Не все обязательные поля заполнены	Заполнены не все поля, настроенные как обязательные для данного типа документа	тип: паспорт РФ номер: 4014785436 дата выдачи: (пустое поле)	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: (пустое поле)	Обязательное поле для паспорта РФ «дата выдачи» не заполнено
10	BAD_SERIES_NUMBER		Значение серии/номера или некорректное для данного типа документа	тип: паспорт РФ	тип: паспорт РФ серия: 45 06 номер: 54343	Некорректный формат номера паспорта (серия

		Пустая или не корректная серия/номер документа		номер: 4 50654343		и номер паспорта должны состоять из 10 арабских цифр)
11	AGE_ISSUE_CLASH	Документ не соответствует возрасту	Дата выдачи документа не соответствует возрасту лица, т.к. документ выдается с определенного возраста	тип: паспорт РФ номер: 7 31122 3565 <u>дата</u> <u>выдачи:</u> <u>21.08.2011</u>  <u>дата</u> <u>рождения</u> <u>я лица:</u> <u>0</u> <u>1.06.2010</u>	тип: паспорт РФ серия: 73 11 номер: 223565 <u>дата выдачи:</u> <u>21.08.2011</u>  <u>дата</u> <u>рождения</u> <u>лица:</u> <u>01.06.2010</u>	Дата выдачи документа «21.08.2011» не соответствует возрасту лица (дата рождения лица «01.06.2010»). Первый паспорт гражданина РФ может быть получен в случае достижения 14-летнего возраста
12	CODE_SERIES_CLASH	Конфликт серии с кодом подразделения	Регион, определенный по коду подразделения паспорта РФ, не соответствует региону, определенному по серии документа	тип: паспорт РФ <u>номер:</u> <u>73</u> <u>11223565</u> дата выдачи: 21.08.2011 <u>код</u> <u>подразделения:</u> <u>392-008</u>	тип: паспорт РФ <u>серия:</u> <u>73 11</u> номер: 223565 дата выдачи: 21.08.2011 <u>код подразделения:</u> <u>39</u> <u>2-008</u>	Первые две цифры серии (73 11) — код ОКАТО региона (Ульяновская область), который не соответствует региону, определенному по первым двум цифрам кода подразделения (39 2-008 — Калининградская область)
13	BAD_ISSUE_DATE	Плохая дата выдачи	Дата выдачи пустая, из будущего, или сильно в прошлом (не входит в диапазон, когда выдавались документы такого типа)	тип: паспорт РФ номер: 4 014785436 дата выдачи: 27.08.3314	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 27.08.3314	На момент 2021 года дата выдачи паспорта «27.08.3314» является датой из будущего
14	SERIES_ISSUE_CLASH	Год печати бланка из будущего или конфликтует с датой выдачи паспорта	Год печати бланка, определенный по серии паспорта, из будущего, или не соответствует году выдачи паспорта (допустимое отклонение -3 года и +5 лет)	тип: паспорт РФ <u>номер:</u> <u>40147854</u> <u>36</u> <u>дата</u> <u>выдачи:</u> <u>24.10.2007</u>	тип: паспорт РФ <u>серия:</u> <u>40 14</u> номер: 785436 <u>дата выдачи:</u> <u>24.10.2007</u> <u>07</u>	Год выдачи паспорта «24.10.2007» не соответствует году печати бланка (третья и четвертая цифры серии паспорта «14» — это год печати бланка). Год выпуска бланка может быть на 3 года меньше или на 5 лет больше даты выдачи документа


15	BAD_CODE	Плохой код подразделения	В коде подразделения отсутствует ровно шесть арабских цифр, или указан код подразделения несуществующего региона или уровня паспортного отдела (определяется по первой паре и третьей цифре, соответственно)	тип: паспорт РФ номер: 7311223565 дата выдачи: 21.08.2011 <u>код подразделения:</u> <u>39208</u>	тип: паспорт РФ серия: 73 11 номер: 223565 дата выдачи: 21.08.2011 <u>код подразделения:</u> <u>(пустое значение)</u>	Код подразделения не соответствует формату (допустимый формат — 6 арабских цифр)
16	BAD_EXPIRY_DATE	Плохая дата окончания документа	Дата окончания срока действия документа имеет неверный формат или меньше даты выдачи паспорта	тип: загранпаспорт РФ серия: 71 номер: 9999999 номер: 719999999 <u>Дата выдачи:</u> <u>27.08.2014</u> <u>Дата окончания срока действия:</u> <u>27.08.2014</u> <u>25.08.2000</u> <u>дата окончания срока действия:</u> <u>25.08.2000</u>	тип: загранпаспорт РФ серия: 71 номер: 9999999 <u>Дата выдачи:</u> <u>27.08.2014</u> <u>Дата окончания срока действия:</u> <u>25.08.2000</u>	Дата окончания срока действия загранпаспорта РФ меньше даты его выдачи. Покрытие документа проверками см. 3 десь
17	EXPIRY_LIMIT_CLASH	Дата окончания больше срока действия	Дата окончания действия документа больше максимально возможного срока действия документа, рассчитанного на основе даты выдачи	тип: загранпаспорт РФ серия: 71 номер: 719999999 <u>Дата выдачи:</u> <u>21.08.2011</u> <u>дата окончания срока действия:</u> <u>21.08.2050</u>	тип: загранпаспорт РФ серия: 71 номер: 9999999 <u>дата выдачи:</u> <u>21.08.2011</u> <u>дата окончания срока действия:</u> <u>21.08.2050</u>	Срок действия загранпаспорта нового образца — 10 лет (для серий 71, 72, 73, 75, 76, 80, 82)
18	MISSED	Неизвестный код качества	Атрибут не проходил стандартизацию			
<b>19 Устаревшие</b>						
20	SER_ISSUE_MATCH	Дата выдачи не соответствует серии (устар.)	Дата выдачи паспорта не соответствует серии документа более чем на 3 года	тип: паспорт РФ серия: 22 номер: 766901 серия: 2217 номер: 766901	тип: паспорт РФ серия: 22 17 номер: 766901 дата выдачи: 24.10.2007	Дата выдачи паспорта «24.10.2007» не соответствует году печати бланка (третья и четвертая цифры

				дата выдачи: 24.10.2007		серии паспорта «17» — это год печати бланка)
21	EMPTY_DATE	Пустая дата выдачи (устар.)	Пустая дата выдачи	тип: паспорт РФ номер: 4 014785436 <u>дата выдачи:</u> <u>(пустое поле)</u>	тип: паспорт РФ серия: 40 14 номер: 785436 <u>дата выдачи:</u> <u>(пустое поле)</u>	Пустая дата выдачи
22	BAD_DATE	Плохая дата выдачи (устар.)	Дата выдачи сильно в прошлом (не входит в диапазон, когда выдавались документы такого типа) или из будущего	тип: паспорт РФ номер: 4 014785436 дата выдачи: 27.08.4314	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 27.08.4314	На момент 2021 года дата выдачи паспорта «27.08.4314» является датой из будущего
23	DOCUMENT_EXPIRED	Документ просрочен (устар.)	Причины проставления кода качества: 1) текущая дата меньше или равна максимально возможному сроку действия документа, рассчитанного на основе даты выдачи; 2) дата окончания действия документа больше или равна текущей дате; 3) дата выдачи документа больше минимально допустимой даты выдачи документа такого типа по возрасту клиента, рассчитанному на основе даты рождения	тип: заграничный паспорт РФ номер: 7 19999999 дата выдачи: 21.08.2011 дата окончания действия: 21.08.2050	тип: заграничный паспорт РФ серия: 71 номер: 9999999 дата выдачи: 21.08.2011 дата окончания срока действия: 21.08.2050	Срок действия заграничного паспорта нового образца — 10 лет (для серий 71, 72, 73, 75, 76, 80, 82)
24	BAD_MANDATORY	Не заполнены все обязательные поля в паспорте (устар.)	В документе отсутствуют обязательные поля: серия/номер /дата выдачи/место выдачи (если проверка включена)	тип: паспорт РФ номер: 4 014785436 <u>дата выдачи:</u> <u>(пустое поле)</u>	тип: паспорт РФ серия: 40 14 номер: 785436 <u>дата выдачи:</u> <u>(пустое поле)</u>	Обязательное поле для паспорта РФ «дата выдачи» не заполнено

# ОТОБРАЖЕНИЕ КОДА КАЧЕСТВА ДОКУМЕНТА

## Отображение кода качества

Недействительность дописывается после кода качества и популярности:

Тип	Паспорт гражданина РФ ▾		
	<input type="checkbox"/> Основной		
Исходный	22 05•726110••••• 		
	<span style="color: red;">■</span> Правильное значение, популярный, недействительный		
Серия	22 05	Номер	726110

Статус «актуальный» не дописывается.

Цвет документа выбирается как наихудший из цвета кода качества, популярности и недействительности. Например, в примере выше он будет красный из-за «недействительный».

# МАРКЕРЫ ПРОВЕРОК ДОКУМЕНТА

Каждому полю документа соответствует определенный набор маркеров, и каждый маркер отображается (только если он проставлен) строго под формой, к которой он прикреплен.

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	<b>Серия (documentSeries)</b>					
2	EMPTY_SERIES	Серия пустая	Пустое значение в поле «серия»	тип: паспорт РФ серия: (пустое поле) номер: 785436 дата выдачи: 27.08.2014 код подразделения: 80-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: (пустое поле) номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Отсутствует серия паспорта гражданина РФ
3	BAD_SERIES_FORMAT	Серия не соответствует формату	Серия паспорта должна состоять ровно из 4 арабский цифр (допускается символ-разделитель между двумя парами цифр)	тип: паспорт РФ серия: 99 9 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 80-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 99 9 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Серия паспорта состоит из 3 арабских цифр, ожидается 4
4	BAD_SERIES_REGION	Регион отсутствует в ОКАТО	Первая пара цифр серии паспорта РФ, информирующих о регионе, в котором напечатан бланк, отсутствует в справочнике кодов ОКАТО	тип: паспорт РФ серия: 00 99 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 00 99 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Первые две цифры серии «00» отсутствуют в справочнике кодов ОКАТО

				код подразде ления: 7 80-121 кем выдан: УФМС по Санкт- Петербур гу		
5	FUTURE _SERIES	Несуществую щая серия (из будущего)	Вторая пара цифр серии паспорта РФ — год печати бланка, который может быть больше текущего года не более чем на 5 лет. На момент стандартизации данных год печати бланка больше текущего года более чем на 5 лет	тип: пас порт РФ <u>серия: 40</u> <u>89</u> номер: 7 85436 дата выдачи: 27.08.2014 код подразде ления: 7 80-121 кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ <u>серия: 40 89</u> номер: 785436 дата выдачи: 27.08.20 14 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Вторая пара цифр серии паспорта РФ — год печати бланка, который может быть больше текущего года не более чем на 5 лет. На момент 2021 года 2089 год является годом из будущего
6	NOT_VA LID_SE RIES	Невалидная серия	Для типов документов, у которых определены допустимые серии, значение не соответствует разрешенным	тип: заг ранпас порт РФ <u>номер:</u> <u>40 99999</u> <u>99</u> дата выдачи: 21.08.2011	тип: загранпаспорт РФ <u>серия: 40</u> номер: 9999999 дата выдачи: 21.08.2 011	Серия загранпаспорта РФ подходит по формату, но не по правилам нумерации серий для этого типа документов. Разрешен диапазон серии 50-98 согласно Постановлению Правительства РФ
7	SERIES _CHANGE	Серия была очищена от мусора	Возможные преобразования серии для паспорта РФ: очистка от мусора (знаков препинания); разделение серии и номера, если они пришли в одном поле; смена местами серии и номера, если были перепутаны	тип: пас порт РФ <u>номер:</u> <u>401/4785</u> <u>436</u> дата выдачи: 27.08.2014 код подразде ления: 7 80-121 кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ <u>серия: 40 14</u> номер: <u>785436</u> дата выдачи: 27.08.2 014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Серия и номер в исходной строке содержатся в одном поле. При стандартизации было выполнено разделение серии и номера на два поля. В поле серия удален мусорный символ



Номер (documentNumber)						
9	EMPTY_NUMBER	Номер пустой	Пустое значение в поле «номер» документа	тип: паспорт РФ серия: 4014 номер: (пустое поле) дата выдачи: 27.08.2014 код подразделения: 80-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 4014 номер: (пустое поле) дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Отсутствует номер паспорта гражданина РФ
10	BAD_NUMBER_FORMAT	Номер не соответствует формату	В поле «номер» значение недопустимого формата	тип: паспорт РФ серия: 4014 номер: 999 дата выдачи: 27.08.2014 код подразделения: 80-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 4014 номер: 999 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Номер паспорта гражданина РФ должен состоять из шести арабских чисел
11	NOT_VALID_NUMBER	Несуществующий номер	Номер не соответствует правилам нумерации серий для этого типа документов	тип: паспорт РФ серия: 4014 номер: 000085 дата выдачи: 27.08.2014 код подразделения: 80-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ сери: 4014 номер: 000085 дата выдачи: 27.08.2014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Номер паспорта гражданина РФ «000085» не соответствует правилам нумерации. Номера паспортов РФ должны быть в диапазоне от 000101 до 999999 (существуют редкие номера в диапазоне от 000001 до 000100)

Петербур гу						
12	NUMBER _CHANGE	Номер был очищен от мусора	Возможные преобразования номера паспорта РФ: оч истка от мусора (знаков препинания); разделение серии и номера, если они пришли в одном поле; смена местами серии и номера, если они были перепутаны	тип: пас порт РФ <u>номер: 4 0147... 85436</u> дата выдачи: 27.08.2014 код подразде ления: 7 80-121 кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ <u>серия: 40 14 номер: 785436</u> дата выдачи: 27.08.2 014 код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Номер паспорта гражданина РФ был разделен от серии и очищен от мусора
<b>13 Дата выдачи (issueDate)</b>						
14	EMPTY_ ISSUE_ DATE	Пустая дата выдачи	Пустое значение в поле «дата выдачи»	тип: пас порт РФ номер: 4 014785436 <u>дата выдачи: (пустое поле)</u> код подразде ления: 7 80-121 кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ серия: 40 14 номер: 785436 <u>дата выдачи: (пустое поле)</u> код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Пустое значение в поле «дата выдачи»
15	BAD_IS SUE_DA TE_FOR MAT	Недопустимы й формат даты	Несуществующая дата или неверный формат даты (когда дата задана в виде строки)	тип: пас порт РФ номер: 4 014785436 <u>дата выдачи: 27814</u> код подразде ления: 7 80-121 кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ номер: 4014785436 <u>дата выдачи: 27814</u> код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Указан недопустимый формат даты. Список форматов даты можно расширять необходимыми форматами

16	BAD_ISSUE_DATE	Недопустима я дата выдачи	Дата не входит в диапазон, когда выдавались документы такого типа	тип: паспорт РФ номер: 4014785436 <u>дата выдачи:</u> <u>01.01.1900</u> код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 40 14 номер: 785436 <u>дата выдачи:</u> <u>01.01.1900</u> код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	Дата выдачи паспорта РФ не входит в диапазон, когда выдавались документы такого типа (для паспорт РФ дата выдачи осуществляется с 1997 года). Для других документов: <ul style="list-style-type: none"> <li>• Заграничный паспорт РФ: с 1993 года;</li> <li>• Дипломатический паспорт: с 1993 года;</li> <li>• Служебный паспорт: с 1993 года;</li> <li>• Паспорт моряка: с 1997 года;</li> <li>• СНИЛС: с 1996 года;</li> <li>• СССР паспорт: до 1996 года включительно;</li> <li>• СССР загран паспорт: до 1992 года включительно;</li> <li>• СССР паспорт МинМорФлота: до 1996 года включительно.</li> </ul>
17	FUTURE_ISSUE_DATE	Дата выдачи из будущего	На момент стандартизации данных дата выдачи документа больше текущей даты	тип: паспорт РФ номер: 4014785436 <u>дата выдачи:</u> <u>21.08.2022</u> код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 40 14 номер: 785436 <u>дата выдачи:</u> <u>21.08.2022</u> код подразделения: 780-121 кем выдан: УФМС по Санкт-Петербургу	На момент 24.05.2022 года дата выдачи паспорта «21.08.2022» является датой из будущего
18	ISSUE_DATE_DOCUMENT_EXPIRED	Документ просрочен	На момент стандартизации данных текущая дата меньше или равна максимально возможному сроку действия документа, рассчитанному на основе даты выдачи	тип: загранпаспорт РФ номер: 719999999 <u>дата выдачи:</u> <u>02.08.2001</u> дата окончания срока	тип: загранпаспорт РФ серия: 71 номер: 9999999 <u>дата выдачи:</u> <u>02.08.2001</u> дата окончания срока действия: (пустое поле)	Документ просрочен, т.к. с момента выдачи загранпаспорта гражданина РФ прошло более 10 лет (срок действия заграничного паспорта составляет 10 лет). В примере дата окончания срока действия не заполнена. Если дата окончания срока действия документа будет указана больше максимально

действия:  
(пустое  
поле)

возможного срока действия  
документа, рассчитанного  
на основе даты выдачи, то  
будет указан другой маркер  
—  
«EXPIRY\_DATE\_LIMIT\_CLAS  
N». Срок действия  
документов:

- Заграничный паспорт РФ: 10 лет
- Дипломатический паспорт: 5 лет
- Служебный паспорт: 5 лет
- Паспорт моряка: 5 лет
- Разрешение на временное проживание: 3 года
- Удостоверение беженца: 3 года
- Водительское удостоверение: 10 лет
- Вид на жительство: 5 лет
- Свидетельство о предоставлении временного убежища: 1 год
- СССР заграничный паспорт: 5 лет
- СССР паспорт МинМорФлота: 5 лет

#### 19 Дата окончания (если предусмотрено для типа ДУЛ) (expiryDate)

20	EMPTY_EXPIRY_DATE	Пустая дата окончания документа	Пустое значение в поле «Истекает» для документа, где дата окончания является обязательным полем	тип: заграничный паспорт РФ номер: 719999999 дата выдачи: 21.08.2011 окончания срока действия: (пустое поле)	тип: заграничный паспорт РФ серия: 719999999 дата выдачи: 21.08.2011 <u>Дата окончания срока действия: (пустое поле)</u>	Пустое значение в поле «дата окончания срока действия»
21	BAD_EXPIRY_DATE	Дата окончания меньше даты выдачи	Дата окончания срока действия документа меньше даты выдачи	тип: заграничный паспорт РФ номер: 719999999 дата выдачи: 21.08.2011	тип: заграничный паспорт РФ серия: 719999999 дата выдачи: 21.08.2011	Заполненная дата окончания заграничного паспорта РФ меньше даты выдачи

			<u>дата</u> <u>окончания</u> <u>срока</u> <u>действия:</u> <u>21.08.20</u> <u>09</u>	<u>дата окончания</u> <u>срока действия:</u> <u>21.08</u> <u>2009</u>		
22	BAD_EX PIRY_D ATE_FO RMAТ	Недопустимый формат даты	Несуществующая дата или неверный формат даты (когда дата задана в виде строки)	тип: загранпаспорт РФ номер: 719999999 дата выдачи: 21.08.2011	тип: загранпаспорт РФ серия: 71 номер: 9999999 дата выдачи: 21.08.2011	Указан недопустимый формат даты. Список форматов даты можно расширить необходимыми форматами
23	EXPIRY _DATE_ DOCUME NT_EXP IRED	Документ просрочен	На момент стандартизации дата окончания действия документа больше или равна текущей дате	тип: загранпаспорт РФ номер: 719999999 дата выдачи: 01.01.2012	Тип: загранпаспорт РФ серия: 71 номер: 9999999 дата выдачи: 01.01.2012	Дата окончания документа должна быть больше или равна текущей дате (дата окончания срока действия загранпаспорта 01.01.2022, текущая дата 01.02.2022)
<b>24 Код подразделения (если предусмотрено для типа ДУЛ) (departmentCode)</b>						
25	EMPTY_ CODE	Код подразделения пустой	Пустое значение в поле «код подразделения»	тип: паспорт РФ номер: 4014785436 дата выдачи: 01.01.2014	тип: паспорт РФ серия: 4014 номер: 785436 дата выдачи: 01.01.2014	Пустое значение в поле «код подразделения»
26	BAD_CO DE_FOR MAT	Код не соответствует формату	В коде подразделения отсутствует ровно шесть арабских цифр	тип: паспорт РФ номер: 4014785436 дата выдачи: 01.01.2014	тип: паспорт РФ серия: 4014 номер: 785436 дата выдачи: 01.01.2014	Код подразделения состоит из 5 арабских цифр, ожидается 6

				<u>код</u> <u>подразде</u> <u>ления:</u> <u>99-999</u> кем выдан: УФМС по Санкт- Петербур гу	кем выдан: УФМС по Санкт-Петербургу	
27	NOT_VA LID_CO DE	Код подразделения несуществующего региона	Первые две цифры кода подразделения, указывающие код региона или субъекта РФ, отсутствуют в справочнике кодов регионов ФНС	тип: паспорт РФ номер: 4 014785436 дата выдачи: 01.01.2014 <u>код</u> <u>подразде</u> <u>ления:</u> <u>009-000</u> кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 01.01.2014 <u>код подразделения: 0</u> <u>09-000</u> кем выдан: УФМС по Санкт-Петербургу	Первые две цифры кода подразделения «00», указывающие код региона (закрепленный за субъектом РФ), отсутствуют в справочнике кодов регионов ФНС
28	BAD_CO DE_LEV EL	Невалидный код уровня подразделения	Третья цифра кода подразделения не соответствует допустимому уровню паспортного отдела. Существует только четыре уровня подразделений: <ul style="list-style-type: none"> <li>• 0 – УФМС;</li> <li>• 1 – ГУВД или МВД региона;</li> <li>• 2 – УВД или ОВД района или города;</li> <li>• 3 – отделение полиции (обычно в селе или деревне).</li> </ul>	тип: паспорт РФ номер: 4 014785436 дата выдачи: 01.01.2014 <u>код</u> <u>подразде</u> <u>ления:</u> <u>609-871</u> кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 01.01.2014 <u>код подразделения: 6</u> <u>09-871</u> кем выдан: УФМС по Санкт-Петербургу	Третья цифра кода подразделения, указывающая на уровень паспортного отдела, равна «9». Существует только четыре уровня подразделений
29	CODE_C HANGE	Код подразделения был изменен относительно исходных данных	Код подразделения очищен от мусора (знаки препинания, лишние пробелы), добавлен дефис в случае его отсутствия (например, 620033 ? 620-033)	тип: паспорт РФ номер: 4 014785436 дата выдачи: 01.01.2014 <u>код</u> <u>подразде</u> <u>ления:</u> <u>620,033</u> кем выдан: УФМС по Санкт- Петербур гу	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 01.01.2014 <u>код подразделения:</u> 620-033 кем выдан: УФМС по Санкт-Петербургу	Код подразделения был очищен от мусорного знака, между тремя цифрами кода добавлен дефис

кем  
выдан:  
УФМС  
по  
Санкт-  
Петербур  
гу

### 30 Кем выдан (если предусмотрено для типа ДУЛ) (*issueAuthority*)

31	EMPTY_ AUTHOR ITY	Место выдачи пустое	Пустое значение в поле «Кем выдан»	тип: пас порт РФ номер: 4 014785436 дата выдачи: 01.01.2014 код подразде ления: 620-033 <u>кем выдан: ( пустое поле)</u>	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 01.01.20 14 код подразделения: 620-033 <u>кем выдан: (пустое поле)</u>	Пустое значение в поле «место выдачи»
----	-------------------------	---------------------------	---------------------------------------	---	---	--

### 32 Тип документа (*qualifierType*)

33	BAD_TY PE	Неизвестный тип документа	В поле «Тип документа» передан тип документа, отсутствующий в справочнике типов ДУЛ	тип: паспорт Азербай джана номер: 40147854 36	тип: паспорт Азербайджана номер: 4014785436	Тип документа в примере « паспорт гражданина Азербайджана» отсутствует в справочнике типов ДУЛ. Для иностранных паспортов необходимо указывать тип «Паспорт иностранного гражданина»
----	--------------	---------------------------------	---	---	---	---

### 34 Кросс проверки

35	EXPIRY_ _DATE_ LIMIT_ CLASH	Дата окончания больше срока действия	Дата окончания больше максимального срока действия документа, рассчитанного на основе даты выдачи	тип: заг ранпасп орт РФ номер : 719999999 дата выдачи: 21.08.2010 <u>дата окончан ия срока действия : 21.08.20 50</u>	тип: загранпаспорт РФ серия: 71 номер: 9999999 дата выдачи: 21.08.20 10 <u>дата окончания срока действия: 21.08.2050</u>	Срок действия загранпаспорта РФ нового образца — 10 лет (для серий 71, 72, 73, 75, 76, 80, 82). Сро к действия документов, удостоверяющих личность: <ul style="list-style-type: none"><li>• Дипломатический паспорт: 5 лет;</li><li>• Служебный паспорт: 5 лет;</li><li>• Паспорт моряка: 5 лет;</li><li>• Разрешение на временное проживание: 3 года;</li><li>• Удостоверение беженца: 3 года;</li><li>• Водительское удостоверение: 10 лет;</li><li>• Вид на жительство: 5 лет;</li><li>• Свидетельство о предоставлении</li></ul>
----	--------------------------------------	---	--	---	--	--

						временного убежища: 1 год;
						<ul style="list-style-type: none"> <li>• СССР заграничный паспорт: 5 лет;</li> <li>• СССР паспорт МинМорФлота: 5 лет.</li> </ul>
36	BAD_AGE	Документ невозможно сверить с датой рождения	Поле «дата рождения» пустое либо неверный формат даты	тип: заграничный паспорт РФ номер: 7 19999999 дата выдачи: 21.08.2010	тип: Заграничный паспорт РФ серия: 71 номер: 9999999 дата выдачи: 21.08.2010	Поле «дата рождения» пустое
				<u>дата рождения лица: (пустое значение)</u>		
37	DOCUMENT_AGE_EXPIRED	Документ просрочен по возрасту	Дата выдачи документа больше минимально допустимой даты выдачи документа такого типа по возрасту клиента, рассчитанному на основе даты рождения (например, паспорт 20, 45 лет + 90 дней)	тип: паспорт РФ номер: 4 000785436 дата выдачи: 03.11.2000 код подразделения: 7 80-121 кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: 40 00 номер: 785436 дата выдачи: 03.11.2000 код подразделения: 7 80-121 кем выдан: УФМС по Санкт-Петербургу	На момент 2022 года лицу исполнилось 47 лет (дата рождения 02.11.1975), дата выдачи паспорта 03.11.2000 (т.е. это дата выдачи паспорта в 25 лет) => паспорт просрочен по возрасту, т.к. в 45 лет получают новый паспорт РФ
				<u>дата рождения лица: 02.11.1975</u>		
38	AGE_IS_SUE_DATED_CLASS	Документ не соответствует возрасту	Дата выдачи документа не соответствует возрасту лица, т.к. документ выдается с определенного возраста	тип: паспорт РФ номер: 4 014785436 дата выдачи: 27.08.2014	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 7 80-121 кем выдан: УФМС по Санкт-Петербургу	Документ не соответствует возрасту, т.к. первый паспорт гражданина РФ может быть получен в случае достижения 14-летнего возраста, а не в год его рождения. Документ выдается с определенного возраста:
				<u>дата рождения лица: 10.06.2014</u>		<ul style="list-style-type: none"> <li>• Паспорт РФ: с 14 лет;</li> <li>• Водительское удостоверение: с 16 лет;</li> </ul>



- Военный билет: с 18 лет;
- Удостоверение офицера: с 18 лет;
- Удостоверение беженца: с 18 лет;
- СНИЛС: с 14 лет человек может получить сам, а до 14 за него могут получить его родители (проверка на возраст не осуществляется);
- СССР паспорт: с 16 лет.

39	SERIES _CODE_ CLASH	Код подразделения конфликтует с ОКАТО серии	Регион кода подразделения не соответствует региону, указанному в серии паспортов РФ). Первые 2 цифры серии паспорта — ОКАТО региона, который должен соответствовать региону из кода подразделения	тип: паспорт РФ номер: <u>9099785436</u> дата выдачи: <u>27.08.1999</u> код подразделения: <u>0-121</u> кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: <u>9099</u> номер: 785436 дата выдачи: 27.08.1999 код подразделения: <u>780-121</u> кем выдан: УФМС по Санкт-Петербургу	ОКАТО региона (90 — Республика Северная Осетия-Алания) не соответствует региону из кода подразделения (78 — Санкт-Петербург)
40	SERIES _ISSUE_ _DATE_ CLASH	Серия не соответствует году выдачи	Год выдачи паспорта РФ не соответствует году, определенному по серии (допустимое отклонение серии -3 года и +5 лет)	тип: паспорт РФ номер: <u>4014785436</u> дата выдачи: <u>27.08.2005</u> код подразделения: <u>780-121</u> кем выдан: УФМС по Санкт-Петербургу	тип: паспорт РФ серия: <u>4014</u> номер: 785436 дата выдачи: <u>27.08.2005</u> код подразделения: <u>780-121</u> кем выдан: УФМС по Санкт-Петербургу	Год выпуска бланка «14» (вторая пара цифр серии) на 9 лет больше даты выдачи документа
41	<b>Возраст</b>					
42	BAD_AGE	Документ невозможно сверить с датой рождения	Поле «дата рождения» пустое либо неверный формат даты	тип: загранпаспорт РФ номер: 719999999	тип: загранпаспорт РФ серия: 71 номер: 9999999	Поле «дата рождения» пустое

дата выдачи: 21.08.2010  
21.08.2010

дата рождения лица:  
(пустое значение)  
рождения лица:  
(пустое значение)

## Пример отображения маркеров проверок документа

Рассмотрим ФЛ с датой рождения 01.01.1960 г. В приведенном примере Паспорт РФ имеет набор маркеров практически для каждого компонента:

1. Серия не соответствует году выдачи — год выдачи по серии отстоит от года выдачи более, чем на 3 года (допустимое отклонение -3 года и +5 лет )
2. Документ не соответствует возрасту — паспорт выдан раньше даты рождения.
3. Регион отсутствует в ОКАТО — среди субъектов РФ не существует региона с кодом ОКАТО «00».
4. Серия из будущего — две последние цифры серии 27, год выдачи считается равным 2027 (на момент создания примера текущий год — 2022).
5. Несуществующий номер — номер паспорта РФ должен начинаться с 000100.
6. Код подразделения несуществующего региона — первые две цифры кода подразделения отсутствуют в справочнике кодов регионов.
7. Недопустимый код уровня паспортного отдела — третья цифра кода подразделения отвечает за уровень паспортного отдела, который ограничен набором цифр от 0 до 3.
8. Недопустимая дата выдачи — паспорт РФ выдается с 1997 года.

Тип: Паспорт гражданина РФ ▾

Основной

Исходный: 0027•000007•01.01.1900•отдел УФМС России по гр. Москве•934-491• 📄  
❗ Плохая серия или номер (серия не соответствует году выдачи, документ не соответствует возрасту)

Серия: 00 27 Номер: 000007  
Регион отсутствует в ОКАТО, серия из будущего, несуществующий номер

Кем выдан: отдел УФМС России по гр. Москве

Код подр.: 934-491  
Код подразделения несуществующего региона, недопустимый код уровня паспортного отдела

Когда выдан: 01.01.1900 📅 Истекает: 📅  
Недопустимая дата выдачи

Доп.:

Стандартизовать Подтвердить Отложить ▾

# СТАТУС АКТУАЛЬНОСТИ ДОКУМЕНТА

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	ACTUAL	Актуальный	Паспорт актуальный, т.к. не является недействительным документом по данным ФМС, просроченным или продленным на замену	тип: паспорт РФ оригинал номер: 4014785436, дата выдачи: 27.08.2014 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу  дата рождения лица: 27.08.2000	тип: паспорт РФ серия: 40 14 номер: 785436 дата выдачи: 27.08.2014 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу	2021 год является текущим годом для указанного примера
2	INVALID	Недействительный	Документ признан недействительным в соответствии с перечнем недействительных документов, предоставляемым ФМС. Код качества используется только в случае, если настроена проверка по перечню недействительных документов	тип: паспорт РФ оригинал номер: 4507654343	тип: паспорт РФ серия: 45 07 номер: 654343	Паспорт недействителен в соответствии с перечнем недействительных документов
3	OUTDATED	Просроченный	Срок на замену документа уже прошёл. Документ просрочен, если: 1) Указана дата истечения, и она меньше текущей даты на момент стандартизации — для всех типов документов, кроме паспорта РФ. Для паспорта РФ статус «просроченный» будет установлен в случае, если с момента изменения возраста владельца в 20 или 45 лет прошло более 90 дней (проверка выполняется, если указаны дата выдачи и дата рождения) или указана дата истечения срока документа, которая меньше текущей даты (на момент стандартизации данных) более чем на 90 дней. Если с момента изменения возраста владельца в 20 или 45 лет прошло менее 90 дней или указана дата истечения срока документа, которая меньше текущей даты менее чем на 90 дней, то документу будет	пример 1: тип: паспорт РФ номер: 4021785436, дата выдачи: 03.09.2015 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу	пример 1: тип: паспорт РФ серия: 40 21 номер: 785436 дата выдачи: 03.09.2015 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу дата окончания срока действия: (пустое значение) дата рождения лица: 03.09.2001 пример 2: тип: паспорт РФ серия: 40 21 номер: 785436 дата выдачи: 03.09.2015	для примера 1: с момента изменения возраста владельца в 20 лет прошло более 90 дней (а именно, 92 дня), 03.12.2021 является текущей датой для указанного примера для примера 2: указана

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
			установлен статус «Продлённый на замену».	Санкт-Петербург	код подразделения: 780-121	дата истечения срока документа, которая меньше текущей даты (03.12.2021 является текущей датой для указанного примера) более чем на 90 дней (а именно, на 91 день)
			2) Указана дата выдачи, и возраст документа больше максимального срока действия.	гу <u>дата</u> <u>окончания срока действия</u> <u>: (пустое значение)</u>  <u>дата рождения лица</u> <u>: 03.09.2001</u> пример 2: тип: паспорт РФ номер: 4021785436,, дата выдачи: 03.09.2015 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу <u>дата</u> <u>окончания срока действия</u> <u>: 04.09.2021</u> дата рождения лица: 03.09.2001	место выдачи: УФМС по Санкт-Петербургу <u>дата окончания срока действия</u> : 04.09.2021 дата рождения лица: 03.09.2001	
4	PROLO NGED	Продлённый на замену	Паспорт пора заменить, но он всё ещё действителен, т.к.: <ul style="list-style-type: none"><li>текущая дата (на момент стандартизации данных) укладывается в срок 90 дней от изменения возраста владельца в 20 или 45 лет;</li><li>указана дата истечения срока документа, которая меньше текущей даты (на момент стандартизации данных) менее чем на 90 дней.</li></ul>	тип: паспорт РФ номер: 4021785436,, дата выдачи: 03.11.2015	тип: паспорт РФ серия: 40 21 номер: 785436 дата выдачи: 03.11.2015 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу дата окончания срока действия: (пустое значение)	Текущая дата (03.12.2021 является текущей датой для указанного примера) укладывается в срок 90 дней от изменения возраста

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
				код подразделения: 780-121	дата рождения лица: 03.11.2001	владельца в 20 лет
				место выдачи: УФМС по Санкт-Петербургу	дата окончания срока действия: (пустое значение)	
				дата рождения лица: 03.11.2001		
				тип: паспорт РФ номер: 4021785436,, дата выдачи: 03.09.2015	тип: паспорт РФ серия: 40 21 номер: 785436 дата выдачи: 03.09.2015 код подразделения: 780-121 место выдачи: УФМС по Санкт-Петербургу	Указана дата окончания срока документа, которая меньше текущей даты (07.12.2021 является текущей датой для указанного примера) менее чем на 90 дней. Если дата окончания срока действия документа не была бы установлена, то документ имел статус «OUTDATE D /Просроченный» (т.к. паспорт, подлежащий замене в связи с достижениями
				код подразделения: 780-121	дата окончания срока действия: 03.12.2021	
				место выдачи: УФМС по Санкт-Петербургу	дата окончания срока действия: 03.12.2021	
				дата рождения лица: 03.09.2001		

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
						ем 20 или 45 лет, действует до получения нового, но не более чем в течение 90 дней. В примере на момент 07.12.2021 прошло более 90 дней с момента наступления 20-летнего возраста)

# ОТОБРАЖЕНИЕ ГРАНУЛЯРНЫХ КОДОВ КАЧЕСТВА И МАРКЕРОВ ДУЛ

Каждому полю документа соответствует определенный набор маркеров, и каждый маркер отображается (только если он проставлен) строго под формой, к которой он прикреплен.

№	Поле (ParentField)	Литера	Локализация
1	Тип документа ( <b>qualifierType</b> )	BAD_TYPE	Сомнительный тип документа
2	Код качества ( <b>qualityCode</b> )	SERIES_ISSUE_DATE_CLASH	Серия не соответствует году выдачи
3		SERIES_CODE_CLASH	Код подразделения конфликтует с ОКАТО серии
4		EXPIRY_DATE_LIMIT_CLASH	Дата окончания больше срока действия
5		DOCUMENT_AGE_EXPIRED	Документ просрочен по возрасту
6	Серия ( <b>documentSeries</b> )	BAD_AGE	Документ невозможно сверить с датой рождения
7		AGE_ISSUE_DATE_CLASH	Документ не соответствует возрасту
8		EMPTY_SERIES	Серия пустая
9		BAD_SERIES_FORMAT	Серия не соответствует формату
10		BAD_SERIES_REGION	Регион отсутствует в ОКАТО
11		NOT_VALID_SERIES	Не существующая серия
12	Номер ( <b>documentNumber</b> )	FUTURE_SERIES	Серия из будущего
13		SERIES_CHANGE	Серия была изменена относительно исходных данных
14		EMPTY_NUMBER	Номер пустой
15		BAD_NUMBER_FORMAT	Номер не соответствует формату
16	Когда выдан ( <b>issueDate</b> )	NOT_VALID_NUMBER	Несуществующий номер
17		NUMBER_CHANGE	Номер был изменен относительно исходных данных
18		EMPTY_ISSUE_DATE	Пустая дата выдачи
19	Кем выдан ( <b>issueAuthority</b> )	BAD_ISSUE_DATE_FORMAT	Неверный формат даты
20		BAD_ISSUE_DATE	Недопустимая дата выдачи
21		FUTURE_ISSUE_DATE	Дата выдачи из будущего
22		ISSUE_DATE_DOCUMENT_EXPIRED	Документ просрочен
23	Код подразделения ( <b>departmentCode</b> )	EMPTY_AUTHORITY	Место выдачи пустое
24		EMPTY_CODE	Код подразделения пустой
25		BAD_CODE_FORMAT	Код не соответствует формату
26		NOT_VALID_CODE	Код подразделения несуществующего региона

27	BAD_CODE_LEVEL	Недопустимый код уровня паспортного отдела	
28	CODE_CHANGE	Код подразделения был изменен относительно исходных данных	
29	Дата окончания ( <b>expiryDate</b> )	EMPTY_EXPIRY_DATE	Пустая дата окончания документа
30	BAD_EXPIRY_DATE_FORMAT	Неверный формат даты	
31	BAD_EXPIRY_DATE	Дата окончания меньше даты выдачи	
32	EXPIRY_DATE_DOCUMENT_EXPIRED	Документ просрочен	

⚠ Маркер SERIES\_CHANGE проставляется в случаях: серия очищена от мусора, часть переложена из номера, либо наоборот, переложена в номер.

## Пример

Рассмотрим ФЛ с датой рождения 01.01.1960 г. В приведенном примере Паспорт РФ имеет набор маркеров практически для каждого компонента:

1. Серия не соответствует году выдачи — год выдачи по серии отстоит от года выдачи более, чем на 3 года (допустимое отклонение -3 года и +5 лет)
2. Документ не соответствует возрасту — паспорт выдан раньше даты рождения.
3. Регион отсутствует в ОКАТО — среди субъектов РФ не существует региона с кодом ОКАТО «00».
4. Серия из будущего — две последние цифры серии 27, год выдачи считается равным 2027 (на момент создания примера текущий год — 2022).
5. Несуществующий номер — номер паспорта РФ должен начинаться с 000100.
6. Код подразделения несуществующего региона — первые две цифры кода подразделения отсутствуют в справочнике кодов регионов.
7. Недопустимый код уровня паспортного отдела — третья цифра кода подразделения отвечает за уровень паспортного отдела, который ограничен набором цифр от 0 до 3.
8. Недопустимая дата выдачи — паспорт РФ выдается с 1997 года.



Тип: **Паспорт гражданина РФ** ▾

Основной

Исходный: **0027•000007•01.01.1900•отдел УФМС России по гр. Москве•934-491** 📄

**❗ Плохая серия или номер (серия не соответствует году выдачи, документ не соответствует возрасту)**

Серия: **00 27** Номер: **000007**

Регион отсутствует в ОКАТО, серия из будущего, несуществующий номер

Кем выдан: **отдел УФМС России по гр. Москве**

Код подр.: **934-491**

Код подразделения несуществующего региона, недопустимый код уровня паспортного отдела

Когда выдан: **01.01.1900** 📅 Истекает:  📅

Недопустимая дата выдачи

Доп.:

[Стандартизовать](#) [Подтвердить](#) [Отложить](#) ▾

# СТАТУС НЕДЕЙСТВИТЕЛЬНОСТИ ДЛЯ ДОКУМЕНТА

## *Статус недействительности для документа*

Код	Литера	Описание
0	<b>ACTUAL</b>	Актуальный
1	<b>INVALID</b>	Найден в справочнике недействительных документов
2	<b>OUTDATED</b>	Просроченный, определенный при сравнении даты выдачи, даты истечения и даты рождения
3	<b>PROLONGED</b>	Дата истечения меньше текущей, но укладывается в срок 90 дней

# СКОРИНГ ПОЛНОТЫ ДУЛ

- Логика расчета скоринга
- Весовые коэффициенты компонентов ДУЛ

Знание числового показателя полноты документа упрощает задачу слияния дубликатов и выбора наиболее ценного победителя из двух одинаковых документов.

## Логика расчета скоринга

Каждый тип ДУЛ имеет набор обязательных компонентов, не все из которых имеют одинаковый вес с точки зрения бизнес-логики. Поэтому:

- Каждый обязательный компонент имеет весовой коэффициент (в процентах).
- В сумме все коэффициенты в разрезе одного типа ДУЛ дают 100%.
- Необязательные компоненты никак не влияют на скоринг.

Пример (для Паспорта РФ):

Серия	Номер	Дата выдачи	Кем выдан	Код подразделения	Дата окончания	fullnessRatio
k=35	k=35	k=20	k=6	k=4	k=0	
28 05	409087	20.10.2011	УФМС России	620-033		100
28 05		20.10.2011	УФМС России	620-033		65
28 05			УФМС России	620-033		45

## Весовые коэффициенты компонентов ДУЛ



- — необязательный компонент
- Для обязательных компонентов выставлен весовой коэффициент (в процентах)

Тип документа	Серия	Номер	Дата выдачи	Кем выдан	Код подразделения	Дата окончания
Паспорт РФ	35	35	20	6	4	
Свидетельство о рождении	35	35	25	5		
Свидетельство о смерти	35	35	25	5		

Загранпаспорт гражданина РФ	35	35	17	3		10
Дипломатический паспорт гражданина РФ	35	35	17	3	—	10
Служебный паспорт гражданина РФ	35	35	17	3	—	10
Водительское удостоверение	35	35	17	3	—	10
Военный билет	35	35	25	5	—	—
Удостоверение личности офицера	35	35	25	5	—	—
Паспорт моряка	35	35	17	3	—	10
Удостоверение вынужденного переселенца	35	35	25	5	—	—
Вид на жительство	35	35	17	3	—	10
Разрешение на временное проживание (для лиц без гражданства)	35	35	17	3	—	10
Удостоверение беженца в РФ	35	35	17	3	—	10
Свидетельство о предоставлении временного убежища	35	35	17	3	—	10
Миграционная карта	35	35	25	5	—	—
Страховое свидетельство обязательного пенсионного страхования, СНИЛС	—	70	30	—	—	—
Паспорт гражданина СССР	35	35	25	5	—	—
Загранпаспорт гражданина СССР	35	35	17	3	—	10
Паспорт Минморфлота	35	35	17	3	—	10

**Без проверок форматов: только минимальные проверки заполненности полей и непротиворечивости дат:**

Паспорт иностранного гражданина	40	40	20	—	—	—
Справка об освобождении из мест лишения свободы	40	40	20	—	—	—
Свидетельство о регистрации ходатайства о признании иммигранта беженцем	40	40	20	—	—	—
Свидетельство о государственной регистрации юридического лица	40	40	20	—	—	—
Свидетельство о внесении записи в Единый государственный реестр юридических лиц	40	40	20	—	—	—
Аккредитация	40	40	20	—	—	—
Лицензия	40	40	20	—	—	—
Свидетельство о постановке на учет в налоговом органе	40	40	20	—	—	—
Документ о присвоении ИНН	40	40	20	—	—	—
Выписка из ЕГРИП	40	40	20	—	—	—
ДУЛ на период принятия гражданства РФ	40	40	20	—	—	—
	35	35	30	—	—	—

Временное удостоверение личности гражданина РФ						
Виза	⊖	80	20	⊖	⊖	⊖
Сертификат адвоката	⊖	80	20	⊖	⊖	⊖
Иностранное свидетельство о рождении	⊖	80	20	⊖	⊖	⊖
НДФЛ	⊖	80	20	⊖	⊖	⊖
Удостоверение ветерана	⊖	80	20	⊖	⊖	⊖
Патент	⊖	80	20	⊖	⊖	⊖
Разрешение на работу	⊖	80	20	⊖	⊖	⊖
Удостоверение личности частного охранника	⊖	80	20	⊖	⊖	⊖
Учетный код ФМС	⊖	80	20	⊖	⊖	⊖

# КОД КАЧЕСТВА ИНН ФЛ

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил ИНН и/или подтвердил его	526317984689	526317984689
3	GOOD	Корректное значение	Все проверки прошли, ИНН корректный	526317984689	526317984689
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)
5	GOOD_CHANGED	Убран мусор	Кроме корректного ИНН исходное значение содержит другие «мусорные» символы, которые были убраны	INN 526317984689	526317984689
<b>6 На ручную обработку</b>					
7	NOT_VALID_CHECKSUM	Не совпадает контрольная сумма	Рассчитанное контрольное число отличается от соответствующего знака в ИНН	111111111111	(пустое значение)
8	NOT_VALID_FORMAT	Некорректное значение	В исходных данных содержится только мусорное текстовое значение и/или ИНН состоит из 12 нулей	текст00000	(пустое значение)
9	NOT_VALID_LENGTH	Некорректная длина	Длина ИНН не равна 12	516281903	(пустое значение)
10	NOT_VALID_TAX_CODE	Некорректный код налоговой	Код налогового органа не найден в справочнике (первые четыре цифры)	000000000040	(пустое значение)
11	MISSED	Отсутствует код качества	Атрибут не проходил стандартизацию		
<b>12 Устаревшие значения</b>					
13	CONFIRMED	Значение подтверждено (ЕГРЮЛ) (устар.)			
14	RESTORED	Значение восстановлено (ЕГРЮЛ) (устар.)			
15	CHANGED	Значение изменено (ЕГРЮЛ) (устар.)			
16	NOT_SURE	Сомнительное значение (устар.)			
17	EDITED	Правильное значение (устар.)			



# КОДЫ КАЧЕСТВА EMAIL

- Код качества адреса электронной почты
- Маркеры проверок и изменений email
- Причины признания email плохим

Связанные страницы:

- EMAIL
- Стандартизация email
- Поиск по email
- Просмотр карточки. Email
- Справочник одноразовых e-mail
- Адрес электронной почты
- Тип email
- Тип домена email



# КОД КАЧЕСТВА EMAIL

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
<b>1 Не требуют проверки человеком</b>						
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил email и /или подтвердил его	adel2yandex.ru	adel@yandex.ru	Любые данные, подтвержденные контролером, имеют статус «CONFIRMED_MANUALLY»
3	GOOD	Правильное значение	Все проверки прошли, адрес корректный	пример 1: syberia@gmail.com пример 2: \sofa2204@mail.ru пример 3: <sofa2204@mail.ru>	пример 1: syberia@mail.com пример 2: sofa2204@mail.ru пример 3: sofa2204@mail.ru	Все проверки прошли, адрес корректный
4	GOOD_CHANGED	Хорошее исправленное значение	В адресе были выполнены изменения (опечатки)	mc@rambler.ry	mc@rambler.ru	В адресе была исправлена опечатка (символ «у» заменен на «и»)
5	EMPTY	Пустое значение	В исходном поле нет букв или цифр )		(пустое значение)	В исходном поле нет букв или цифр
<b>6 На ручную обработку</b>						
7	GOOD_DISPOSABLE	Временный e-mail	Домен принадлежит к «временным»	xjf91838@0-mail.com	xjf91838@0-mail.com	Домен принадлежит к «временным»
8	NOT_VALID	Некорректное значение	примеры представлены на странице Причины признания email плохим			
9	TYPE_MISMATCH	Неправильный тип	Исходный тип некорректный	email: ubivash@gmail.com тип: FAX	email: UBIVASH@GMAIL.COM тип: OTHER	Электронная почта пришла как телефон с типом, который для нее не предусмотрен

# МАРКЕРЫ ПРОВЕРОК И ИЗМЕНЕНИЙ EMAIL

## Маркеры проверок и изменений email

Может быть несколько значений в строке (разделитель « » — пробел).

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	LOCAL_PUNY	Найдена punycode- последовательность	В локальной части (до символа «@») есть punycode-последовательность ( <b>xn--</b> в начале строки или после точки)	xn--p1ai@mydomain.ain.ru	xn--p1ai@mydomain.ru	
2	LOCAL_QUOTE_START	Удалены кавычки из локальной части	Локальная часть (до символа «@») начинается с одинарной кавычки. Одинарная кавычка допустима согласно RFC, но чаще всего это мусор	'vasiliy@mydomain.ain.com	vasiliy@mydomain.com	
3	DOMAIN_INTRANET	Внутренний домен предприятия	Домен из списка внутренних доменов предприятия	petrovna@intranet.customer.ru	petrovna@intranet.customer.ru	Должен быть подключен справочник заказчика
4	DOMAIN_TEMP	Одноразовый домен	Домен из списка одноразовых	badr3adf@temp-mail.com	badr3adf@temp-mail.com	Должен быть подключен справочник одноразовых email
5	DOMAIN_MISPRINT	Исправлена опечатка в доменной части	Исправили опечатку в домене	vasya@rsmbler.ru	vasya@rambler.ru	
6	HAS_UNPARSED_PARTS	Удален лишний текст	Текст в исходной строке не является частью email	пример 1: [mailto:sofa2204@mail.ru] пример 2: e mail sofa2204@mail.ru пример 3: email: sofa2204@mail.ru пример 4: Алексе й Викторо вич sofa2204@mail.ru	пример 1: sofa2204@mail.ru пример 2: sofa2204@mail.ru пример 3: sofa2204@mail.ru пример 4 sofa2204@mail.ru	Есть текст из исходной строки, который не является частью email
7	SEVERAL_EMAILS	Более одного email в исходной строке	В исходной строке присутствовало несколько email через символ-разделитель (стандартный набор	vasya@mail.ru vasya@g	первый номер: vasya@mail.ru второй номер: vasya@gmail.com	В зависимости от настроек ЕК, результатом стандартизации является либо все найденные email в строке (для

			символов: пробел, «. », «,», «;», «\», «/», «^»)	mail.com vasya@a.org	третий номер: vasya@a.org	каждого email будет проставлен маркер SEVERAL_EMAILS), либо первый найденный email в исходной строке
8	GARBAGE	Удален мусор	Убрали «мусорные» символы	пример 1: ivan@bk.ru пример 2: sofa 2204@mail.ru	пример 1: ivan@bk.ru пример 2: sofa 2204@mail.ru	В примере 2 пробел в доменной части считается «мусорным» и удаляется
9	TRANSPARENT	Проведена трансграфика	Заменяли русскую букву «с» на английскую «C» в случае опечатки. Если в email присутствует русская буква «C» при остальных латинских, а также если в домене она записана через ACE (xn--om-nmc)	kutarova@gmail.xn--om-nmc	kutarova@gmail.com	
10	MISPRI NT	Исправлена опечатка в доменной части	Исправили опечатку в домене	grisha@mail.ru	grisha@mail.ru	
11	LOCAL_ SUB_ ADDRESSING	Локальная часть содержит тег, который будет проигнорирован почтовым сервером	Локальная часть (до символа «@») содержит тег, который будет проигнорирован почтовым сервером	ivanivanov+personal@gmail.com	ivanivanov+personal@gmail.com	Для адреса вида ivanivanov@gmail.com синонимами являются ivanivanov+web@gmail.com, ivanivanov+market@gmail.com, ivanivanov+personal@gmail.com?и пр. Письма по всем этим адресам будут доставлены на адрес ivanivanov@gmail.com. Плюс-адресация позволяет пользователям регистрироваться в различных сервисах, используя разные псевдонимы электронной почты, а затем удобно фильтровать все электронные письма от этих сервисов. Знак плюс – тег подадресации для всех доменов, кроме yahoo
12	LOCAL_ ALIAS	Локальная часть содержит символы, которые будут проигнорированы или изменены	Локальная часть (до символа «@») содержит символы, которые будут проигнорированы или изменены почтовым сервером	i.v.a.n.ivanov@gmail.com	i.v.a.n.ivanov@gmail.com	Gmail игнорирует точки в адресах, то есть почтовый сервис считает, что ivanivanov@gmail.com, ivan.ivanov@gmail.com или i.v.a.n.i.v.a.n.o.v@gmail.com – это один и тот же ящик

почтовым сервером						
13	DOMAIN_ALIAS	Есть синоним в доменной части из списка	В доменной части содержится алиас. Почтовые алиасы (псевдонимы, синонимы) — это разные по написанию адреса, относящиеся к одному и тому же почтовому ящику. Например, синонимы для @yandex.ru — @ya.ru, @yandex.kz, @yandex.ua, @yandex.com, @yandex.by	ivanivanov@ <b>googlemail.com</b>	ivanivanov@ <b>googlemail.com</b>	Синоним для @gmail.com — @googlemail.com. Почта, отправленная на адрес ivanivanov@googlemail.com или ivanivanov@googl.com, придет в один и тот же ящик
14	DOMAIN_EMPTY	Доменная часть пуста	Доменная часть пуста после удаления мусора и /или исправления опечаток в домене	alina_a@ <b>123</b>	alina_a	При стандартизации email в доменной части произошла замена недопустимых значений (цифр «123») на пустую строку, в результате получили пустой домен
<b>15 Устаревшие</b>						
16	LOCAL_EMPTY	Локальная часть пуста (не используется)	Если локальная часть пуста, то будет указан маркер качества HAS_UNPARSED_EMAIL, либо строка уйдет в неразобранную часть другого email	@gmail.com	@gmail.com	

# ПРИЧИНЫ ПРИЗНАНИЯ EMAIL ПЛОХИМ

## Причины признания email плохим

Может быть несколько значений в строке (разделитель « » — пробел).

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	LOCAL_ BAD_SY MBOL	Некорректны й символ в локальной части	Некорректный символ в локальной части email (до знака «@»)	пример 1: sofa 2204 @mail.ru пример 2: ДАНА— 21@RIO. EDU	пример 1: 2204@mail.ru пример 2: 21@RIO.EDU	Ставится всегда, если есть слева неразобранная часть. Кроме случаев, когда: <ul style="list-style-type: none"><li>• есть одиночные спецсимволы / \ в начале;</li><li>• email отформатирован почтовым клиентом: <pre>&lt;example@domain.local&gt; [mailto:example@domain.local]</pre></li><li>• неразобранная часть: содержит пробел и слов</li></ul>

---

о e-mail;  
заказ  
нчи  
вает  
ся  
пунктуа  
ции  
й с  
про  
белом  
м  
(напри  
мер, «  
: »);  
содержи  
т  
текст  
на  
кирилл  
ице,  
заказ  
нчи  
ваю  
щий  
ся  
про  
белом.  
м.

---

2	LOCAL_BAD_LENGTH	<b>Некорректная длина локальной части</b>	Некорректная длина в локальной части email (до знака «@»). Максимальная длина — 63 символа	"very long long bla lba lbal very long long long bla lba lbal very long long long bla lba lbal very long long long bla lba lbalv ery long long long bla lba lbal very long long long bla
---	------------------	---	--	--

---

				<b>lba lba!"</b> @email. com	
3	LOCAL_ FREAKY	<b>Запрещенные символы в локальной части</b>	Корректная локальная часть email (до знака «@») содержит символы <b>!#\$%&amp;''/= ?^` {}~</b> . В основном эти символы запрещают крупные сервисы, в явном виде их обычно нельзя ввести в поле логина	<b>ivanov?</b> @gmail. com	
4	LOCAL_ BAD_SP ECIFIC	<b>Локальная часть адреса не соответствует правилам доменов</b>	Локальная часть адреса не соответствует правилам почтового сервиса	пример 1: <b>abb@</b> mail.ru пример 2: <b>ivanov</b> --ivan@y andex.ru	пример 1: для доменов mail.ru, list. ru, bk.ru, inbox.ru допустимое количество символов в логине — от 4 до 31 включительн о.  пример 2: недопустимо 2 и более знака «дефис» подряд Правила доменов описаны зде сь
5	LOCAL_ BLOCK_ PREFIX	<b>Локальная часть в списке заблокирован ных префиксов</b>	Локальная часть адреса запрещена, т. к. находится в справочнике заблокированных префиксов	<b>jobs@a.</b> org	Если подключен с правочник з аблокирован ных префиксов, то для email, имеющих запрещенну ю локальную часть, будет указан маркер LOCAL_BLOC K_PREFIX
6	LOCAL_ NON_AS CII	<b>Локальная часть содержит отличные от базового ASCII символы</b>	Локальная часть содержит отличные от базового ASCII символы (в базовые символы кириллица не входит)	пример 1: <b>вася@</b> почта. рус пример 2: <b>вася@</b> mail.ru	пример 1: <b>вася@почт</b> а.рус пример 2: <b>вася@mail.</b> ru
7	DOMAIN _BAD_S YMBOL	<b>Не распознан домен</b>	Некорректный символ в домене. Выставляется в одном из случаев:	<b>vasyan@</b> <b>ru</b>	

			<ul style="list-style-type: none"> <li>• Доменная часть не может корректно быть сконvertирована в ACE (punycode);</li> <li>• Какая-либо часть домена начинается с дефиса или заканчивается дефисом;</li> <li>• Отсутствует домен второго уровня (кроме доменов из списка внутренних доменов предприятия, когда Фактор ставит тип домена «domainType» = «INTRA NET»)</li> </ul>		
8	DOMAIN _BAD_L ENGTH	<b>Некорректная длина доменного имени</b>	<p>Некорректная длина домена второго уровня. Правила для доменов второго уровня:</p> <ol style="list-style-type: none"> <li>1. Минимальная длина: <ul style="list-style-type: none"> <li>• .info, .pro — 3 символа;</li> <li>• все остальные — 2 символа.</li> </ul> </li> <li>2. Максимальная длина — 63 символа.</li> <li>3. Отдельно разрешены следующие однобуквенные домены: <ul style="list-style-type: none"> <li>• a.org;</li> <li>• e.org;</li> <li>• f.org;</li> <li>• i.org;</li> <li>• q.org;</li> <li>• u.org;</li> <li>• v.org;</li> <li>• x.org;</li> <li>• z.org;</li> <li>• b.pro;</li> <li>• q.com;</li> <li>• x.com;</li> <li>• z.com;</li> <li>• i.net.</li> </ul> </li> </ol>	test@ver ylongvery longveryv erylongve rylongv eryverylo ngverylo ngveryve rylongver ylongvery .com	Длина домена второго уровня превышает максимальн о допустимое значение
9	DOMAIN _NX_TLD	<b>Не существующи й домен первого уровня</b>	Несуществующий домен первого уровня: домен первого уровня не найден в списке доменов первого уровня (источник IANA)	vasya@m y.nonexis tent	
10	DOMAIN _BAD_S LD_IDN	<b>Некорректны й домен второго уровня</b>	Некорректная письменность в домене второго уровня (либо явно запрещена для домена первого уровня, либо смешение разных письменностей)	пример 1: user@ проверка .ru пример 2: user@	



				наполов инугusски .su	
11	DOMAIN _IP	Домен является прямым IP адресом	Домен является прямым IP адресом	l33t. haxor@0 xdeadbeef	
12	DOMAIN _SUSPI CIOUS_ SCRIPT	Сомнительна я письменность в доменной части	В доменной части используется письменность, не входящая в список: <ul style="list-style-type: none"> <li>• Латиница;</li> <li>• Кириллица;</li> <li>• Греческая письменность;</li> <li>• Армянская письменность;</li> <li>• Грузинская письменность;</li> <li>• Арабская письменность.</li> </ul>	call.me. lao@chin ese.??	
13	HAS_UN PARSED _EMAIL	В исходной строке не найден email	В исходной строке не найдено ни одного email	abc. example. com	
14	DOMAIN _TEMP	Одноразовый домен	Домен из списка одноразовых	badr3adf@ temp- mail.com	badr3adf@temp-mail. com Должен быть подключен справочник одноразовых email

# КОДЫ КАЧЕСТВА АДРЕСА

- Код валидации адреса
  - Проставление дополнительных кодов проверок
  - Отображение расширения валидатора в интерфейсе
  - Код проверки адреса (устаревшие значения)
- Код качества координат
- Код полноты адреса
- Маркер изменения адреса
- Маркер невалидности адреса
- Статус разбора адреса
  - Отображение в интерфейсе
  - Проставление статусов преобразований
- Соответствие уровню иерархии в ФИАС

Связанные страницы:

- Адрес
- Тип адреса

# КОД ВАЛИДАЦИИ АДРЕСА

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRME D_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил адрес и /или подтвердил его	Россия г Москва ул Абрамцевская	127572, Россия, г Москва, ул Абрамцевская, д.11, корп.1, кв. 228
3	SELECTED_ BY_KLADR_M ANUALLY	Выбран вручную из ФИАС	Выбран вручную из ФИАС	127572, Россия, г Москва, ул Абрамцевская, д.11, корп.1, кв. 228	127572, Россия, г Москва, ул Абрамцевская, д.11, корп.1, кв. 228
4	VALIDATED	Уверенное распознавание	Адрес распознан успешно	Россия г Москва ул Абрамцевская д.11 корп.1 кв. 228	127572, Россия, г Москва, ул Абрамцевская, д.11, корп.1, кв.228
5	VALIDATE_ D_HAS_UNPA RSED_PARTS	Разобрано, остались неразобранные части	Присутствует неразобранная часть, но при этом коды изменения говорят о том, что адрес разобран успешно	Россия 680000, к рай Хабаровски й, г Хабаровск, ул Истомина, дом 22а, пом.П (1-3)	Россия 680000, край Хабар овский, г Хабаровск, ул Истомина, дом 22а, пом.П <u>Сомнительная часть адреса: 1-3</u>
<b>6 На ручную обработку</b>					
7	NOT_VALI DATED_HAS_ UNPARSED_P ARTS	На проверку, неразобранные части	Для некоторых частей исходного адреса не было найдено соответствия в результате разбора	<u>иапиап</u> Россия г Москва ул Абрамцевская д.11 корп.1 кв. 228	127572, Россия, г Москва, ул Абрамцевская, д. 11, корп. 1, кв. 228 <u>Не разобрано: ИАПИАП</u>
8	NOT_VALI DATED_HAS_ AMBI	На проверку, неоднозначность	Имеется несколько вариантов разбора адреса. Неоднозначность может во зникать, например, из-за одинаковых названий улиц в городе	Россия, Республика Карелия, Петрозаводск, Соломенское шоссе, 11	185031, Россия, Республика Карелия, г Петрозаводск, ш Соломенское (р-н Пески), д. 11  Имеется несколько вариантов разбора адреса (улица находится в трех районах): р-н Соломенское, р-н Северная промзона, р-н Пески
9	NOT_VALI DATED_NOT_ ENOUGH_DA TA	Недостаточно данных для разбора	Результирующий адрес опустошён, коды изменения компонент пусты	670047	(пустое значение)
10	NOT_VALI DATED_DOU BTFUL_RESU LT	Сомнительный результат разбора	Результирующий адрес опустошён, неуверенное распознавание	633128 Краснообск СИБНИИЗХИМБ 346 кв. 46	(пустое значение)

11	NOT_VALI DATED_HAS_ WEAK_STREE T_WITHOUT_ PARENTS	На проверку, слабое распознавание улицы, отсутствуют административные единицы более высокого уровня	На проверку, слабое распознавание улицы, отсутствуют административные единицы более высокого уровня	Московская область Московский 45- 186 кв. 86	141601, Россия, обл Московская, г Клин, пер Московский, д. 45, корп. 186, кв. 86 Имеется иной вариант разбора адреса: Россия обл Московская г Наро- Фоминск г Верея проезд Московский
12	NOT_VALI DATED_HAS_ WEAK_NP_WI THOUT_PARE NTS	На проверку, слабый НП, отсутствуют административные единицы более высокого уровня	На проверку, слабое распознавание НП, отсутствуют административные единицы более высокого уровня	<u>с. Стерлибаш</u> <u>ул. Тукаево дом</u> 35 кв 17	453489, Россия, Респ Башкортостан, р-н Аургазинский, с Тукаево, д. 35, кв. 17 Наименование города отсутствует, не разобрано: <u>Стерлибаш ул</u>
13	NOT_VALI DATED_HAS_ WEAK_CITY_ WITHOUT_P ARENTS	На проверку, слабое распознавание города, отсутствуют административные единицы более высокого уровня	На проверку, слабое распознавание города, отсутствуют административные единицы более высокого уровня	141706 Горки д кв. 30	143100, Россия, обл Московская, г Руза, д Горки, кв. 30 Имеется несколько вариантов указания города: г Ступино, г Можайск, г Солнечногорск
14	NOT_VALI DATED_HAS_ AMBI_CITY_ WITHOUT_P ARENTS	На проверку, неоднозначное распознавание города, отсутствуют административные единицы более высокого уровня	На проверку, неоднозначное распознавание города, отсутствуют административные единицы более высокого уровня	110000 Чехова 58 Костанайская Костанайский Костанай Чехова 56 кв. 98	Россия, обл Московская, г Чехов, км 58-й, д. 56, кв. 98 Не разобрано: Костанайская Костанайский Костанай
15	NOT_VALI DATED_HAS_ UNACTUAL_ CITY_WITHO UT_PARENTS	На проверку, неактуальное название города, отсутствуют административные единицы более высокого уровня	На проверку, неактуальное название города, отсутствуют административные единицы более высокого уровня	660036 <u>Красная</u> <u>рск</u> Академгородок д. 50 стр. 28 Институт леса им В Н Сукачева кв. 11	660036, Россия, край Красноярский, г <u>Железногорск</u> , тер. СНТ N 11, д. 50, стр. 28 Не разобрано: Академгоро док, институт леса им В, Сукачева кв
16	NOT_VALI DATED_HAS_ FOREIGN_AD DRESS_WITH _MISPRINTS	На проверку, иностраннй адрес с опечатками	На проверку, иностраннй адрес с опечатками	060500 Атырауская область/Кзылж гинский район Миялы ул Т. Аубакирова кв. 94	Казахстан, обл Атырауская, р-н Кзылж гинский, с Миялы Не разобрано: 060500, ул Т. Аубакирова кв. 94
17	NOT_VALI DATED_HAS_ WEAK_HOUS E_PART_LET TER	На проверку, слабое распознавание дома	На проверку, слабое распознавание дома	Восточный СНТ., д.388, г. Гусь- Хрустальный, Владимирская обл., 601560	601560, Россия, обл Владимирская, р-н Гусь- Хрустальный, снт Восточный, д. 388г
18	NOT_VALI DATED_HAS_ POPULAR_ST REET_OR_NP _WITHOUT_P ARENT	На проверку, популярный адрес или НП без административной единицы более высокого уровня	На проверку, популярный адрес или НП без административной единицы более высокого уровня	Кировская обл., г. Кировстуденчес кий проезд д.12 кв.67	612820, Россия, обл Кировская, р-н Верхнекамский, г Кирс, ул Проезд, д. 12, кв. 67 <u>Сомнительная часть</u> <u>адреса:</u> <u>Кировстуденческий</u>

19	NOT_VALIDATED_HAS_STREET_IN_CITY_AREA_OR_STREET_IN_CITY	На проверку, неоднозначное распознавание улицы	На проверку, неоднозначное распознавание улицы	Россия, Московская, Ногинск, 14, 55, (полн: ул. Спортивная)	142409, Россия, обл Московская, г Ногинск, пгт Ногинск-9, ул Спортивная, д. 14, кв. 55 Не разобрано: ПОЛН
20	MISSED	Неизвестный код	Атрибут не проходил стандартизацию		

## Устаревшие коды

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Устаревшие</b>					
2	NOT_VALIDATED_INCORRECT_ZELENOGRAD_KORPUS	некорректный для Зеленограда корпус	В адресе указан город Зеленоград, не указана улица и указан дом или корпус с номером, меньшим, чем минимальный номер корпуса, найденного в текущей версии КЛАДР для пустой улицы		
<b>3 Исключены в релизе 20.3. Могут встречаться в адресах, созданных до 2020 года</b>					
4	NOT_VALIDATED_FOREIGN	Иностраннный адрес	Адрес был распознан в соответствии со справочниками иностранных адресов, но содержит дополнительную информацию, для которой соответствие не было найдено		
5	NOT_VALIDATED_HAS_NO_KEY_LADR_RECORD	На проверку, нет в адресном справочнике	В процессе распознавания, помимо базового адресного справочника, была также использована информация из дополнительных справочников ФАКТОР		
<b>6 Исключены в 2016 году. Могут встречаться у записей, созданных ранее</b>					
7	NOT_VALIDATED_HAS_ASSUMPTION	На проверку, предположение	Для улицы, населённого пункта или города результирующего адреса не найдено однозначного соответствия в исходном адресе		
8	NOT_VALIDATED_HAS_NO_MAIN_POINTS	На проверку, нет основных частей	Адрес не содержит ни города, ни населённого пункта, ни индекса		
9	NOT_VALIDATED_HAS_NUMBER_STREET_ASSUMPTION	На проверку, предположение по улице	Адрес не позволяет точно определить, что в нём указано – улица и номер дома или номерная улица		
10	NOT_VALIDATED_HAS_HOUSE_WITHOUT_STREET_OR_NUMBER	На проверку, нет улицы или НП	Адрес содержит номер дома, но не содержит улицы или населённого пункта, к которому он относится		
11	NOT_VALIDATED_HAS_EXTENSION_WITHOUT_HOUSE	На проверку, нет дома	Адрес содержит части расширения дома, но не содержит номер дома, к которому они относятся		
12	NOT_VALIDATED_EXCEEDED_HOUSE_NUMBER	На проверку, большой номер дома	Указан слишком большой номер дома		

13	NOT_VALIDATED_INCORRECT_HOUSE	На проверку, некорректный дом	Формат записи номера дома в результирующем адресе не соответствует стандарту ФАКТОРа
14	NOT_VALIDATED_INCORRECT_HOUSE_EXTENSION	На проверку, некорректное расширение дома	Формат записи номера одной из частей расширения дома в результирующем адресе не соответствует стандарту ФАКТОРа
15	NOT_VALIDATED_DICTIONARY	На проверку, не по справочнику	Адрес не является точно разобранным по справочнику

# ПРОСТАВЛЕНИЕ ДОПОЛНИТЕЛЬНЫХ КОДОВ ПРОВЕРOK

Логика проставление дополнительных кодов проверок на основе кодов изменения адресов.

- **VALIDATED\_HAS\_UNPARSED\_PARTS**  
Пример
- **NOT\_VALIDATED\_NOT\_ENOUGH\_DATA**
- **NOT\_VALIDATED\_DOUBTFUL\_RESULT**  
Пример

## VALIDATED\_HAS\_UNPARSED\_PARTS

Адрес разобран корректно, при этом имеется расширение валидатора.

Поле	Возможные значения кодов изменения
<b>index_changing_code</b>	<b>CORRECT; EMPTY</b>
<b>country_changing_code</b>	<b>TYPE; CORRECT</b>
<b>subject_changing_code</b>	<b>CORRECT; TYPE; CHANGE_LEVEL; MISPRINT</b>
<b>district_changing_code</b>	<b>CORRECT; EMPTY; CHANGE_LEVEL; TYPE</b>
<b>city_changing_code</b>	<b>CORRECT; REPLACE; MISPRINT; CHANGE_LEVEL; TYPE</b>
<b>settlement_changing_code</b>	<b>CORRECT; EMPTY; MISPRINT; CHANGE_LEVEL; TYPE</b>
<b>street_changing_code</b>	<b>CORRECT; TYPE; MISPRINT</b>
<b>house_changing_code</b>	<b>CORRECT; TYPE</b>
<b>validation_status</b>	<b>NOT_VALIDATED_HAS_UNPARSED_PARTS</b>

### Пример

В адресе «РОССИЯ,680000,край Хабаровский,г Хабаровск,ул Истомина,дом 22а,,пом.П (1-3)» в неразобранную часть уйдет «Пом. П», но при этом адрес разобран успешно.

## NOT\_VALIDATED\_NOT\_ENOUGH\_DATA

Недостаточно данных для разбора, все компоненты адреса пусты.

Поле	Возможные значения
<b>index_changing_code</b>	EMPTY;
<b>country_changing_code</b>	EMPTY;
<b>subject_changing_code</b>	EMPTY;
<b>district_changing_code</b>	EMPTY;
<b>city_changing_code</b>	EMPTY;
<b>street_changing_code</b>	EMPTY;
<b>validation_status</b>	любой

## NOT\_VALIDATED\_DOUBTFUL\_RESULT

Сомнительный результат разбора.

Поле	Возможные значения
<b>index_changing_code</b>	<b>!=CORRECT; !=REPLACE</b>
<b>country_changing_code</b>	OMIT; EMPTY;
<b>subject_changing_code</b>	OMIT; EMPTY; RENAME
<b>district_changing_code</b>	OMIT; EMPTY; RENAME
<b>city_changing_code</b>	OMIT; EMPTY
<b>settlement_changing_code</b>	OMIT; EMPTY; TYPE
<b>street_changing_code</b>	OMIT; EMPTY;TYPE; DEFAULT
<b>validation_status</b>	любой
all_changes	не пусто

### Пример

«м Калужская первый вагон из центра, через длинный переход, выход направо, одну ост в строн от центра, перейти профсоюзную улицу 5 мин. до бензозаправки ТНК за ней здание Книга эксперт, 3 этаж» ? «644000 Россия обл Омская г Омск пгт Первый Отряд ВПО ВПЧ-15 3» — Вариант разбора сомнителен, подобные адреса зануляются, и им проставляется соответствующий код.



# ОТОБРАЖЕНИЕ РАСШИРЕНИЯ ВАЛИДАТОРА В ИНТЕРФЕЙСЕ

В зависимости от кода проверки адреса, расширение валидатора в интерфейсе называется по-разному.

1. Если адрес имеет статус проверки NOT\_VALIDATED\_HAS\_AMBI, поле называется «Варианты разбора», и каждый вариант разбора отображается на новой строке. Если вариантов больше фиксированного числа, отображается многоточие.

Тип: Фактический | Основной

Исходный: 119607 МОСКВА ВЕРНАДСКОГО ПР-КТ 29 ОФ.1202 КВ. 34

Код качества: ■ Пригоден для почтовой рассылки

Код проверки: ■ На проверку, неоднозначность, дом не найден в ФИАС

Варианты разбора:

- 108830 РОССИЯ Г МОСКВА П ВОРОНОВСКОЕ КВ-Л 34
- 108818 РОССИЯ Г МОСКВА П ДЕСЕНОВСКОЕ КВ-Л 34
- 108829 РОССИЯ Г МОСКВА П КЛЕНОВСКОЕ КВ-Л 34
- ...

108811 | Россия | Центральный

г	Москва
п	Внуковское
Тип	Город
Тип	Район города
Тип	Населённый пункт
кв-л	34

дом: 29 | к: | стр: | кв: | Индекс(ФИАС): 108811

Офис: 1202

2. В иных случаях, если расширение валидатора непустое, поле называется «Не разобрано» и показывает несвалидированную часть адреса.

Тип: Фактический | Основной

Исходный: 385730 ТУЛЬСКИЙ, МАЯКОВСКИЙ РАЙОН, РЕСПУБЛИКА АДЫГЕЯ ПЕРВОМАЙСКАЯ УЛИЦА, Д.400 КВ. 92

Код качества: ■ Пригоден для почтовой рассылки

Код проверки: ■ На проверку, неразобранные части

Не разобрано: МАЯКОВСКИЙ, РАЙОН

385730 | Россия | Южный

Респ	Адыгея
р-н	Майкопский
Тип	Город
Тип	Район города
п	Тульский
ул	Первомайская

дом: 400 | к: | стр: | кв: 92 | Индекс(ФИАС): 385730

Часовой пояс: Московское время (MSK)

# КОД ПРОВЕРКИ АДРЕСА (УСТАРЕВШИЕ ЗНАЧЕНИЯ)

## Устаревшие коды проверки адреса

№	Литера	Локализация	Расшифровка
1			Исключены в релизе 20.3. Могут встречаться в адресах, созданных до 2020 года
2	NOT_VALIDATE D_ FOREIGN	Иностраннный адрес	Адрес был распознан в соответствии со справочниками иностранных адресов, но содержит дополнительную информацию, для которой соответствие не было найдено
3	NOT_VALIDATE D_ HAS_NO_KLADR _RECORD	На проверку, нет в адресном справочнике	В процессе распознавания, помимо базового адресного справочника, была также использована информация из дополнительных справочников ФАКТОР
4			Исключены в 2016 году. Могут встречаться у записей, созданных ранее
5	NOT_VALIDATE D_ HAS_ASSUMPTION	На проверку, предположение	Для улицы, населённого пункта или города результирующего адреса не найдено однозначного соответствия в исходном адресе
6	NOT_VALIDATE D_ HAS_NO_MAIN_POINTS	На проверку, нет основных частей	Адрес не содержит ни города, ни населённого пункта, ни индекса
7	NOT_VALIDATE D_ HAS_NUMBER_STREET_ASSUMPTION	На проверку, предположение по улице	Адрес не позволяет точно определить, что в нём указано – улица и номер дома или номерная улица
8	NOT_VALIDATE D_ HOUSE_WITHOUT_STREET_OR_NP	На проверку, нет улицы или НП	Адрес содержит номер дома, но не содержит улицы или населённого пункта, к которому он относится
9	NOT_VALIDATE D_ HOUSE_EXTENSION_WITHOUT_HOUSE	На проверку, нет дома	Адрес содержит части расширения дома, но не содержит номер дома, к которому они относятся
10	NOT_VALIDATE D_ EXCEEDED_HOUSE_NUMBER	На проверку, большой номер дома	Указан слишком большой номер дома
11	NOT_VALIDATE D_ INCORRECT_HOUSE	На проверку, некорректный дом	Формат записи номера дома в результирующем адресе не соответствует стандарту ФАКТОРА
12	NOT_VALIDATE D_ _	На проверку, некорректное расширение дома	Формат записи номера одной из частей расширения дома в результирующем адресе не соответствует стандарту ФАКТОРА

---

INCORRECT\_HO  
USE\_EXTENSION

---

13 NOT\_VALIDATE На проверку, не по Адрес не является точно разобранным по справочнику  
D\_ справочнику  
DICTIONARY

---

# КОД ПОЛНОТЫ АДРЕСА

№	Литера	Локализация	Описание	Исходно	Стандартизировано
1	GOOD	Пригоден для почтовой рассылки	Пригоден для почтовой рассылки	Россия г Москва ул Абрамцевская д.11 корп.1 кв. 228	127572, Россия, г Москва, ул Абрамцевская, д. 11, корп. 1, кв. 228
2	ON_DEM AND	До востребования	В адресе указаны индекс, город и фраза «до востребования»	Москва, 129626, до востреб	129626, Россия, г Москва
3	POSTAL_BOX	Абонентский ящик	В адресе указан абонентский ящик и нет дома /квартиры	Москва, Турчанинов пер, 129626, а /я 33	119034, Россия, г Москва, пер Турчанинов, а/я 33
4	UNDEF_01	Не определен регион	По исходным компонентам адреса нельзя определить регион	Солидарности 97а	(пустое значение)
5	UNDEF_02	Не определен город или НП	Город или населенный пункт не могут быть определены по исходному адресу (определены только регион и район)	Область московская район истринский	Россия, обл Московская, р-н Истринский
6	UNDEF_03	Не определена улица	В исходном адресе не найдена улица, при этом населённый пункт имеет тип «город»	Екатеринбург	620000, Россия, обл Свердловская, г Екатеринбург
7	UNDEF_04	Не определен номер дома	В исходном адресе не найден номер дома	180725, Псковская обл., д.Илово	180725, Россия, обл Псковская, р-н Себежский, д Илово
8	UNDEF_05	Не определена квартира /офис	Адрес не содержит цифр, означающих номер квартиры, при этом населенный пункт имеет тип «город»	Москва Спартаковская дом 2	105066, Россия, г Москва, ул Спартаковская, д. 2
9	UNDEF_06	Не определен	Другая причина неопределенности адреса. Пока что не ставим никаким адресам. Но если в ФИАС появятся новые уровни, может появиться и это значение		
10	UNDEF_07	Иностранный адрес	Иностранный адрес	375010 Ереван Кокбацц 29	Армения, г Ереван
11	MISSED	Неизвестный код	Атрибут не проходил стандартизацию		

# МАРКЕР ИЗМЕНЕНИЯ АДРЕСА

Маркер изменения формируется с указанием компонента, который был изменен. Например: **CITY\_TYPE** — восстановили тип города «город» (в маркере указаны два компонента через нижнее подчеркивание — **CITY** и **TYPE**), **COUNTRY\_OMIT** — страна пропущена в исходном адресе, в результирующем адресе присутствует (в маркере указаны два компонента через нижнее подчеркивание — **COUNTRY** и **OMIT**).

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	CORRECT	Компонент указан корректно в исходном адресе	Компонент указан корректно в исходном адресе	Россия Москва	101000, Россия, г Москва	<ul style="list-style-type: none"> <li><b>COUNTRY_CORRECT</b> — страна указана корректно</li> <li><b>CITY_CORRECT</b> — город указан корректно</li> </ul> <p>Прочие маркеры изменения адреса во всех примерах опущены для краткости (указаны только те маркеры, которые описывают рассматриваемую ситуацию)</p>
2	EMPTY	Компонента нет в результирующем адресе	Компонента нет в результирующем адресе	Россия Москва пер Турчан инов	119034, Россия, г Москва, пер Турчанинов	<ul style="list-style-type: none"> <li><b>DISTRICT_EMPTY</b> — нет наименования района</li> <li><b>SETTLEMENT_EMPTY</b> — нет наименования населенного пункта</li> <li><b>HOUSE_EMPTY</b> — нет дома</li> </ul>
3	OMIT	Компонент пропущен в исходном адресе, в результирующем — присутствует	Компонент пропущен в исходном адресе, в результирующем — присутствует	Москва пер Турчан инов	119034, Россия, г Москва, пер Турчанинов	<b>COUNTRY_OMIT</b> — компонент «страна» пропущен в исходном адресе, в результирующем присутствует
4	MISPRI NT	Исправили опечатку в названии компонента	Исправили опечатку в названии компонента	119034 Россия Москва г пер Турча инов	119034, Россия, г Москва, пер Турчани нов	<b>STREET_MISPRINT</b> — исправили опечатку в названии улицы
5	TYPE	Восстановили или исправили тип компонента	Восстановили или исправили тип компонента	пример 1: Россия Москва Турчанин ов  пример 2: Чернь ул. Ленина, 17/24	пример 1: 119034, Россия, г Москва, пер Турчанинов  пример 2: 301090, Россия, обл Тульская, р-н Чернский, рп Чернь, ул Ленина, д. 17/24	<p>для примера 1:</p> <ul style="list-style-type: none"> <li><b>SUBJECT_TYPE</b> — восстановили тип субъекта (региона) «город»</li> <li><b>CITY_TYPE</b> — восстановили тип города «город» для компонента «Москва»</li> </ul>

- **STREET\_TYPE** — восстановили тип улицы «переулок» для компонента «Турчанинов»

для примера 2:

- **SETTLEMENT\_TYPE** — восстановили тип населённого пункта «рп» (рабочий посёлок)
- **HOUSE\_TYPE** — восстановили тип здания «д.» (дом)

6	DEFAULT	Город или улица были восстановлены из списка по умолчанию	Улица и/или город была восстановлена по умолчанию, в исходном адресе отсутствует	Калужская Область, Перемышльский район, село Рождественно, улица Парковая, дом 6 кв. 42	249123, Россия, обл Калужская, р-н Перемышльский, с Рождественно, ул Парковая, д. 6, кв. 42	<p><b>STREET_DEFAULT</b> Улица была восстановлена по умолчанию, в исходном адресе отсутствует.</p> <p>Если на вход будет передан адрес, по которому можно однозначно определить разбор, то произойдет разбор адреса независимо от значения настройки «город по умолчанию». Если же адрес не может быть определён однозначно, то фильтр будет искать его в городах из списка по умолчанию. Город и улицу по умолчанию можно настраивать</p>
7	RENAME	Актуализовали устаревшее наименование	Актуализовали устаревшее наименование	190000, Россия, г Ленинград	190000, Россия, г Санкт-Петербург	<ul style="list-style-type: none"> <li>• <b>SUBJECT_RENAME</b> — актуализовали субъект (регион) «Санкт-Петербург»</li> <li>• <b>CITY_RENAME</b> — актуализовали наименование города (Ленинград ? Санкт-Петербург)</li> </ul>
8	REPLACE У	У выбранного компонента и актуализованного отличаются родители	У выбранного компонента и актуализованного отличаются родители	Челябинская область, г Копейск, ул 26 Партсъезда	456658, Россия, обл Челябинская, г Копейск, ул 26 Партсъезда	<p>Населенный пункт (поселок Октябрьский) стал городом (г Копейск). В результате чего компоненты адреса имеют различных «родителей»:</p> <ul style="list-style-type: none"> <li>• <b>SETTLEMENT_REPLACE</b> (у исходного адреса населенный пункт (п. Октябрьский) принадлежит городу Копейск, а у актуальной версии адреса населенный пункт (г Копейск) принадлежит</li> </ul>

						Челябинской области (в результате актуализации адреса населенный пункт «поселок Октябрьский» стал городом «Копейск»))
						<ul style="list-style-type: none"> <li>• <a href="#">STREET_REPLACE</a> (у исходного адреса улица принадлежит посёлку Октябрьский, а у актуальной версии городу Копейск)</li> </ul>
9	<a href="#">CHANGE_LEVEL</a>	Подобранный компонент после актуализации переехал на другой уровень в ФИАС	Подобранный компонент после актуализации переехал на другой уровень в ФИАС	Челябинск ая, п Октябрьск ий ул 26 Партсъезда	456658, Россия, обл Челябинская, г Копейск, ул 26 Партсъезда	<a href="#">SETTLEMENT_CHANGE_LEVEL</a> — населенный пункт (поселок Октябрьский) стал городом (г Копейск), в результате чего населенный пункт переехал на другой уровень в ФИАС
10	<a href="#">REDUNDANT</a>	Части подобранного компонента в исходном адресе разделены неиспользованными словами и/или запятой	Части подобранного компонента в исходном адресе разделены неиспользованными словами и/или запятой	Город Верхняя Пышма,, Улица <b>Витамин</b> ,3	Россия, обл Свердловская, г Верхняя Пышма, п Черемшанка, ул 3-я	<a href="#">STREET_REDUNDANT</a> , неразобранная часть «ВИТАМИН»
11	<a href="#">RENAME ( )</a>	Индекс в исходном адресе совпадает с одним из предыдущих (неактуальных) для этой записи. В результате вернули актуальный	Индекс в исходном адресе совпадает с одним из предыдущих (неактуальных) для этой записи. В результате вернули актуальный			<a href="#">INDEX_RENAME</a>
12	<a href="#">CHANGE ( )</a>	Изменение в первых двух цифрах индекса	Изменение в первых двух цифрах индекса, которые указывают на принадлежность к региону	443011 Нижний Новгород пр-кт Ленина 1	603011, Россия, обл Нижегородская, г Нижний Новгород, пр-кт Ленина, д. 1	<a href="#">INDEX_CHANGE</a>
13	<a href="#">CLARIFY Y ( )</a>	Уточнен почтовый индекс	Две или три последних цифры «0» в почтовом индексе (т.е. «XXXX00» или «XXX000») были уточнены	603000 Нижний Новгород пр-кт Ленина 1	603011, Россия, обл Нижегородская, г Нижний Новгород, пр-кт Ленина, д. 1	<a href="#">INDEX_CLARIFY</a>
14	<a href="#">MISPRINT ( )</a>	Уточнен почтовый индекс	У измененного почтового индекса совпали как	603036 Нижний Новгород		<a href="#">INDEX_MISPRINT</a>

---

минимум две первые цифры (кот орые указывают на принадлежность к региону) с исходным почтовым индексом	пр-кт Ленина 1	603011, Россия, обл Нижегородская, г Нижний Новгород, пр-кт Ленина, д. 1
--	-------------------	---

---

**Проставляется для:**

- индекса,
- страны,
- субъекта,
- района,
- города,
- населенного пункта,
- планировочной структуры, причём по умолчанию код изменения планировочной структуры выводится ещё и в коде той же компоненты, что и наименование.
- улицы,
- домовая части.



# МАРКЕР НЕВАЛИДНОСТИ АДРЕСА

№	Литера	Описание
1	NOT_VALIDATED_HAS_UNPARSED_PARTS	На проверку, есть неразобранные части
2	NOT_VALIDATED_HAS_AMBI	На проверку, неоднозначность
3	NOT_VALIDATED_HAS_WEAK_STREET_WITHOUT_PARENTS	На проверку, слабое распознавание улицы, отсутствуют административные единицы более высокого уровня
4	NOT_VALIDATED_HAS_WEAK_NP_WITHOUT_PARENTS	На проверку, слабый НП, отсутствуют административные единицы более высокого уровня
5	NOT_VALIDATED_HAS_WEAK_CITY_WITHOUT_PARENTS	На проверку, слабое распознавание города, отсутствуют административные единицы более высокого уровня
6	NOT_VALIDATED_HAS_AMBI_CITY_WITHOUT_PARENTS	На проверку, неоднозначное распознавание города, отсутствуют административные единицы более высокого уровня
7	NOT_VALIDATED_HAS_UNACTUAL_CITY_WITHOUT_PARENTS	На проверку, неактуальное название города, отсутствуют административные единицы более высокого уровня
8	NOT_VALIDATED_HAS_FOREIGN_ADDRESS_WITH_MISPRINTS	На проверку, иностранный адрес с опечатками
9	NOT_VALIDATED_HAS_WEAK_HOUSE_PART_LETTER	На проверку, слабое распознавание дома
10	<b>NOT_VALIDATED_HAS_POPULAR_STREET_OR_NP_WITHOUT_PARENT</b>	На проверку, популярный адрес или НП без административной единицы более высокого уровня

# КОД КАЧЕСТВА КООРДИНАТ

Код показывает степень точности найденных географических координат. Работает при наличии справочника координат.

№	Литера	Локализация	Описание	Исходно	Описание примера
1	EXACT_HOUSE	Координаты с точностью до расширения дома	Координаты с точностью до расширения дома	г Майкоп, ул Промышленная, д 46А	Координаты найдены с точностью до дома. Широта: 44.6250764 Долгота: 40.097972
2	NEAREST_HOUSE	Координаты с точностью до номера дома, расширение (корпус, строение, литера) отличается	Координаты найдены с точностью до дома (расширение отличается) по соседнему дому (ближайший номер дома с такой же кратностью. Если существует два таких номера дома, то выбирается тот, что меньше)	г Майкоп, ул Промышленная, д 46Б	Координаты найдены с точностью до номера дома, корпус «Б» не найден. Широта: 40.097972 Долгота: 44.6250764
3	STREET	Координаты найдены с точностью до улицы	Координаты с точностью до улицы или планировочной структуры	г Майкоп, ул Промышленная, д 47А	Координаты найдены с точностью до улицы, дом «47А» не найден. Широта: 40.1088668 Долгота: 44.6234436
4	SETTLEMENT	Координаты найдены с точностью до населённого пункта	Координаты найдены с точностью до населённого пункта	Тамбовская обл, Жердевский р-н, поселок Садовый, ул Вишневая, д 10	Координаты найдены с точностью до населённого пункта (поселение Садовый). Широта: 51.8328893 Долгота: 41.380118
5	CITY	Координаты найдены с точностью до города	Координаты найдены с точностью до города	Россия, г Краснодар, Парк культуры	Координаты найдены с точностью до города Краснодар. Широта: 38.9759647 Долгота: 45.0401604
6	NOT_FOUND	Координаты для адреса не найдены	Координаты для адреса не найдены	Россия Башкортостан респ Буздякский р-н рзд 1417 км	Координаты для адреса не найдены. Широта: (пустое поле) Долгота: (пустое поле)
7	NOT_CHECKED	Справочник недоступен	Справочник с координатами недоступен	г Майкоп, ул Промышленная, д 46А	Если справочник географических координат недоступен, то в поле «точность гео» указывается значение «справочник недоступен»
8	ORIGINAL	Использованы оригинальные координаты из источника	Использованы оригинальные координаты из источника	г Майкоп, ул Промышленная, д 46А Широта: 44.6250764 Долгота: 40.097972	Использованы оригинальные координаты из источника

---

9	MISSED	Неизвестно	Неизвестно	г Майкоп, ул Промышлен ная, д 46А	Если справочник географических координат не подключен, то в поле «точность гео» указывается значение «неизвестно»
---	--------	------------	------------	---	---

---

# СТАТУС РАЗБОРА АДРЕСА

Статусы преобразований показывают, до какого уровня «Единый клиент» уверен в разборе. Адреса с такими статусами можно использовать в случаях, когда не нужен точный адрес до дома, а достаточно определенного уровня.

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	CITY_BY_INDEX	Адрес до города разобран по индексу	Адрес до города разобран по индексу	670047 Улица Иртышская,6	670047, Россия, Респ Бурятия, г Улан-Удэ, ул Иртышская, д. 6	Адрес до города Улан-Удэ разобран по индексу 670047
2	STREET_BY_INDEX	Адрес до улицы разобран по индексу	Адрес до улицы разобран по индексу. <i>В настоящее время невозможно разобрать адрес до улицы по индексу. В новых версиях справочника ФИАС (от ГАР) индексов у улиц нет. Справочник содержит индексы только для домов</i>	—	—	—
3	HOUSE_LEVEL	Адрес уверенно разобран до дома	Адрес уверенно разобран до дома	125009 Россия город Москва пл Тверская д. 26/1	125009, Россия, г Москва, пл Тверская, д. 26/1	Уверенное распознавание, дом не найден в ФИАС
4	STREET_LEVEL	Адрес уверенно разобран до улицы	Адрес уверенно разобран до улицы	125009 Россия город Москва пл Тверская	125009, Россия, г Москва, пл Тверская	Адрес уверенно разобран до улицы Тверская
5	SETTLEMENT_LEVEL	Адрес уверенно разобран до населенного пункта	Адрес уверенно разобран до населенного пункта	Россия г Нижний Тагил тер Стр Урожай 23	Россия, обл Свердловская, г Нижний Тагил, тер Стр Урожай, д. 23	Уверенное распознавание, дом не найден в ФИАС
6	CITY_LEVEL	Адрес уверенно разобран до города	Адрес уверенно разобран до города	125009 Россия город Москва	125009, Россия, г Москва	Адрес уверенно разобран до города Москва
7	DISTRICT_LEVEL	Адрес уверенно разобран до региона	Адрес уверенно разобран до региона	Россия Республика Башкортостан	Россия, Респ Башкортостан	Адрес уверенно разобран до региона Башкортостан
8	SUBJECT_LEVEL	Адрес уверенно разобран до	Адрес уверенно разобран до субъекта	94800 Украина, Луганская область г.	Украина, обл Луганская, г Должанск	Адрес уверенно разобран

		субъекта		Свердловск ул. Безребельног о 7-а кв.7		до субъекта Не разобрано: 94800, ул. Безребель ного 7-а кв.7
9	COUNTRY_LEVEL	Адрес уверенно разобран до страны	Адрес уверенно разобран до страны	Россия	Россия	Адрес уверенно разобран до страны Россия

## ОТОБРАЖЕНИЕ В ИНТЕРФЕЙСЕ

Статус преобразований расположен над кодом качества. Поле не отображается, если на основе кодов изменений ни один из настроенных статусов не сработал.

Тип	Фактический ▾	<input type="checkbox"/> Основной							
Исходный	РОССИЯ 115093 ГОРОД МОСКВА ГОРОД МОСКВА УЛИЦА ПАВЛОВСКАЯ 📄								
Статус преобразований	Адрес уверенно разобран до улицы								
Код качества	■ Не определен номер дома								
Код проверки	■ Уверенное распознавание								
	115093	Россия	Центральный						
г	Москва								
Тип	Район								
г	Москва								
Тип	Район города								
Тип	Населённый пункт								
ул	Павловская								
дом		к		стр		кв		Индекс(ФИАС)	115093

Тип	Фактический ▾	<input type="checkbox"/> Основной							
Исходный	606900, НИЖЕГОРОДСКАЯ ОБЛАСТЬ, ШАХУНСКИЙ РАЙОН, РП ВАХТАН, УЛ КАРПОВСКАЯ, Д 22А КВ. 36 📄								
Статус преобразований	Адрес до города разобран по индексу								
Код качества	■ Пригоден для почтовой рассылки								
Код проверки	■ Уверенное распознавание								
	606900	Россия	Приволжский						
обл	Нижегородская								
Тип	Район								
г	Шахунья								
Тип	Район города								
рп	Вахтан								
ул	Карповская								
дом	22А	к		стр		кв	36	Индекс(ФИАС)	606900

# ПРОСТАВЛЕНИЕ СТАТУСОВ ПРЕОБРАЗОВАНИЙ

Статус преобразований заполняется на основе набора маркеров и позволяет рассказать больше об адресе.

## CHANGES\_STATUS

Локализация	Code	Changing Codes							
		index	country	subject	district	city	settlement	street	house
Адрес до города разобран по индексу	CITY_BY_INDEX	CORRECT	OMIT			OMIT			
Адрес до улицы разобран по индексу	STREET_BY_INDEX	CORRECT	OMIT			OMIT		OMIT	
Адрес разобран до дома	HOUSE_LEVEL	CORRECT	TYPE	CORRECT	CORRECT	CORRECT	CORRECT	CORRECT	CORRECT
		EMPTY	CORRECT	TYPE	EMPTY	REPLACE	EMPTY		TYPE
		RENAME	MISPRINT	CHANGE_LEVEL	CHANGE_LEVEL	MISPRINT	MISPRINT		MISPRINT
				MISTPINT	TYPE	CHANGE_LEVEL	CHANGE_LEVEL		
		CLARIFY				TYPE	TYPE		
		MISPRINT							
Адрес уверенно разобран до улицы	STREET_LEVEL	CORRECT	TYPE	CORRECT	CORRECT	CORRECT	CORRECT	CORRECT	HE CORRECT
		EMPTY	CORRECT	TYPE	EMPTY	REPLACE	EMPTY		HE TYPE
		RENAME	MISPRINT	CHANGE_LEVEL	CHANGE_LEVEL	MISPRINT	MISPRINT		MISPRINT
				MISTPINT	TYPE	CHANGE_LEVEL	CHANGE_LEVEL		
		CLARIFY				TYPE	TYPE		
		MISPRINT							
Адрес уверенно разобран до населенного пункта	SETTLEMENT_LEVEL	CORRECT	TYPE	CORRECT	CORRECT	CORRECT	CORRECT	EMPTY	
		EMPTY	CORRECT	TYPE	EMPTY	REPLACE	MISPRINT	OMIT	
		RENAME	MISPRINT	CHANGE_LEVEL	CHANGE_LEVEL	MISPRINT	CHANGE_LEVEL		
				MISTPINT	TYPE	CHANGE_LEVEL	TYPE		
		CLARIFY				TYPE			

			MISP RINT						
Адрес уверенно разобран до города	CITY_LEVEL	CORRECT	TYPE	CORRECT	CORRECT	CORRECT	EMPTY	EMPTY	
		EMPTY	CORRECT	TYPE	EMPTY	REPLACE	OMIT	OMIT	
		RENAME	MISP RINT	CHANGE _LEVEL	CHANGE _LEVEL	MISPRINT			
		CLARIFY		MISTPINT	TYPE	CHANGE _LEVEL			
							TYPE		
				MISP RINT					
Адрес уверенно разобран до региона	DISTRICT_LEVEL	CORRECT	TYPE	CORRECT	CORRECT	EMPTY	EMPTY	EMPTY	
		EMPTY	CORRECT	TYPE	EMPTY	OMIT	OMIT	OMIT	
		RENAME	MISP RINT	CHANGE _LEVEL	CHANGE _LEVEL				
		CLARIFY		MISTPINT	TYPE				
				MISP RINT					
Адрес уверенно разобран до субъекта	SUBJECT_LEVEL	CORRECT	TYPE	CORRECT	EMPTY	EMPTY	EMPTY	EMPTY	
		EMPTY	CORRECT	TYPE	OMIT	OMIT	OMIT	OMIT	
		RENAME	MISP RINT	CHANGE _LEVEL		RENAME			
		CLARIFY		MISTPINT					
				MISP RINT					
Адрес уверенно разобран до страны	COUNTRY_LEVEL	CORRECT	TYPE	EMPTY	EMPTY	EMPTY	EMPTY	EMPTY	
		EMPTY	CORRECT	OMIT	OMIT	OMIT	OMIT	OMIT	
		RENAME	MISP RINT	RENAME		RENAME			
		CLARIFY							
				MISP RINT					



# СООТВЕТСТВИЕ УРОВНЮ ИЕРАРХИИ В ФИАС

№	Литера	Уровень, до которого адрес соответствует ФИАС	Номер уровня по ФИАС
1	<i>FIAS_COUNTRY</i>	страна (в адресе указана только страна Россия)	0
2	<i>FIAS_SUBJECT</i>	регион	1
3	<i>FIAS_DISTRICT</i>	район	3
4	<i>FIAS_CITY</i>	город	4
5	<i>FIAS_CITY_AREA</i>	внутригородской район	5
6	<i>FIAS_SETTLEMENT</i>	населённый пункт	6
7	<i>FIAS_ADDITIONAL_TERRITORY</i>	дополнительный адресообразующий элемент	90
8	<i>FIAS_ADDITIONAL_DEPENDENT</i>	подчиненный дополнительный адресообразующий элемент	91
9	<i>FIAS_PLANNING_STRUCTURE</i>	планировочная структура	65
10	<i>FIAS_AUTONOMY</i>	автономия (не заполняется в ФИАС, неактуальный уровень)	2
11	<i>FIAS_LOCATION</i>	городские и сельские поселения (пока не заполняется в ФИАС)	35
12	<i>FIAS_LAND_LOT</i>	земельный участок (пока не заполняется в ФИАС)	75
13	<i>FIAS_STREET</i>	улица	7
14	<i>FIAS_HOUSE</i>	дом (точное совпадение дома).  Допущение: дом с номером «4А» совпадёт с записью в ФИАС «дом 4 литер А»	-
15	<i>FIAS_RANGE_HOUSE</i>	дом из диапазона номеров домов (точное попадание в номер из диапазона)	-
16	<i>EMPTY</i>	пустой адрес	-
17	<i>FOREIGN_ADDRESS</i>	иностраннный адрес	-
18	<i>UNKNOWN_ADDRESS</i>	адрес получен из дополнительных справочников или страна для адреса не указана	-

# КОДЫ КАЧЕСТВА ТЕЛЕФОНА

- Код качества телефона
- Маркеры проверок и изменений телефона
- Причины, почему телефон не распознан

Связанные страницы (классификаторы полей телефона):

- Мобильность телефона
- Тип телефона

# КОД КАЧЕСТВА ТЕЛЕФОНА

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	<b>Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил номер телефона и/или подтвердил его	7 495 4	7 495 4167425	Любые данные, подтвержденные контролером, имеют статус «CONFIRMED_MANUALLY»
3	GOOD	Корректный телефон	Все проверки прошли, телефон корректный	4954167425	7 495 4167425	Телефон распознан гарантированно. Цифры «7», «+7», «8», «+8» в начале номера указывать необязательно
4	GOOD_REPLACE_CODE	Изменен код	Старый код города или региона заменен на новый согласно схеме модификации телефонных кодов	(0742) 436666	7 (4742) 436666	Старый код города Липецк «0742» был заменен на новый «4742»
5	GOOD_REPLACE_NUMBER	Изменен номер	Телефонный номер заменен согласно схеме модификации АТС	495 2845 554 дата актуальности: 17.01.2004	7 (495) 6845554	Первая цифра телефонного номера «2» была заменена на «6» при указанной дате актуальности, меньшей даты проведения перенумерации (т.е. меньшей 19.01.2004)
6	GOOD_REPLACE_CODE_NUMBER	Изменен код и номер	Система заменила код и номер телефона согласно схеме модификации АТС	(34392) 47649	7 3439 347649	Старый код города «34392» заменен на новый «3439», изменена первая цифра в номере телефона (согласно схеме модификации АТС)
7	GOOD_CITY_CONFLICT	Конфликт по городу	Телефон распознан, обнаружен конфликт по городу	(8-47148) 2-46-14 адрес: г. Курск	7 47148 24614	Выделенный код города «47148» относится к городу Железнодорожск и не совпадает с кодом города Курска, определенным по адресу
8	GOOD_REGION_CONFLICT	Конфликт по региону	Телефон распознан, обнаружен конфликт регионов	499-9559759 адрес: г. Волгоград	7 499 9559759	Выделенный код региона (г. Москва) не совпадает с кодом региона, определенным по адресу (г. Волгоград)
9	FOREIGN	Иностраннй телефон	Задан иностранный адрес. Если код города не указан, то он будет определен по адресу	2223345 адрес: г. Киев	380 44 2223345	

---

**10 На ручную обработку**

---

11	CODE_A MBI	Неоднозначный код	Код города для телефона не удается восстановить однозначно	9559759 адрес: г. Москва	7 495 9559759	Номер «9559759» существует в справочнике Росвязи с кодами «495» и «499»
12	GARBAGE	Убран мусор	Телефон содержал «мусорные» символы, которые были убраны при очистке	49570577 12 спросить Олега	7 495 7057712	
13	GOOD_CITY	Восстановлен город	Код города был определен по адресу	51-20-86 адрес: г. Курск	7 4712 512086	Код города был определен по адресу (г. Курск)
14	GOOD_EXTRA_PHONE	Данные содержат более одного телефона	Исходные данные содержат более одного телефона, выделен первый	(499) 262-99-01 (499) 262-90-95	7 499 2629901	
15	OUT_OF_RANGE	Номер не найден в справочнике Росвязи	Номер отсутствует в справочнике Росвязи	35243110 20	7 35243 11020	
16	UNDEF	Некорректный телефон	Телефон не может быть распознан	927123100	(пустое значение)	
17	MISSED	Неизвестный код	Атрибут не проходил стандартизацию			
18	EMPTY	Пустой исходный телефон	Исходная строка пуста или не содержит цифр			

---

# МАРКЕРЫ ПРОВЕРОК И ИЗМЕНЕНИЙ ТЕЛЕФОНА

Может быть несколько значений

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	REPLAC ED_CODE	Изменен код	Старый код города или региона заменен на новый согласно схеме модификации телефонных кодов	70957755 068	7 495 7755068	
2	REPLAC ED_NUM BER	Изменен номер	Телефонный номер заменен согласно схеме модификации АТС. Первая цифра телефонного номера «2» был заменена на «6» при указанной дате актуальности, меньшей даты проведения перенумерации (т.е. меньшей или равной 19.01.2004)	телефон: 495 2845 554 дата актуальности: 19.01.2004	телефон: 7 495 68455 54 дата актуальности: 19.01.2004	Первая цифра телефонного номера «2» была заменена на «6» при указанной дате актуальности, меньшей даты проведения перенумерации (т.е. меньшей или равной 19.01.2004)
3	FOREIGN	Иностранн ый телефон	Задан иностранный телефон. Если код города не указан, то он будет определен по адресу	телефон: 442480705 адрес: Киев Украина	телефон: 380 44 2480705 адрес: Киев Украина	
4	EXTRA_ NUMBER	Телефон содержит добавочный номер	Телефон содержит добавочный номер с соответствующим комментарием, например, «доб. 52», «д. 52»	88123752 154 доб. 34	телефон: 8 812 3752154 доб.: 34	
5	MULTI CHANNEL	Телефон многоканальн ый	Указан многоканальный телефон (через символ «/» или в скобках указаны варианты окончания номера). Стандартизируем первый из указанных телефонов	пример 1: 780055005 01/02 пример 2: 8-(8617)-6783-64(65)	пример 1: 7 800 55005 01 пример 2: 7 8617 6783 64	
6	DOUBLE _CITY_ CODE	Задублирован код города	В исходных данных задублирован код города (актуальный и/или устаревший) или номер указан при помощи «выхода через 2 на межобласть»	пример 1: 7095495 48 7187448 пример 2: 095095 7187448 пример 3: 748231 823123720	пример 1: 7 495 71874 48 пример 2: 7 495 71874 48 пример 3: 7 48231 237 20	пример 1: задублированный код города Москвы («095» — устаревший, «495» — актуальный) был исправлен на актуальный;  пример 2: задублированный устаревший код города Москвы («095») был исправлен на актуальный;

пример 3 (номер указан при помощи «выхода через 2 на межобласть»):

- +7 — код страны (Россия)
- 48231 — код региона (Тверская область)
- 8231 — «выход» на межобласть через «82» (замена первых трех символов кода региона, у телефонов внутри региона первые три цифры это один и тот же код региона) и «31» — это оставшиеся символы кода региона (начиная с четвертого), которые внутри региона указывают на конкретный город/нп (код города Бежецк)
- 23720 — номер телефона

7	SUSPICIOUS_REPEATITIONS	Телефоны с повторяющимися цифрами	Поставляется для телефонов, в которых: <ul style="list-style-type: none"> <li>• 6 одинаковых цифр подряд;</li> <li>• 4 одинаковые цифры подряд и это единственные цифры;</li> <li>• 3 одинаковые пары цифр подряд;</li> <li>• 2 одинаковые тройки цифр подряд.</li> </ul>	пример 1: 79123123 пример 2: 7916177777 пример 3: 7916777777	пример 1: 7 912 312312 пример 2: 7 916 17777 пример 3: (пустое значение)	пример 3: если одинаковых цифр более 6, то номер считается мусорным, при стандартизации очищается
8	SUSPICIOUS_ORDER	Телефон с упорядоченными цифрами	Телефон или добавочная часть содержит возрастающую или убывающую числовую последовательность (например, 123456, 98765)	74956789123	7 495 6789123	
9		Восстановили	В исходных данных код города			Телефонный код

	CITY_CODE_THOUT_FIRST_DIGIT	первую цифру кода города	был указан без первой цифры. Код города определен по адресу	телефон: 115660223 адрес: г. Мамоново	телефон: 7 411 5660223 адрес: г. Мамоново	города «411» относится к г. Мамоново
10	GARBAGE	Убран мусор	Телефон содержал «мусорные» символы, которые были убраны при очистке	7923129 7115666	7 923 129 7115	
11	RESTORED_PHONE	Восстановлен код города по предыдущему номеру	Восстановлен код города по предыдущему номеру	(843) 266-37-35, 512-84-79	первый номер из строки: 7 843 2663735 второй номер из строки: 7 843 5128479	Из строки извлекли два номера телефона, для второго номера в строке восстановили код города по первому номеру. Для такой функции требуется подключение специального фильтра
12	RESTORED_ADDRESS	Восстановлен код города по адресу	Телефонный код города восстановлен по адресу лица	телефон: 2511418 адрес: г. Красноярск	телефон: 7 391 25114 18 адрес: г. Красноярск	Телефонный код «391» относится к г. Красноярск
13	RESTORED_DEFAULT	Восстановлен код города по умолчанию	Восстановлен код города, который устанавливается по умолчанию для тех номеров, у которых код города не указан	телефон: 398-86-07 код города по умолчанию: Москва	телефон: 7 495 3988607 код города по умолчанию: Москва	
14	RESTORED_COMMENT	Изменили тип телефона по комментарию	Изменили тип телефона по комментарию	телефон: 74991212 423 (моб) тип телефона: домашний	телефон: 7 499 1212423 тип телефона: мобильный	
15	RESTORED_PHONE	Изменили тип телефона по наличию добавочного или многоканальности	Изменили тип телефона по наличию добавочного или многоканальности	телефон: 8 (800) 100 24 24 доб. 123 тип телефона: домашний	телефон: 7 800 1002424 доб.: 123 тип телефона: рабочий	
16	RESTORED_DIRECTORY	Изменили тип телефона по вхождению в справочники	Изменили тип телефона по вхождению в справочники (Россвязь и другие)	телефон: 90216121 20 тип телефона: домашний	телефон: 7 902 1612120 тип телефона: мобильный	

17	SEVERAL_PHONES	Данные содержат более одного телефона	Исходные данные содержат более одного телефона	пример 1: телефон: 74991212 42374991 212423 пример 2.1: телефон: 398-46-47 398-86-07 город: (пустое значение) пример 2.2: телефон: 398-46-47 398-86-07 город: Красноярск	пример 1: первый номер из строки: 7 499 1212423 второй номер из строки: 7 499 1212423 пример 2.1: первый номер из строки: (пустое значение) второй номер из строки: (пустое значение) пример 2.2: первый номер из строки: 7 495 3984647 второй номер из строки: 7 495 3988607	В зависимости от настроек ЕК, результатом стандартизации является либо все найденные номера в строке (для каждого номера телефона будет проставлен маркер SEVERAL_PHONES), либо первый найденный номер в исходной строке.  пример 2.1: данные содержат более одного телефона, но не хватает символов для разбора (отсутствует код города), поэтому стандартизированные номера пустые  пример 2.2: данные содержат более одного телефона, восстановлен код города по адресу (город Красноярск), в результате чего стандартизированные номера имеют непустые значения (в отличие от примера 2.1)
18	AMBI	Есть несколько вариантов разбора телефона	Есть несколько вариантов разбора телефона, не удаётся выбрать лучший	телефон: 791 52 16 город: Москва	телефон: 7 495 7915216 город: Москва	Код города для телефона не удаётся восстановить однозначно. Код города Москвы 495 и 499
19	OUT_OF_RANGE	Телефон не попадает в диапазоны справочников РосСвязи	Телефон не попадает в диапазоны справочников РосСвязи	+7 997 801-23-45		Номер корректный по всем признакам, но отсутствует в справочнике нумерации РосСвязи
20	BDPN	Телефон найден в БДПН	Телефон найден в базе данных перенесенных номеров мобильных телефонов	79016260001	7 901 6260001	Номер телефона не попадает в диапазоны справочников РосСвязи, но есть в БДПН (номер считается корректным)



# ПРИЧИНЫ, ПОЧЕМУ ТЕЛЕФОН НЕ РАСПОЗНАН

№	Литера	Локализация	Описание	Исходно	Стандартизировано
1	NOT_PHONE	Слишком короткий номер	Номер состоит из менее, чем 5 цифр. Вероятно, числовая последовательность не является номером телефона	4532	(пустое значение)
2	TOO_FEW_DIGITS	Не хватает символов для разбора	Номер состоит из 5 — 9 цифр. Возможно, в номере телефона пропущены цифры	92544007	(пустое значение)
3	NOT_PARSABLE	Нераспознанный номер	Неоднозначный номер, состоящий из 10 или 11 цифр	<ul style="list-style-type: none"><li>• 7911021906, 8960834483 — в российской мобильном пропущена 1 цифра;</li><li>• +7 8422 73764 — в стационарном номере пропущена 1 цифра;</li><li>• +7 907 569 89 56 — нет кода 907 для мобильных;</li><li>• (951)2192788 -2ой номер — комментарий содержит цифру рядом с номером;</li><li>• +1 917 859 2020 + Татарстан — номер из США, но похож на российский мобильный и адрес передан российский.</li></ul>	(пустое значение)
4	TOO_MANY_DIGITS	Слишком длинный номер	Слишком длинный номер из 12 или более цифр	78123274717796268324	(пустое значение)

# КОДЫ КАЧЕСТВА ФИО И ПОЛА

- Общий код качества ФИО
- Статус распознавания ФИО
- Причины отправки ФИО на ручной разбор
- Маркеры проверки и изменения ФИО
- Маркеры проверки и изменения фамилии
- Маркеры проверки и изменения имени
- Маркеры проверки и изменения отчества
- Код качества пола

# ОБЩИЙ КОД КАЧЕСТВА ФИО

## Общий статус распознавания ФИО

Статус распознавания ФИО может принимать одно значение.

№	Литера	Локализация	Описание
1	NOT_SURE	Сомнительное значение	Проставляется в случае, если при распознавании ФИО были получены маркеры причин NOT_SURE
2	EDITED	Правильное значение	Во всех иных случаях и изменениях ФИО, не приводящих к NOT_SURE

# СТАТУС РАСПОЗНАВАНИЯ ФИО

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
<b>1 Не требуют проверки человеком</b>						
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значения ФИО и/или подтвердил их	ФИО: Вяхорев Александр пол: (пустое значение)	Ф: Вяхорев И: Александр О: Иванович пол: мужской	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значения ФИО и подтвердил их
3	EDITED	Правильное значение	Все проверки прошли, ФИО корректное	пример 1: ФИО: Каменская Наталья Александровна пол: женский пример 2: ФИО: Вяхорев Александр р Александрович пол: (пустое значение)	пример 1: Ф: Каменская И: Наталья О: Александровна пол: женский пример 2: Ф: Вяхорев И: Александр О: Александрович пол: мужской	пример 1: все проверки прошли, ФИО корректное пример 2: все проверки прошли, ФИО корректное, пол определен по ФИО
<b>4 На ручную обработку</b>						
5	NOT_SURE	Сомнительное значение	примеры представлены на странице Причины отправки ФИО на ручной разбор			
6	UNPROCESSED	Значение без стандартизации	Для значения ФИО, распознанного как иностранное, отключена стандартизация (ФИО с данным кодом качества не отправляются на ручной разбор)	Фамилия: Гвон Имя: Су Ен	Фамилия: Гвон Имя: Су Ен	
7	MISSED	Неизвестный код	Атрибут не проходил стандартизацию			

# СТАТУС РАСПОЗНАВАНИЯ ФИО (БЕЗ ГРАНУЛЯРНЫХ КК ФИО)

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
<b>1 Не требуют проверки человеком</b>						
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значения ФИО и/или подтвердил их	ФИО: Вяхорев Александр пол: (пустое значение)	Ф: Вяхорев И: Александр О: (пустое значение) пол: мужской	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значения ФИО и/или подтвердил их
3	EDITED	Правильное значение	Все проверки прошли, ФИО корректное	пример 1: ФИО: Каменская Наталья Александровна пол: женский пример 2: ФИО: Вяхорев Александр Александрович пол: (пустое значение)	пример 1: Ф: Каменская И: Наталья О: Александровна пол: женский пример 2: Ф: Вяхорев И: Александр О: Александрович пол: мужской	пример 1: все проверки прошли, ФИО корректное пример 2: все проверки прошли, ФИО корректное, пол определен по ФИО
<b>4 На ручную обработку</b>						
5	NOT_SURE	Сомнительное значение	Требуется ручная обработка	пример 1: Вяхорев Александр Александрович пример 2: Геннадий Валерьевич Лютиков пример 3: Каменский Александр Александрович пример 4: Александр Евгеньевич Ивашин пример 5: Вяхорев Александр Владимирович пример 6: Олег Владимирович Лютиков пример 7: Вяхорев Александр Александрович пример 8: Валентина Николаевна Думцева пример 9: ФИО: Каменский Наталья Александрович пол: неизвестен (исходный пол не задан, результирующий не однозначен)	пример 1: Вяхорев Александр Александрович пример 2: Геннадий Валерьевич Лютиков пример 3: Каменский Александр Александрович пример 4: Александр Евгеньевич Ивашин пример 5: Вяхорев Александр Владимирович пример 6: Олег Владимирович Лютиков пример 7: Вяхорев Александр Александрович пример 8: Валентина Николаевна Думцева пример 9: ФИО: Каменский Наталья Александрович пол: неизвестен (исходный пол не задан, результирующий не однозначен)	пример 1: исправлена опечатка в отчестве пример 2: исправлена опечатка в имени пример 3: изменено окончание фамилии в соответствии с определенным по ФИО полом пример 4: изменено окончание отчества в соответствии с определенным по ФИО полом пример 5: нет отчества, ФИО неполное пример 6: нет фамилии, ФИО неполное пример 7: убран мусор пример 8: убран мусор из имени и фамилии

			Евгеньев на Ивашин пример 5: Вягоре в Александр пример 6: Олег Владими рович пример 7: Вягоре в Александр р Александр рович 31 /05/1995 пример 8: В\\але нтина Николаев на Думце7 ва пример 9: ФИО: Кам енский Наталья Александр рович пол: (пус тое значение)	Поле «пол» опущено для краткости примеров 1 – 8. Для примеров 1–7 определен пол «мужской», для примера 8 определен пол «женский»	пример 9: неоднозначность пола
6	MISSED	Неизвестный код	Атрибут не проходил стандартизацию		

# ПРИЧИНЫ ОТПРАВКИ ФИО НА РУЧНОЙ РАЗБОР

## Маркеры причин некорректности ФИО

Может быть несколько значений для ФИО

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	FNAME_EMPTY	Имя пустое	Имя не найдено	ФИО: Умарова Ивановна	Ф: Умарова И: (пустое значение) О: Ивановна	Фамилию, имя и отчество можно указать в одном поле, система сама разделит их на отдельные составляющие
2	LNAME_EMPTY	Фамилия пустая	Фамилия не найдена	ФИО: Илья Владимирович	Ф: (пустое значение) И: Илья О: Владимирович	
3	MNAME_EMPTY	Отчество пустое	Отчество не найдено	ФИО: Иванов Илья	Ф: Иванов И: Илья О: (пустое поле)	
4	RETRANSLIT_N OT_SURE	Транслитерация неоднозначна	Была проведена обратная транслитерация символов с латиницы в кириллицу, возможно несколько вариантов написания измененных символов	ФИО: Parfenov Dmitry Vladimirovich	Ф: Парфенов И: Дмитрий О: Владимирович	Возможно несколько вариантов формирования обратной транслитерации с латиницы в кириллицу. Для фамилии «Parfenov» Буквы E и Ё могут передаваться одинаково: E, YE.
5	REDUNDANCY	Дублирование частей	ФИО содержит продублированную информацию (2 и более одинаковых слова)	ФИО: Иванов Иванов Сергей Владимирович	Ф: Иванов И: Сергей О: Владимирович	Фамилию, имя и отчество можно указать в одном поле, система сама разделит их на отдельные составляющие
6	TRANSEXUAL	Изменено окончание	Произведено изменение окончания фамилии или отчества в соответствии с определенным по ФИО полом	Ф: Иванов И: Ирина О: Владимировна	Ф: Иванова И: Ирина О: Владимировна	
7	PREFIX	Иностранное префикс	Исходные данные содержат приставки иностранных фамилий: ДЕ, Д', ФОН, ДЕЗ, ВАН, ГЕР, О', МАК, ЛЕ, ДА	ФИО: Леблан	Ф: Леблан И: Диана О: Владимировна	

				Диана Владими ровна		
8	UNNATURAL	Нестандартный порядок слов	Нестандартный порядок слов в исходном ФИО, например: 1. Фамилия была передана в поле «Имя», а имя — в поле «Фамилия», отчество — в поле «Отчество», и при объединении полей получается «неестественный» порядок ФИО — ИФО. 2. ФИО были переданы одной строкой в порядке, отличном от «Фамилия», «Имя», «Отчество».	ФИО: Мария Михайловна Иванова	Ф: Иванова И: Мария О: Михайловна	ФИО были переданы одной строкой в порядке, отличном от «Фамилия», «Имя», «Отчество»
9	CONFLICT	Конфликт компонент	В ФИО присутствуют две компоненты, которые претендуют на одно и тоже место	ФИО: Кудрявцев Юрий Евгений	Ф: Кудрявцев И: Юрий О: Евгений (отсутствует в справочниках)	
10	GENDER_UNKNOWN	Не удалось определить пол	Исходный пол не задан и все компоненты ФИО не найдены в справочнике и не попали под эвристики определения пола на основе окончания фамилий и отчеств	ФИО: Энхбаатар Санравдорж пол: (пустое значение)	Ф: Энхбаатар (отсутствует в справочниках) И: Санравдорж (отсутствует в справочниках) О: (пустое значение) пол: (пустое значение)	
11	GENDER_NOT_EDITED	Противоречие по полу	Противоречие по полу между компонентами ФИО	ФИО: Петров Екатерина Владимирович	Ф: Петров И: Екатерина О: Владимирович	
12	EMPTY	Пустое ФИО	Пустая строка исходного ФИО и всех компонентов ФИО	ФИО: (пустое поле) Ф: (пустое значение) И: (пустое значение) О: (пустое значение)	Ф: (пустое значение) И: (пустое значение) О: (пустое значение)	
13	GARBAGE	Убран мусор	В исходном ФИО есть «мусор», который был убран. Например, нецензурная лексика или имена героев (Крокодил Гена, Чебурашка и т.п.)	пример 1: ФИО: Крокодил Гена пример 2: ФИО: Валентина Николаевна Думцев	пример 1: Ф: (пустое значение) И: (пустое значение) О: (пустое значение) пример 2: Ф: Думцева И: Валентина О: Николаевна	



				a 31/05 /1995		
14	TOO_L ONG	Слишком длинный компонент ФИО	Компонент ФИО более 15 символов	Ф: Ивано ва И: Ирина О: Предп ринимат ель без образова ния юр. лица Ива нова Ирина	Ф: Иванов И: Ирина О: Предприниматель без образования юр. лица Иванова Ирина	Настраивается дополнительно, по умолчанию не используется
15	CHANGE	ФИО изменено	Компонент ФИО был изменен. Причины: произведена замена компонента ФИО по справочникам, изменено окончание в соответствии с определенным полом или исправлена опечатка в одну букву	ФИО: Ку дрявцев Екатерин а Владими ровна	Ф: Кудрявцева И: Екатерина О: Владимировна	
16	LEAVE_ SOURCE_ _COMPO NENTS	Оставлен переданный порядок компонент, но он требует проверки	Автоматический разбор ФИО по компонентам противоречит присланному гранулярному ФИО	Ф: Жолд ошбек Кызы И: Гулда О: (пусто е значение)	Ф: Жолдошбек Кызы И: Гулда О: (пустое значение)	
17	DISPLA CEMENT	Компоненты перепутаны местами	Исходные компоненты ФИО перепутаны местами (например, фамилия была передана в поле «Фамилия», а имя и отчество — в поле «Имя»). При разборе перенесли компонент ФИО из одного поля в другое	Ф: Ивана йко И: Ольга Сергеевна О: (пусто е значение)	Ф: Иванайко И: Ольга О: Сергеевна	

## Отображение в интерфейсе

Маркеры невалидности ФИО отображаются вместе с кодом качества ФИО.

Автор AL:183327534

Исходное ФИО A • A • A

■ Сомнительное значение (присутствуют инициалы, пол не определен, дублирование частей, убран мусор, пустое имя, пустая фамилия, пустое отчество)

Фамилия

Не найдено в исходной строке

Имя

Не найдено в исходной строке

Отчество

Не найдено в исходной строке

Пол

Исходный не задан

Стандартизовать Подтвердить Отложить

# МАРКЕРЫ ПРОВЕРКИ И ИЗМЕНЕНИЯ ФИО

## Маркеры проверки и изменения ФИО

Гранулярные коды качества ФИО — это составной набор маркеров и кодов качества Фамилии, Имени, Отчества и пола. Которые в совокупности дают представление о статусе разбора ФИО и изменениях, произведенных в процессе стандартизации.

Маркеры изменений ФИО и пола говорят об изменениях, которые были произведены с ФИО в процессе стандартизации. Может быть несколько значений.

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	FNAME_ EMPTY	Имя пустое	Имя не найдено	ФИО: Умар ова Ивановна	Ф: Умарова И: (пустое значение) О: Ивановна	Фамилию, имя и отчество можно указать в одном поле, система сама разделит их на отдельные составляющие
2	LNAME_ EMPTY	Фамилия пустая	Фамилия не найдена	ФИО: Илья Владимиро вич	Ф: (пустое значение) И: Илья О: Владимирович	
3	MNAME_ EMPTY	Отчество пустое	Отчество не найдено	ФИО: Иван ов Илья	Ф: Иванов И: Илья О: (пустое значение)	
4	FNAME_ EMPTY_ OR_UNL IST	Имя пустое или отсутствует в справочниках	Имя не заполнено или отсутствует в справочниках	ФИО: Иван ов Владимиро вич	Ф: Иванов И: (пустое значение) О: Владимирович	
5	FNAME_ UNLIST	Имя отсутствует в справочниках	Имя отсутствует в справочниках	ФИО: Иван ов Серёжа В ладимиров ич	Ф: Иванов И: Серёжа О: Владимирович	
6	FNAME_ NOT_GO OD_MNA ME_EMP TY	Имя нехорошее и отчество пустое	Имя отсутствует в справочнике или разобрано и приведено к справочному значению. Отчество пустое	ФИО: Семе нюк Катя	Ф: Семенюк И: Катя О: (пустое значение)	
7	FNAME_ UNLIST _MNAME _EMPTY	Имя отсутствует в справочниках и отчество пустое	Имя отсутствует в справочниках и отчество пустое	ФИО: Семенюк Ка тя	Ф: Семенюк И: Катя О: (пустое значение)	
8	LNAME_ UNLIST	Фамилия отсутствует в справочниках	Фамилия отсутствует в справочниках	ФИО: Гунхи лд Илья Владимиро вич	Ф: Гунхилд И: Илья О: Владимирович	
9	LNAME_ NOT_GO OD_MNA ME_EMP TY	Фамилия нехорошая и отчество пустое	Фамилия разобрана и приведена к справочнику. Отчество пустое	ФИО: Иван овп Анна	Ф: Иванова И: Анна О: (пустое значение)	

10	LNAME_ UNLIST _MNAME _EMPTY	Фамилия отсутствует в справочниках и отчество пустое	Фамилия отсутствует в справочниках и отчество пустое	ФИО: Гунхи лд Илья	Ф: Гунхилд И: Илья О: (пустое значение)	
11	MNAME_ EMPTY_ OR_UNL IST	Отчество пустое или отсутствует в справочниках	Отчество не заполнено или отсутствует в справочниках	ФИО: Иван ов Илья Се мкуфц	Ф: Иванов И: Илья О: Семкуфц	
12	MNAME_ UNLIST	Отчество отсутствует в справочниках	Отчество отсутствует в справочниках	ФИО: Иван ов Илья Се мкуфц	Ф: Иванов И: Илья О: Семкуфц	
13	LNAME_ AND_FN AME_EM PTY_OR _UNLIST	Фамилия и имя пустые или отсутствуют в справочниках	Фамилия и имя пустые или отсутствуют в справочниках	ФИО: Ильи нична	Ф: (пустое значение) И: (пустое значение) О: Ильинична	
14	LNAME_ AND_MN AME_EM PTY_OR _UNLIST	Фамилия и отчество пустые или отсутствуют в справочниках	Фамилия и отчество пустые или отсутствуют в справочниках	ФИО: Евген ия	Ф: (пустое значение) И: Евгения О: (пустое значение)	
15	FNAME_ AND_MN AME_EM PTY_OR _UNLIST	Имя и отчество пустые или отсутствуют в справочниках	Имя и отчество пустые или отсутствуют в справочниках	ФИО: Иван ов	Ф: Иванов И: (пустое значение) О: (пустое значение)	
16	RETRAN SLIT_N OT_SURE	Транслитерац ия неоднозначна	Была проведена обратная транслитерация символов с латиницы в кириллицу, возможно несколько вариантов написания измененных символов	ФИО: Parfe nov Dmitry Vladimirovich	Ф: Парфенов И: Дмитрий О: Владимирович	Возможно несколько вариантов формирования обратной транслитерации с латиницы в кириллицу. Для фамилии «Parfenov» Буквы Е и Ё могут передаваться одинаково: Е, УЕ.
17	TOO_LO NG	Слишком длинный компонент ФИО	Один из компонентов ФИО содержит более 15 символов	ФИО: Иван ова Ирина Предприни матель без образования я юр. лица Ивано ва Ирина	Ф: Иванова И: Ирина О: Предприниматель без образования юр. лица Иванова Ирина	Настраивается дополнительно, по умолчанию не используется
18	LEAVE_ SOURCE	Оставлен переданный	Автоматический разбор ФИО по компонентам противоречит	Ф: Жолдош бек Кызы	Ф: Жолдошбек Кызы	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
	_COMPONENTS	порядок компонент, но он требует проверки	присланному гранулярному ФИО	И: Гулзада О: (пустое значение)	И: Гулзада О: (пустое значение)	
19	REDUNDANCY	Дублирование частей	ФИО содержит продублированную информацию (2 и более одинаковых слова)	ФИО: <b>Иванов Иван Сергеевич</b> Владимирович	Ф: <b>Иванов</b> И: <b>Сергей</b> О: <b>Владимирович</b>	
20	TRANSSEXUAL	Изменено окончание в соответствии с определенным полом	Произведено изменение окончания фамилии или отчества в соответствии с определенным по ФИО полом	Ф: <b>Иванов</b> И: <b>Ирина</b> О: <b>Владимировна</b>	Ф: <b>Иванова</b> И: <b>Ирина</b> О: <b>Владимировна</b>	
21	PREFIX	Исходные данные содержат приставки иностранных фамилий	Исходные данные содержат приставки иностранных фамилий: <b>ДЕ, Д', ФОН, ДЕЗ, ВАН, ГЕР, О', МАК, ЛЕ, ДА</b>	ФИО: <b>Ле</b> <b>блан Диана Владимировна</b>	Ф: <b>Ле Блан</b> И: <b>Диана</b> О: <b>Владимировна</b>	
22	MISPRINT	Исправлены опечатки в одну букву	Исправлена опечатка в одну букву в одном или нескольких компонентах	ФИО: <b>Коларски Цветомир</b> Стефанович	Ф: <b>Коларски</b> И: <b>Цветамир</b> О: <b>Стефанович</b>	
23	UNNATURAL	Нестандартный порядок слов в исходном ФИО	Нестандартный порядок слов в исходном ФИО, например: 1. Фамилия была передана в поле «Имя», а имя — в поле «Фамилия», отчество — в поле «Отчество». При «объединении» полей получается «неестественный» порядок ФИО — ИФО; 2. ФИО были переданы одной строкой в порядке, отличном от «Фамилия», «Имя», «Отчество».	ФИО: <b>Мария Михайловна Иванова</b>	Ф: <b>Иванова</b> И: <b>Мария</b> О: <b>Михайловна</b>	ФИО были переданы одной строкой в порядке, отличном от «Фамилия», «Имя», «Отчество»
24	CONFLICT	Конфликт компонент	В ФИО присутствуют две компоненты, которые претендуют на одно и тоже место	ФИО: <b>Кудрявцев Юрий</b> <b>Евгений</b>	Ф: <b>Кудрявцев</b> И: <b>Юрий</b> О: <b>Евгений</b> (отсутствует в справочнике отчеств)	
25	GENDER_UNKNOWN	Не удалось определить пол	Все компоненты мусорные, пустые или не справочные	ФИО: <b>Энхбаатар Санравдорж</b> пол: (пустое значение)	Ф: <b>Энхбаатар</b> (отсутствует в справочниках) И: <b>Санравдорж</b> (отсутствует в справочниках) О: (пустое значение) пол: (пустое значение)	
26		ФИО	На входе нет пола, есть		Ф: <b>Хан</b> (unisex)	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
	GENDER _UNISEX	подходит любому полу	универсальный по полу компонент, остальные тоже универсальные, пустые или не справочные	ФИО: Хан Влабимир Игоровичь пол: <i>(пустое значение)</i>	И: Влабимир (отсутствует в справочниках) О: Игоровичь (отсутствует в справочниках) пол: <i>(пустое значение)</i>	
27	GENDER _AMBI	Разнополые компоненты	Противоречие между значениями пола, определёнными по разным компонентам	пример 1: ФИО: Иван ов Ирина пол: <i>(пустое значение)</i>	пример 1: Ф: Иванов (М) И: Ирина (Ж) О: <i>(пустое значение)</i> пол: <i>(пустое значение)</i> неоднозначен	Результирующий пол неоднозначен — может быть как женского, так и мужского пола.
28	GENDER _CHANGED	Пол отличается от исходного	Результирующий пол отличается от исходного	ФИО: Кара мелькина Марина Владимир овна пол: <b>мужской</b>	Ф: Карамелькина И: Марина О: Владимировна пол: <b>женский</b>	
29	TRANSG RAPH	Применена трансграфика	Заменены вхождения в строку латинских букв, которые имеют визуальные соответствия в русском алфавите	ФИО: Рудне в Иван Владимиро вич	Ф: Руднев И: Иван О: Владимирович	
30	RETRAN SLIT	Применена транслитерация	Была проведена обратная транслитерация с латиницы в кириллицу	ФИО: Kazan tsev Dmitry Andreevich	Ф: Казанцев И: Дмитрий О: Андреевич	
31	DEFICI ENCY	Присутствуют инициалы	Вместо имени и/или отчества указаны инициалы	ФИО: Нику лина М В	Ф: Никулина И: М О: В	
32	GENDER _NOT_EDITED	Противоречие по полу	Противоречие по полу между компонентами ФИО	ФИО: Петро в Екатерина Владимиро вич	Ф: Петров И: Екатерина О: Владимирович	
33	EMPTY	Пустое ФИО	Пустая строка исходного ФИО и всех компонентов ФИО	ФИО: <i>(пустое значение)</i> Ф: <i>(пустое значение)</i> И: <i>(пустое значение)</i> О: <i>(пустое значение)</i>	Ф: <i>(пустое значение)</i> И: <i>(пустое значение)</i> О: <i>(пустое значение)</i>	
34	GARBAGE	Убраны мусорные слова из исходного ФИО	В исходном ФИО есть "мусор", который был убран. Например, нецензурная лексика или имена героев (Крокодил Гена, Чебурашка и т.п.)	пример 1: ФИО: <b>Крокодил Гена</b> пример 2: ФИО: В\\ал ентина Николаевна	пример 1: Ф: <i>(пустое значение)</i> И: <i>(пустое значение)</i> О: <i>(пустое значение)</i> пример 2: Ф: Думцева	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
				Думце7ва 3 1/05/1995	И: Валентина О: Николаевна	
35	STRONG_CHANGE	Произведена замена по справочникам	ФИО изменено по справочникам опечаток и/или уменьшительно-ласкательных слов	ФИО: Иванове Иринка Владимировна	Ф: Иванов И: Ирина О: Владимировна	
36	CHANGE	В компоненты ФИО внесены изменения	Компонент ФИО был изменен. Причины: произведена замена компонента ФИО по справочникам, изменено окончание в соответствии с определенным полом, или исправлена опечатка в одну букву	ФИО: Кудрявцев Екатерина Владимировна	Ф: Кудрявцева И: Екатерина О: Владимировна	Агрегирующий маркер, который срабатывает при изменении любого компонента ФИО. Появляется при маркерах: <ul style="list-style-type: none"> <li>• STRONG_CHANGE (произведена замена по справочникам)</li> <li>• TRANSSEXUAL (изменено окончание в соответствии и с определенным полом)</li> <li>• MISPRINT (исправлены опечатки в одну букву). Фактор изменил один или несколько компоненто в, чтобы привести их к справочнику</li> </ul>
37	DISPLACEMENT	Исходные компоненты ФИО перепутаны местами	Исходные компоненты ФИО перепутаны местами (например, фамилия была передана в поле «Фамилия», а имя и отчество — в поле «Имя»)	Ф: Иванайко И: Ольга Сергеевна О: (пустое значение)	Ф: Иванайко И: Ольга О: Сергеевна	При разборе ФИО был перенесен компонент «Сергеевна» из поля «Имя» в поле «Отчество»
38	VERY_FEW_UNIQUE_CHARS_IN_FIO	Очень мало уникальных букв в ФИО	Для длины ФИО содержится очень мало уникальных букв	ФИО: Вовав Вова Вовович	Ф: Вовав И: Вова О: Вовович	ФИО проверяется на предмет срабатывания

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
	IQUE_L ETTERS _TOTAL					данного маркера только в том случае, когда два любых компонента ФИО не являются справочными. Оставшийся компонент может быть любым.
39	FNAME _PROB ABLY_G ARBAGE	Значение в имени похоже на мусор	В имени содержится последовательность, распознанная как мусор	ФИО: Андреев Аэо Иванович	Ф: Андреев И: Аэо О: Иванович	Срабатывает, когда имя имеет маркер UNLIST + любой из следующего списка: <ul style="list-style-type: none"> <li>• ONLY_VOWELS_SHORT (только гласные буквы, длиной 3 символа);</li> <li>• VOWELS_LETTERS (больше 2 гласных подряд);</li> <li>• CONSONANTS_LETTERS (больше 3 согласных подряд);</li> <li>• RARE_SUBSTRINGS (содержит мин. 2 трехбуквенные редкие для справочных ФИО последовательности и букв).</li> </ul>
40	FNAME _GARB AGE	Мусор в имени	В имени содержится последовательность, распознанная как мусор	ФИО: Штольц Ауэи Иванович	Ф: Штольц И: Ауэи О: Иванович	Срабатывает, когда имя имеет маркер: <ol style="list-style-type: none"> <li>1. ONLY_CONSONANTS</li> </ol>

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
						<p>(только согласные буквы).</p> <p>2. UNLIST + любой из следующего списка:</p> <ul style="list-style-type: none"> <li>• ONLY_VOWELS_LONG (только гласные буквы, длиной 4 и более символа);</li> <li>• IDENTICAL_LETTERS (содержит больше 2 одинаковых букв подряд).</li> </ul>
41	MNAM E_PROB ABLY_G ARBAGE	Значение в отчестве похоже на мусор	В отчестве содержится последовательность, распознанная как мусор	ФИО: Петро в Андрей Схфтаунович	Ф: Петров И: Андрей О: Схфтаунович	<p>Срабатывает, когда отчество имеет маркер UNLIST + любой из следующего списка:</p> <ul style="list-style-type: none"> <li>• ONLY_VOWELS_SHORT (только гласные буквы, длиной 3 символа);</li> <li>• VOWELS_LETTERS (больше 2 гласных подряд);</li> <li>• CONSONANTS_LETTERS (больше 3 согласных подряд);</li> <li>• RARE_SUBSTRING</li> </ul>



№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
						(содержит мин. 2 трехбуквенные редкие для справочных ФИО последовательности и букв).
42	MNAM E_GARB AGE	Мусор в отчестве	В отчестве содержится последовательность, распознанная как мусор	ФИО: Василий Аркадий Еееклович	Ф: Васильев И: Аркадий О: Еееклович	Срабатывает, когда отчество имеет маркер:  1. ONLY_CONSONANTS (только согласные буквы). 2. UNLIST + любой из следующего списка: <ul style="list-style-type: none"> <li>ONLY_VOWELS_LONG (только гласные буквы, длиной 4 и более символа);</li> <li>IDENTICAL_LETTERS (содержит больше 2 одинаковых букв подряд).</li> </ul>
43	LNAME _PROB ABLY_G ARBAGE	Значение в фамилии похоже на мусор	В фамилии содержатся последовательность, распознанная как мусор	ФИО: Уосол э Андрей Иванович	Ф: Уосолэ И: Андрей О: Иванович	Срабатывает, когда фамилия имеет маркер UNLIST + любой из следующего списка:  <ul style="list-style-type: none"> <li>ONLY_VOWELS_SHORT</li> </ul>


№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
						<p>(только гласные буквы, длиной 3 символа);</p> <ul style="list-style-type: none"> <li>• VOWELS_LETTERS (больше 2 гласных подряд);</li> <li>• CONSONANTS_LETTERS (больше 3 согласных подряд);</li> <li>• RARE_SUBSTRINGS (содержит мин. 2 трехбуквенные редкие для справочных ФИО последовательности и букв).</li> </ul>
44	LNAME _GARB AGE	Мусор в фамилии	В фамилии содержатся последовательность, распознанная как мусор	ФИО: Тргдв Андрей Николаевич	Ф: Тргдв И: Андрей О: Николаевич	<p>Срабатывает, когда фамилия имеет маркер:</p> <ol style="list-style-type: none"> <li>1. ONLY_CONSONANTS (только согласные буквы).</li> <li>2. UNLIST + любой из следующего списка: <ul style="list-style-type: none"> <li>• ONLY_VOWELS_LONG (только гласные буквы, длиной 4 и более символа);</li> <li>• IDENTICAL_LETTERS (содержит</li> </ul> </li> </ol>

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
						больше 2 одинако вых букв подряд).
45	FIO_GA RBAGE	Мусор в ФИО	Компоненты ФИО похожи на мусорные значения	ФИО: Бобобо Максим Бобобович	Ф: Бобобо И: Максим О: Бобобович	Срабатывает в следующих случаях:  1. Пара маркеров UNLIST + VERY_FEW_UNIQUE_LETTERS встречается одновременно как минимум у двух компоненто в ФИО;  2. Пара маркеров UNLIST + UNUSUAL_SUBSTRING встречается одновременно как минимум у двух компоненто в ФИО.
46	FIO_PR OBABL Y_GAR BAGE	ФИО похоже на мусор	Компоненты ФИО похожи на мусорные значения	ФИО: Хаяаг а Иван Гаяача	Ф: Хаяага И: Иван О: Гаяача	Срабатывает, когда пара маркеров UNLIST и FEW_UNIQUE_LETTERS встречается одновременно как минимум у двух компонентов ФИО

## Отображение в интерфейсе

Маркеры изменений в ФИО подсвечены зеленым цветом.

Автор: AL-549343


Исходное ФИО: Куава Екатерина 

**Сомнительное значение** (применена транслитерация, применена транслитерация, ФИО изменено, произведена замена по справочникам, транслитерация неоднозначна, пустое отчество)

Фамилия: Куава  
Отсутствует в справочниках

Имя: Екатерина  
Приведено к справочнику

Отчество:   
Не найдено в исходной строке

Пол: Женский   
Исходный не задан

Стандартизировать Подтвердить Отменить

## Примеры обработки

Литера	Локализация	Исходно	Стандартизировано	Объяснение проставленного кода качества	Маркеры изменений FULLNAME_EDITED_CHANGES	Маркеры некорректности FULLNAME_EDITED_CHANGES
<b>На ручную обработку</b>						
NOT_SURE	Сомнительное значение	Смирнов Петр Петрови Чебурашка	Смирнов Петр Петрович	Исправлена опечатка в отчестве. Убран мусор	STRONG_CHANGE <ul style="list-style-type: none"> <li>ФИО изменено по справочникам опечаток</li> </ul>	CHANGE <ul style="list-style-type: none"> <li>В к изи</li> <li>Уб ис</li> </ul>
		Мартirosян Г Г	Мартirosян Г Г	<ul style="list-style-type: none"> <li>Убрана дублирующая информация</li> <li>Инициалы вместо имени и отчества</li> </ul>	DEFICIENCY <ul style="list-style-type: none"> <li>Вместо имени и/или отчества указаны инициалы</li> </ul>	REDUNDANCE <ul style="list-style-type: none"> <li>ФИ пр ин</li> <li>Пу</li> <li>Пу</li> </ul>
		Ivanova Ludmila Petriovna	Иванова Людмила Петровна	Транслитерация латиницы в кириллицу	RETRANSLIT <ul style="list-style-type: none"> <li>Была проведена обратная транслитерация из латиницы в кириллицу</li> </ul>	RETRANSLIT <ul style="list-style-type: none"> <li>Пр тр: в к во: ва</li> </ul>
		Высоцкaя Ирина Ивановна	Высоцкая Ирина Ивановна	Изменено окончание фамилии в соответствии с определенным по ФИО полом		CHANGE <ul style="list-style-type: none"> <li>В к изи</li> <li>Из со: оп</li> </ul>

Ольга Алексеев на	Ольга Алексеевна	Нет фамилии, ФИО неполное		LNAME
				• Пу
Куаеа Ек aTeRна	Куаеа Екатерина	Произведена транслитерация и трансграфика, исправлена опечатка в имени	TRANSGRAPH STRONG_CHANGE RETRANSLIT	CHANG MNAME
			<ul style="list-style-type: none"> <li>• Заменены латинские буквы в имени</li> <li>• Произведена замена по справочникам опечаток (пропущена "и")</li> <li>• Произведена обратная транслитерация</li> </ul>	<ul style="list-style-type: none"> <li>• В к из</li> <li>• Тр. не</li> <li>• Пу</li> </ul>

# МАРКЕРЫ ПРОВЕРКИ И ИЗМЕНЕНИЯ ФАМИЛИИ


## Гранулярный код качества фамилии

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	STRONG _CHANGE	Произведена замена по справочникам	Компонент изменен по справочникам опечаток и уменьшительно-ласкательных слов	ФИО: <b>Иванов</b> Ирина Владимировна	Ф: <b>Иванова</b> И: Ирина О: Владимировна	Фамилию, имя и отчество можно указать в одном поле, система сама разделит их на отдельные составляющие
2	TRANSEXUAL	Изменено окончание по полу	Изменено окончание в соответствии с определенным по ФИО полом	ФИО: <b>Иванов</b> Ирина Владимировна	Ф: <b>Иванова</b> И: Ирина О: Владимировна	
3	PREFIX	Иностраннный префикс	Компонент содержит иностранную приставку: ДЕ, Д', ФОН, ДЕЗ, ВАН, ГЕР, О', МАК, ЛЕ, ДА	ФИО: <b>Леблан</b> Диана Владимировна	Ф: <b>Леблан</b> И: Диана О: Владимировна	
4	TOO_LONG	Значение слишком длинное	Компонент содержит более 15 символов	ФИО: <b>Христорождественский</b> Петр Ильич	Ф: <b>Христорождественский</b> И: Петр О: Ильич	
5	UNLIST	Отсутствует в справочниках	Распознанный компонент отсутствует в справочниках	ФИО: <b>Комаан</b> Петр Ильич	Ф: <b>Комаан</b> И: Петр О: Ильич	
6	EMPTY	Не найдено в исходной строке	Не найдено в исходной строке (возможно, пустая строка)	пример 1: <b>Елена Владимировна</b> пример 2: <b>ООЕлена Владимировна</b> пример 3: <b>Елена Владимировна</b>	для примера 1, 2 и 3: Ф: <i>(пустое значение)</i> И: Елена О: Владимировна	
7		Компонент	В исходной строке при			для примера 1 и 2:

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
	UNNATURAL	взят из позиции для другого компонента	«объединении» полей получается «неестественный» порядок ФИО — например, ИФО	пример 1: <b>Иван Иванов</b> В ладимир ович пример 2: <b>Иван Иванов</b> О: Влади мирович	<b>Ф: Иванов</b> <b>И: Иван</b> О: Владимирович	пример 1: ФИО передано одной строкой в порядке, отличном от «фамилия — имя — отчество». пример 2: ФИО передано гранулярно, при «объединении» полей исходного ФИО получается «неестественный» порядок ФИО — ИФО
8	OK_NOT_CHANGED	Корректное значение* (устар.)  *статус не показываем в интерфейсе	Разобрано, без изменений найдено в справочнике	ФИО: Ивахненко Ивахненко Александр Алексеевич	Ф: Ивахненко И Александр О: Алексеевич	
9	OK_CHANGED	Приведено к справочнику (устар.)	Разобрано, приведено к справочнику	ФИО: Шарый Вера Андреевна	Фамилия: Шарая И Вера О: Андреевна	
10	ONLY_CONSONANTS	Только согласные буквы	Компонент длиной 3 или более символов состоит только из согласных русских букв	ФИО: <b>Трдв</b> Андрей Петрович	Ф: <b>Трдв</b> И: Андрей О: Петрович	
11	ONLY_VOWELS_SHORT	Только гласные буквы	Компонент длиной 3 символа состоит только из гласных русских букв	ФИО: <b>Аоэ</b> Андрей Петрович	Ф: <b>Аоэ</b> И: Андрей О: Петрович	
12	ONLY_VOWELS_LONG	Только гласные буквы	Компонент длиной 4 или более символа состоит только из гласных русских букв	ФИО: <b>Иаэия</b> Андрей Петрович	Ф: <b>Иаэия</b> И: Андрей О: Петрович	
13	IDENTICAL LETTERS	Больше 2 одинаковых буквы подряд	Компонент содержит подряд 3 или более одинаковые русские буквы	ФИО: <b>Еердров</b> Андрей Петрович	Ф: <b>Ееердов</b> И: Андрей О: Петрович	
14	VOWELS_LETTERS	Больше 2 гласных подряд	Компонент содержит подряд 3 или более гласных русских буквы	ФИО: <b>Боыэндарчук</b> Андрей Петрович	Ф: <b>Бобыэндарчук</b> И: Андрей О: Петрович	
15	CONSONANTS_LETTERS	Больше 3 согласных подряд	Компонент содержит подряд 4 или более согласных русских буквы	ФИО: <b>Кртдвшкин</b> Андрей Петрович	Ф: <b>Кртдвшкин</b> И: Андрей О: Петрович	
16		Мало	Для длины компонента	ФИО: <b>Хая</b>	Ф: <b>Хаяага</b>	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
	FEW_UNIQUE_LETTERS	уникальных букв	содержится мало уникальных русских букв	ага Андр ей Петрович	И: Андрей О: Петрович	
17	VERY_FEW_UNIQUE_LETTERS	Очень мало уникальных букв	Для длины компонента содержится очень мало уникальных русских букв	ФИО: Боб обов Анд рей Петрович	Ф: Бобобов И: Андрей О: Петрович	
18	UNUSUAL_SUBSTRING	Значение похоже на мусор	Компонент содержит трехбуквенную последовательность, которая не встречается в справочных ФИО	ФИО: Вп лапъ Анд рей Петрович	Ф: Вплапъ И: Андрей О: Петрович	
19	RARE_SUBSTRING	Значение похоже на мусор	Компонент содержит как минимум две трехбуквенных последовательности, которые являются очень редкими для справочных ФИО	ФИО: Уос олэ Анд рей Петрович	Ф: Уосолаэ И: Андрей О: Петрович	

## Отображение в интерфейсе

Исходное ФИО Шарый Вера Андреевна 

■ Сомнительное значение (ФИО изменено, изменено окончание)

Фамилия   
Приведено к справочнику

Имя

Отчество



# МАРКЕРЫ ПРОВЕРКИ И ИЗМЕНЕНИЯ ИМЕНИ


## Гранулярный код качества имени

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	STRONG_CHANGE	Произведена замена по справочникам	Компонент изменен по справочникам опечаток и уменьшительно-ласкательных слов	пример 1: Иванов Ад ексндр И ванович пример 2: Иванова Иринка И вановна	для примера 1: Ф: Иванов И: Александр О: Иванович для примера 2: Ф: Иванова И: Ирина О: Ивановна	Фамилию, имя и отчество можно указать в одном поле, система сама разделит их на отдельные составляющие
2	PREFIX	Иностраннный префикс	Компонент содержит иностранную приставку: ДЕ, Д', ФОН, ДЕЗ, ВАН, ГЕР, О', МАК, ЛЕ, ДА	ФИО: Иванова ЛЕ Диана	Ф: Иванова И: ЛЕ Диана О: (пустое значение)	
3	UNLIST	Отсутствует в справочниках	Распознанный компонент отсутствует в справочниках	ФИО: Нигматуллин Акзал Гар аевич	Ф: Нигматуллин И Акзал О: Гараевич	Распознанное имя Акзал отсутствует в справочниках имен
4	DEFICIENCY	Присутствует инициал	Вместо полного значения указан инициал	ФИО: Ива нов И Ива нович	Ф: Иванов И: И О: Иванович	
5	TOO_LONG	Значение слишком длинное	Компонент содержит более 15 символов	ФИО: Ива нов Грейг ивовалла сис Иван ович	Ф: Иванов И: Грейгивовалласис О: Иванович	
6	EMPTY	Не найдено в исходной строке	Не найдено в исходной строке (возможно, пустая строка)	пример 1: Лукьянов Филиппо вич пример 2: Ф: Лукьянов И: Филиппо вич О: (пустое значение)	для примера 1, 2: Ф: Лукьянов И: (пустое значение) О: Филиппович	
7	MISPRI NT	Исправлена опечатка	Исправлена опечатка в одну букву	ФИО: Кол арски Цветоми р Стефанов ич	Ф: Коларски И: Цветамир О: Стефанович	Изменили одну букву в имени и нашли компонент в справочнике имен
8	UNNATURAL	Компонент взят из позиции для	В исходной строке при «объединении» полей получается «неестественный» порядок ФИО	пример 1: Иван Иванов В	для примера 1 и 2: Ф: Иванов И: Иван	пример 1: ФИО передано одной строкой в

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
		другого компонента	— например, ИФО	ладими р вич пример 2: Ф: <b>Иван</b> И: <b>Иванов</b> О: Влади мирович	О: Владимирович	порядке, отличном от «фамилия — имя — отчество». пример 2: ФИО передано гранулярно, при «объединении» полей исходного ФИО получается «неестественный» порядок ФИО — ИФО
9	OK_NO T_CHA NGED	Корректное значение* (устар.)  *статус не показываем в интерфейсе	Разобрано, без изменений найдено в справочнике	ФИО: Лукьянов Владими р Филиппо вич	Ф: Лукьянов И Владимир О: Филиппович	
10	OK_CH ANGED	Приведено к справочнику (устар.)	Разобрано, приведено к справочнику	ФИО: Кожевни кова Аллиса Борисовна	Ф: Кожевникова И Аллиса О: Борисовна	
11	ONLY_C ONSONA NTS	Только согласные буквы	Компонент длиной 3 или более символов состоит только из согласных русских букв	ФИО: Иванов <b>Т</b> <b>ргдв</b> Бори сович	Ф: Иванов И: <b>Тргдв</b> О: Борисович	
12	ONLY_V OWELS_ SHORT	Только гласные буквы	Компонент длиной 3 символа состоит только из гласных русских букв	ФИО: Иванов <b>А</b> <b>эо</b> Борисо вич	Ф: Иванов И: <b>Аэо</b> О: Борисович	
13	ONLY_V OWELS_ LONG	Только гласные буквы	Компонент длиной 4 или более символа состоит только из гласных русских букв	ФИО: Иванов <b>А</b> <b>яиеу</b> Бор исович	Ф: Иванов И: <b>Аяиеу</b> О: Борисович	
14	IDENTI CAL_LE TTERS	Больше 2 одинаковых буквы подряд	Компонент содержит подряд 3 или более одинаковые русские буквы	ФИО: Иванов <b>А</b> <b>ааандрей</b> Борисович	Ф: Иванов И: <b>Ааандрей</b> О: Борисович	
15	VOWELS _LETTE RS	Больше 2 гласных подряд	Компонент содержит подряд 3 или более гласных русских буквы	ФИО: Иванов <b>Б</b> <b>оыэрис</b> Б орисович	Ф: Иванов И: <b>Боыэрис</b> О: Борисович	
16	CONSON ANTS_L ETTERS	Больше 3 согласных подряд	Компонент содержит подряд 4 или более согласных русских буквы	ФИО: Иванов <b>К</b> <b>ртдвени</b> <b>мин</b> Бори сович	Ф: Иванов И: <b>Кртдвениамин</b> О: Борисович	
17	FEW_UN IQUE_L ETTERS	Мало уникальных букв	Для длины компонента содержится мало уникальных русских букв	ФИО: Иванов <b>Га</b> <b>яача</b> Бори	Ф: Иванов И: <b>Гаяача</b> О: Борисович	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
				сович		
18	VERY_FEW_UNIQUE_LETTERS	Очень мало уникальных букв	Для длины компонента содержится очень мало уникальных русских букв	ФИО: Иванов <b>Б</b> <b>обоб</b> Бор исович	Ф: Иванов И: <b>Бобоб</b> О: Борисович	
19	UNUSUAL_SUBSTRING	Значение похоже на мусор	Компонент содержит трехбуквенную последовательность, которая не встречается в справочных ФИО	ФИО: Иванов <b>В</b> <b>плапъ</b> Бо рисович	Ф: Иванов И: <b>Вплапъ</b> О: Борисович	
20	RARE_SUBSTRING	Значение похоже на мусор	Компонент содержит как минимум две трехбуквенных последовательности, которые являются очень редкими для справочных ФИО	ФИО: Иванов <b>О</b> <b>лэуос</b> Бор исович	Ф: Иванов И: <b>Олэуос</b> О: Борисович	

## Отображение в интерфейсе

Исходное ФИО Кожевникова Аллиса Борисовна 

■ Сомнительное значение (ФИО изменено)

Фамилия

Имя   
Приведено к справочнику

Отчество

Пол  ▾  
Исходный не задан

# МАРКЕРЫ ПРОВЕРКИ И ИЗМЕНЕНИЯ ОТЧЕСТВА


## Гранулярный код отчества

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	STRONG _CHANGE	Произведена замена по справочникам	Компонент изменен по справочникам опечаток и уменьшительно-ласкательных слов	ФИО: Иванов Юрий <b>А</b> дре евич	Ф: Иванов И: Юрий О: <b>А</b> ндреевич	Фамилию, имя и отчество можно указать в одном поле, система сама разделит их на отдельные составляющие
2	TRANSSEXUAL	Изменено окончание по полу	Изменено окончание в соответствии с определенным по ФИО полом	ФИО: Иванова Ирина <b>А</b> ндреевич	Ф: Иванова И: Ирина О: Андреев <b>н</b> а	
3	PREFIX	Иностраный префикс	Компонент содержит иностранную приставку: ДЕ, Д', ФОН, ДЕЗ, ВАН, ГЕР, О', МАК, ЛЕ, ДА	Ф: Иванов И: Иван О: <b>Д</b> е Иванович	Ф: Иванов И: Иван О: <b>Д</b> е Иванович	
4	MISPRI NT	Исправлена опечатка	Исправлена опечатка в одну букву			Изменили одну букву в отчестве и нашли компонент в справочнике отчеств
5	UNNATURAL	Компонент взят из позиции для другого компонента	В исходной строке при «объединении» полей получается «неестественный» порядок ФИО — например, ИФО			
6	DEFICIENCY	Присутствует инициал	Вместо полного значения указан инициал	ФИО: Иванова Ирина <b>А</b>	Ф: Иванова И: Ирина О: <b>А</b>	
7	TOO_LONG	Значение слишком длинное	Компонент содержит более 15 символов	ФИО: Иванова Ирина <b>Х</b> ри сторожес твенская	Ф: Иванова И: Ирина О: <b>Х</b> ристорождественская	
8	UNLIST	Отсутствует в справочниках	Распознанный компонент отсутствует в справочниках	ФИО: Иван ова Ирина <b>Х</b> ристор ожес твенская	Ф: Иванова И: Ирина О: <b>Х</b> ристорождественская	
9	EMPTY	Не найдено в исходной строке	Не найдено в исходной строке (возможно, пустая строка)	ФИО: Лукьянов Владимир	Ф: Лукьянов И: Владимир О: (пустое значение)	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
10	OK_NO T_CHA NGED	Корректное значение (устар.)  *статус не показываем в интерфейсе	Разобрано, без изменений найдено в справочнике	ФИО: Лукьянов Владимир Филиппов ич	Ф: Лукьянов И: Иван О: Филиппович	
11	OK_CH ANGED	Приведено к справочнику (устар.)	Разобрано, приведено к справочнику	ФИО: Венцеров Юрий Алекс адрович	Ф: Венцеров И: Юрий О: Александрович	
12	ONLY_C ONSONA NTS	Только согласные буквы	Компонент длиной 3 или более символов состоит только из согласных русских букв	ФИО: Васи льев Андре й Тргдвч	Ф: Васильев И: Андрей О: Тргдвч	
13	ONLY_V OWELS_ SHORT	Только гласные буквы	Компонент длиной 3 символа состоит только из гласных русских букв	ФИО: Васильев А ндрей Аоэ	Ф: Васильев И: Андрей О: Аоэ	
14	ONLY_V OWELS_ LONG	Только гласные буквы	Компонент длиной 4 или более символа состоит только из гласных русских букв	ФИО: Васильев А ндрей Иаэ ия	Ф: Васильев И: Андрей О: Иаэия	
15	IDENTI CAL_LE TTERS	Больше 2 одинаковых буквы подряд	Компонент содержит подряд 3 или более одинаковые русские буквы	ФИО: Васильев А ндрей Ееер дович	Ф: Васильев И: Андрей О: Ееердович	
16	VOWELS _LETTE RS	Больше 2 гласных подряд	Компонент содержит подряд 3 или более гласных русских буквы	ФИО: Васильев А ндрей Бой эндарчуко вич	Ф: Васильев И: Андрей О: Бойэндарчукович	
17	CONSON ANTS_L ETTERS	Больше 3 согласных подряд	Компонент содержит подряд 4 или более согласных русских буквы	ФИО: Васильев А ндрей Крт двишкино вич	Ф: Васильев И: Андрей О: Кртдвишкинович	
18	FEW_UN IQUE_L ETTERS	Мало уникальных букв	Для длины компонента содержится мало уникальных русских букв	ФИО: Васильев А ндрей Хаяа га	Ф: Васильев И: Андрей О: Хаяага	
19	VERY_F EW_UNI QUE_LE TTERS	Очень мало уникальных букв	Для длины компонента содержится очень мало уникальных русских букв	ФИО: Васильев А ндрей Бобо бо	Ф: Васильев И: Андрей О: Бобобо	
20	UNUSUA L_SUBS TRING	Значение похоже на мусор	Компонент содержит трехбуквенную последовательность, которая не встречается в справочных ФИО	ФИО: Васильев А ндрей Впл апь	Ф: Васильев И: Андрей О: Вплапь	
21	RARE_S UBSTRIN G	Значение похоже на мусор	Компонент содержит как минимум две трехбуквенных последовательности, которые	ФИО: Васильев А ндрей Yoco	Ф: Васильев И: Андрей О: Yocolé	

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
			являются очень редкими для справочных ФИО	лэ		

## Отображение в интерфейсе

Исходное ФИО Венцеров Юрий Александрович 

■ Сомнительное значение (ФИО изменено)

Фамилия

Имя

Отчество

Приведено к справочнику

Пол  ▾

Исходный не задан

# КОД КАЧЕСТВА ПОЛА

## Гранулярный код качества пола

Может быть несколько значений

№	Литера	Локализация	Описание	Исходно	Стандартизировано
1	OK_NOT_CHANGED	Корректный* *статус не показываем в интерфейсе	Пол определён и совпал с исходным полом	ФИО: Захлебин Василий Меркурьевич пол: мужской	Фамилия: Захлебин Имя: Василий Отчество: Меркурьевич пол: мужской
2	OK_CHANGED	Корректный, не совпадает с исходным	Пол определён, не совпал с исходным полом	ФИО: Захлебин Василий Меркурьевич пол: женский	Фамилия: Захлебин Имя: Василий Отчество: Меркурьевич пол: мужской
3	UNDEF	Исходный не задан	Исходный пол не задан	ФИО: Захлебин Василий Меркурьевич пол: (пустое значение)	Фамилия: Захлебин Имя: Василий Отчество: Меркурьевич пол: мужской
4	AMBI	Не однозначен	Результирующий пол неоднозначен	ФИО: Кочетов Эрнестина Эрнестинович пол: (пустое значение)	Фамилия: Кочетов Имя: Эрнестина Отчество: Эрнестинович пол: не определен

## Отображение в интерфейсе

Автор: AL:503402443

Исходное ФИО: Иванов Екатерина Бекмамдоровна  
Сомнительное значение (пол неоднозначен, ФИО изменено, произведена замена по справочникам, пол не определен, противоречие по полу)

Фамилия: Иванов

Имя: Екатерина  
Приведено к справочнику

Отчество: Бекмамдоровна  
Отсутствует в справочниках

Пол: Неизвестно  
Исходный не задан, не однозначен

Стандартизовать Подтвердить Отложить

## ПРОЧИЕ КОДЫ КАЧЕСТВА

- Код качества наименования ЮЛ
- Код качества ИНН ЮЛ
- Код качества КИО
- Код качества КПП
- Код качества ОГРН
- Код качества ОКПО
- Код качества СНИЛС
- Код качества ОКВЭД
- Код качества VIN-номера
- Код качества даты рождения
- Популярность атрибута
- Признак террориста



# КОД КАЧЕСТВА НАИМЕНОВАНИЯ ЮЛ

№	Литера	Локализация	Описание
1	<b>Не требуют проверки человеком</b>		
2	CONFIRMED_ MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил наименование организации и/или подтвердил его
3	EDITED	Правильное значение	Правильное значение
4	<b>На ручную обработку</b>		
5	MISSED	Неизвестный код	Неизвестный код
6	NOT_SURE	Сомнительное значение	Сомнительное значение

# КОД КАЧЕСТВА ИНН ЮЛ

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUAL	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил ИНН и/или подтвердил его	1430567890	1430567890
3	GOOD	Корректное значение	Все проверки прошли, ИНН корректный	1430567890	1430567890
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)
5	GOOD_CHANGED	Убран мусор	Кроме корректного ИНН исходное значение содержит другие «мусорные» символы, которые были убраны	INN 1430567890	1430567890
<b>6 На ручную обработку</b>					
7	NOT_VALID_CHECKSUM	Не совпадает контрольная сумма	Рассчитанное контрольное число отличается от соответствующего знака в ИНН	7722523661	(пустое значение)
8	NOT_VALID_FORMAT	Некорректное значение	В исходных данных содержится только мусорное текстовое значение и/или ИНН состоит из 10 нулей	пример 1: 0000000000 пример 2: текст	пример 1: (пустое значение) пример 2: (пустое значение)
9	NOT_VALID_LENGTH	Некорректная длина	Длина ИНН не равна 10	516281903	(пустое значение)
10	NOT_VALID_TAX_CODE	Некорректный код налоговой	Код налогового органа не найден в справочнике (первые четыре цифры)	0000000040	(пустое значение)
11	MISSED	Отсутствует код качества	Атрибут не проходил стандартизацию		
<b>12 Устаревшие значения</b>					
13	CONFIRMED	Значение подтверждено (ЕГРЮЛ) (устар.)			
14	RESTORED	Значение восстановлено (ЕГРЮЛ) (устар.)			
15	CHANGED	Значение изменено (ЕГРЮЛ) (устар.)			
16	NOT_SURE	Сомнительное значение (устар.)			

---

17 EDITED Правильное  
значение (устар.)

---

# КОД КАЧЕСТВА КИО

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил КИО и/или подтвердил его	КИО: 00000 ИНН: 7712092445 092445	КИО: 09244 ИНН: 7712092445
3	GOOD	Корректное значение	Все проверки прошли, КИО корректный	КИО: 09244 ИНН: 7712092445 092445	КИО: 09244 ИНН: 7712092445
4	GOOD_CHANGED	Корректное значение	Нецифровые символы удалены, КИО корректный	КИО: уу092р44нн ИНН: дддд77120 92445а	КИО: 09244 ИНН: 7712092445
5	EMPTY	Пустое значение	Исходное значение КИО пустое или после удаления нецифровых символов значение КИО стало пустым	(пустое значение)	(пустое значение)
<b>6 На ручную обработку</b>					
7	NOT_VALID	Некорректное значение	Значение КИО не соответствует заполненному значению ИНН	КИО: 09248 ИНН: 7712092445 092445	КИО: (пустое значение) ИНН: 7712092445
8	NOT_VALID_FORMAT	Некорректное значение	Значение КИО содержит больше или меньше 5 цифр, либо все 5 цифр одинаковые и ИНН пустой	КИО: 11111 ИНН: (пустое значение)	КИО: (пустое значение) ИНН: (пустое значение)
9	MISSED	Отсутствует код качества	Атрибут не проходил стандартизацию		

# КОД КАЧЕСТВА КПП

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
1	<b>Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил КПП и/или подтвердил его	11111111	503601001	Любые данные, подтвержденные контролером, имеют статус «CONFIRMED_MANUALLY»
3	GOOD	Корректное значение	Все проверки прошли, КПП корректный	503601001	503601001	
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)	
5	GOOD_CHANGED	Убран мусор	Кроме корректного КПП исходное значение содержит другие «мусорные» символы, которые были убраны	KPP 503601001	503601001	
6	INCORRECT_RESIDENT	Не соответствует резидентству	Причина постановки на учет (5-й и 6-й знаки КПП) не соответствует признаку резидента. Допустимые значения для российских организаций – от 1 до 50, для иностранных – от 51 до 99	КПП: 771801001 признак резидента: нерезидент	КПП: 771801001 признак резидента: нерезидент	Причина постановки на учет (5 и 6 знаки КПП) соответствует значению «01» (резидент), при этом указано, что организация – нерезидент
7	<b>На ручную обработку</b>					
8	NOT_VALID_LID	КПП не соответствует ИНН	Код налоговой КПП не соответствует коду налоговой ИНН (первые четыре цифры КПП и ИНН). Ситуация несоответствия допустима в случае, когда ЮЛ меняет локацию и встает на учет в новом налоговом органе (ИНН остаётся прежним, а КПП меняется)	КПП: 77260111 ИНН: 2721225692	КПП: (пустое значение) ИНН: 2721225692	Код налоговой КПП («7726») не соответствует коду налоговой ИНН («2721»)
9	NOT_VALID_FORMAT	Некорректное значение	Исходное значение содержит только мусор	11111111	(пустое значение)	

10	NOT_VA LID_LE NGTH	<b>Некорректная длина</b>	Исходное значение некорректно, т.к. не проходит проверку на длину (допустимо 9 символов)	2345680 (пустое значение)	Исходное значение имеет 7 символов при допустимой длине КПП 9 символов
11	NOT_VA LID_RE ASON_C ODE	<b>Некорректная причина</b>	Некорректная причина постановки на учёт (5-6 знаки КПП). Примеры кодов причин: <ul style="list-style-type: none"> <li>• 01 — КПП присвоен организации в связи с постановкой на учет по месту ее нахождения;</li> <li>• 02, 03, 04, 05, 31 или 32 — КПП присвоен организации по месту нахождения обособленного подразделения организации;</li> <li>• 06, 08 — КПП присвоен организации по месту нахождения принадлежащего ей недвижимого имущества;</li> <li>• 10, 29 — КПП присвоен организации по месту нахождения принадлежащих ей транспортных средств в зависимости от вида транспортных средств;</li> <li>• 50 — КПП присвоен в связи с постановкой на учет в качестве крупнейшего налогоплательщика.</li> </ul>	664348462 (пустое значение)	Код причины постановки и на учёт («48») является несуществующим
12	NOT_VA LID_TA X_CODE	<b>Некорректный код налоговой</b>	Код налогового органа не найден в справочнике (с 1-го по 4-й знаки)	123456789 (пустое значение)	Код налогового органа («1234») является несуществующим
13	MISSED	Отсутствует код качества	Атрибут не проходил стандартизацию		
<b>14 Устаревшие значения</b>					
15	CONFIRMED	Значение подтверждено (ЕГРЮЛ) (устар.)			
16	RESTORED	Значение восстановлено (ЕГРЮЛ) (устар.)			
17	CHANGED	Значение изменено (ЕГРЮЛ)			

---

(устар.)

---

# КОД КАЧЕСТВА ОГРН

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
<b>1 Не требуют проверки человеком</b>						
2	CONFIRMED MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил ОГРН и/или подтвердил его	ОГРН 00000000 00000	1042129002630	Любые данные, подтвержденные контролером, имеют статус «CONFIRMED MANUALLY»
3	GOOD	Корректное значение	Все проверки прошли, ОГРН корректный	10421290 02630	1042129002630	
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)	
5	GOOD_CHANGED	Убран мусор	Кроме корректного ОГРН исходное значение содержит другие «мусорные» символы, которые были убраны	OGRN 10 42129002 630	1042129002630	
<b>6 На ручную обработку</b>						
7	NOT_VALID_CHECKSUM	Не совпадает контрольная сумма	Последняя цифра ОГРН не совпадает с контрольной суммой	30459042 0900227	(пустое значение)	Последняя цифра ОГРН «7» не совпадает с контрольной суммой, рассчитываемой по алгоритму
8	NOT_VALID_FIRST_DIGIT	Некорректная первая цифра	Значение некорректно, т.к. не проходит проверку на начальную цифру	14588339 5000341	(пустое значение)	Для 15-символьного ОГРН (т.е. ОГРНИП) первая цифра не может иметь значение «1» (допустимо «3» или «4»)
9	NOT_VALID_FORMAT	Некорректное значение	Исходное значение содержит только «мусорные» значения	OGRN 00000000 00000	(пустое значение)	



10	NOT_VA LID_LE NGTH	<b>Некорректная длина</b>	Значение некорректно, не проходит проверку на длину	<b>30238018 94379888</b> (пустое значение)	ОГРН должен состоять из 13 или 15 (ОГНИП) цифр
11	NOT_VA LID_SU BJECT	<b>Некорректны й код субъекта</b>	Код субъекта РФ (4-й и 5-й знаки) не найден в справочнике код региона (согласно 65-й статье Конституции)	<b>30493042 0900226</b> (пустое значение)	Код субъекта « 93» не найден в справочни ке код региона
12	NOT_VA LID_YE AR	<b>Некорректны й год</b>	Некорректный год внесения записи в ЕГРЮЛ. Год (вторая и третья цифра ОГРН) должны соответствовать двум последним цифрам года внесения записи в государственный реестр. Допустимое значение от 02 до двух последних цифр текущего года (на момент стандартизации)	<b>17733000 00000</b> (пустое значение) текущий год на момент стандарт изации: 2022	Год (вторая и третья цифра) ОГ РН могут быть от «02 » до последних двух цифр текущего года
13	MISSED	Отсутствует код качества	Атрибут не проходил стандартизацию		
<b>14 Устаревшие значения</b>					
15	CONFI RMED	Значение подтвержден о (ЕГРЮЛ) (устар.)			
16	RESTO RED	Значение восстановлен о (ЕГРЮЛ) (устар.)			
17	CHAN GED	Значение изменено (ЕГРЮЛ) (устар.)			

# КОД КАЧЕСТВА ОКПО

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значение ОКПО и/или подтвердил его	0001	49440748
3	GOOD	Корректное значение	Все проверки прошли, ОКПО корректный	49440748	49440748
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)
5	GOOD_CHANGED	Убран мусор	Кроме корректного ОКПО исходное значение содержит другие «мусорные» символы, которые были убраны	ОКПО 49440748	49440748
<b>6 На ручную обработку</b>					
7	NOT_VALID_CHECKSUM	Не совпадает контрольная сумма	Значение некорректно, не проходит проверку на контрольную сумму	49440749	(пустое значение)
8	NOT_VALID_FORMAT	Некорректный формат	Значение ОКПО имеет недопустимую длину числового значения (допустимо 8 или 10 цифр)	00000000 00000	(пустое значение)
9	MISSED	Неизвестный код качества	Атрибут не проходил стандартизацию		

# КОД КАЧЕСТВА СНИЛС

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил СНИЛС и/или подтвердил его	СНИЛС 000000000 00	165-300-180 28
3	GOOD	Корректное значение	Все проверки прошли, СНИЛС корректный	165-300-180 28	165-300-180 28
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)
5	GOOD_CHANGED	Убран мусор	Кроме корректного СНИЛС исходное значение содержит другие «мусорные» символы, которые были убраны	СНИЛС-16 5-300-180 28	165-300-180 28
<b>6 На ручную обработку</b>					
7	NOT_VALID_CHECKSUM	Не совпадает контрольная сумма	Контрольная сумма в исходном СНИЛС не проходит проверку	165300180 10	(пустое значение)
8	NOT_VALID_LENGTH	Некорректная длина	Исходное значение слишком короткое	166-343-434	(пустое значение)
9	NOT_VALID_FORMAT	Некорректное значение	Исходное значение некорректно: состоит из нулей	000-000-000 00	(пустое значение)
10	MISSED	Неизвестный код качества	Атрибут не проходил стандартизацию		

# КОД КАЧЕСТВА ОКВЭД

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значение ОКВЭД и/или подтвердил его	###	02.01.1
3	GOOD	Корректное значение	Все проверки прошли, код ОКВЭД корректный	пример 1: 02.01.1 пример 2: 64.19 64.20	пример 1: 02.01.1 пример 2: 64.19 64.20
4	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)
5	GOOD_CHANGED	Убран мусор	Исходное значение содержит корректный код ОКВЭД совместно с другими «мусорными» символами, которые были удалены	пример 1: оквэд 02.01.1 пример 2: 64.19, 64.20	пример 1: 02.01.1 пример 2: 64.19 64.20
<b>6 На ручную обработку</b>					
7	NOT_VALID_FORMAT	Некорректное значение	Исходное значение содержит только «мусор», либо структура кода не соответствует ОКВЭД	пример 1: оквэд ### пример 2: 64196420	пример 1: (пустое значение) пример 2: (пустое значение)
8	NOT_VALID_CODE	Некорректный код	Несуществующее значение ОКВЭД	00.00	(пустое значение)
9	NOT_VALID_LID	Значение некорректно	Значение некорректно, а именно: если в исходной строке несколько ОКВЭД и последнее значение корректно, но для одного из предыдущих ОКВЭД указано несуществующее или мусорное значение	6420 64.20	64.20
10	MISSED	Отсутствует код качества	Атрибут не проходил стандартизацию		

# КОД КАЧЕСТВА VIN-НОМЕРА

Мусорные строки получают код качества **EMPTY** и не попадут на ручной разбор

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Описание примера
<b>1 Не требуют проверки человеком</b>						
2	CONFIRMED_MANUALLY	Подтвержден о контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил VIN-номера и/или подтвердил его	00000	WMWRC31030TK98375	Любые данные, подтвержденные контролером, имеют статус «CONFIRMED_MANUALLY»
3	NOT_CHANGED	Корректное значение	Все проверки прошли, VIN-номер корректный	WMWRC31030TK98375	WMWRC31030TK98375	VIN-номер корректный
4	CHANGED	Номер исправлен	Исходный VIN-номер содержал кириллические символы, которые были исправлены	WMWRC31030TK98375	WMWRC31030TK98375	Исходный VIN-номер содержит кириллические символы «М» и «С» (схожие по написанию с символами латиницы), которые были исправлены на символы латиницы
<b>5 На ручную обработку</b>						
6	NOT_VALID	Пустое значение	Исходный VIN-номер некорректный	WMWRC31030TK98375	(пустое значение)	Последние 4 знака VIN-номера должны состоять из цифр (в примере 3 цифры)
7	EMPTY	Пустое значение	Исходное значение пустое	(пустое значение)	(пустое значение)	Исходное значение пустое
		Пустое значение	<p>Вместо VIN-номера указано значение, которое определяется как мусорное и подлежит удалению при стандартизации. Причины:</p> <ul style="list-style-type: none"> <li>• строка без чисел и русских или английских символов — только спецсимволы или пробелы;</li> <li>• повтор любого символа внутри VIN, кроме 0 и 1, 4 раза</li> <li>• повтор «1» более 5 раз</li> <li>• повтор «0» более 6 раз</li> <li>• только цифры, без букв</li> </ul>	<p>пример 1: 12345678901234567</p> <p>пример 2: WMWRC00000TK975</p>	<p>пример 1: (пустое значение)</p> <p>пример 2: (пустое значение)</p>	<p>Пример 1: исходный VIN-номер не может состоять только из цифр</p> <p>Пример 2: исходный VIN-номер не может содержать цифру «0» более 6 раз</p>

---

---

8	MISSED	Неизвестный код качества	Атрибут не проходил стандартизацию
---	--------	--------------------------	------------------------------------

---

# КОД КАЧЕСТВА ДАТЫ РОЖДЕНИЯ

№	Литера	Локализация	Описание	Исходно	Стандартизировано
1	<b>Не требуют проверки человеком</b>				
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил дату рождения и/или подтвердил ее	17.03.2070	17.03.1970
3	EDITED	Правильное значение	Все проверки прошли, дата рождения корректная	17-03-2070	17.03.1970
4	<b>На ручную обработку</b>				
5	EMPTY_DATE	Пустая дата	Пустое значение даты	(пустое значение)	(пустое значение)
6	PAST_DATE	Дата из прошлого	Дата меньше либо равна 01.01.1900	01.01.1900	(пустое значение)
7	FUTURE_DATE	Дата из будущего	На момент стандартизации дата рождения являлась датой из будущего	17.10.3021	(пустое значение)
8	CURRENT_DATE	Текущая дата	На момент стандартизации дата рождения являлась текущей датой или датой, на 1-2 дня меньше текущей. Проверка на то, что по ошибке в анкете не забита текущая дата вместо даты рождения (-1-2 дня — запас на то, пока данные дойдут по интеграциям до ЕК)	16.11.2021* *текущая дата на момент стандартизации	(пустое значение)
9	BAD_FORMAT_DATE	Некорректная дата	Несуществующая дата или неверный формат даты (когда дата задана в виде строки)	пример 1: 30.02.2018* *тридцатого февраля не бывает пример 2: 197дек33	(пустое значение)
10	GARBAGE_DATE	Мусорное значение	Дата найдена в справочнике мусорных дат рождения	01.06.2006* *дата из примера может не являться мусорной в реальной базе заказчика	(пустое значение)
11	DEFAULT_DATE	Значение по умолчанию	Дата найдена в справочнике дефолтных дат рождения	01.01.1753* *дата из примера может не являться дефолтной в реальной базе заказчика	(пустое значение)
12		Меньше	С указанной даты рождения прошло не более	01.06.2008*	(пустое значение)

	YOUNG_ DATE	возраста обслуживания	четырнадцать лет на момент стандартизации данных	*на момент 2021 года
13	MISSED	Неизвестный код качества	Атрибут не проходил стандартизацию	
<b>14 Устаревшие значения</b>				
15	NOT_SURE	Сомнительное значение	Сомнительное значение	
16	GOOD_DATE	null (устар.)	null (устар.)	



## ПОПУЛЯРНОСТЬ АТТРИБУТА

№	Литера	Локализация	Описание
1	NONE	Обычный	Популярности нет или она не определена
2	POPULAR	Популярный	Часто встречаемое значение (сомнительно)
3	VERY_POPULAR	Массовый	Очень часто встречаемое значение (плохо)

# ПРИЗНАК ТЕРРОРИСТА

№	Литера	Локализация	Описание	Исходно	Стандартизировано	Пояснение примера
1	NOT_TERRORIST	Не террорист	Совпадений в перечне террористов не найдено	ФИО: Иванов Иван Иванович признак террориста: не известен	ФИО: Иванов Иван Иванович признак террориста: не террорист	Физическое лицо с такими ФИО не найдены ни в одном из черных списков
2	SUSPECT	Подозрительный	Найдено совпадение в перечне террористов	ФИО: Абдулло в Равиль Хусяинович ДР: 03.06.1959 признак террориста: не известен	ФИО: Абдулло Равиль Хусяинович ДР: 03.06.1959 признак террориста: подозрительный	Физическое лицо с такими ФИО и ДР найден в справочнике террористов
3	UNKNOWN	Неизвестен	Проверка по перечню террористов не выполнялась	ФИО: Абдулло Равиль Хусяинович ДР: 03.06.1959 признак террориста: не известен	ФИО: Абдулло Равиль Хусяинович ДР: 03.06.1959 признак террориста: неизвестен	Проверка по перечню террористов не выполнялась

# КОД КАЧЕСТВА SWIFT

№	Литера	Локализация	Описание	Исходно	Стандартизировано
<b>1 Не требуют проверки человеком</b>					
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значение SWIFT и/или подтвердил его	###	EKNCRU4EXXX
3	EDITED	Корректное значение	Все проверки прошли, SWIFT корректный	EKNCRU4 EXXX	EKNCRU4EXXX
<b>4 На ручную обработку</b>					
5	NOT_SURE	Сомнительное значение	Исходное значение не удовлетворяет формату SWIFT	SA78RRU 8KBLGQA S@#	(пустое значение)

# КОД КАЧЕСТВА БИК

№	Литера	Локализация	Описание	Исходно	Стандартизировано
1	<b>Не требуют проверки человеком</b>				
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком. Менеджер данных вручную исправил значение БИК и/или подтвердил его	###	042304001
3	EDITED	Корректное значение	Все проверки прошли, БИК корректный	042304001	042304001
4	<b>На ручную обработку</b>				
5	NOT_SURE	Сомнительное значение	Исходное значение не удовлетворяет формату БИК	04001@#	(пустое значение)

# КОД КАЧЕСТВА РЕКВИЗИТА

№	Литера	Локализация	Описание
1	<b>Не требуют проверки человеком</b>		
2	CONFIRMED_MANUALLY	Подтверждено контролером	Корректность значения подтверждена человеком
3	EDITED	Правильное значение	Все проверки прошли, реквизит корректный
4	<b>На ручную обработку</b>		
5	NOT_SURE	Сомнительное значение	Сомнительное значение
6	MISSED	Неизвестный код	Атрибут не проходил стандартизацию
7	UNPROCESSED	Значение без стандартизации	Значение не изменено механизмами стандартизации

# КОДЫ КАЧЕСТВА ДОКУМЕНТА

Коды могут использоваться, например, в [CleanService.passport](#) (если используются старая версия фильтра документов). Коды качества нового фильтра указаны на странице [Код качества документа ФЛ](#)

Коды указаны по убыванию приоритета

№	Литера	Локализация	Описание	Исходно	Стандартизировано
1	<b>Не требуют проверки человеком</b>				
2	CONF IRMED_ MANUA LLY	<a href="#">Данные о документе подтверждены контролером</a>	Корректность значений подтверждена контролером (любые данные, подтвержденные контролером, имеют статус «CONFIRMED_MANUALLY»)	Тип документа: паспорт гражданина РФ Серия и номер паспорта: 4 014785436 Дата выдачи паспорта: 2 7.08.2014 Код подразделения: 780-121 Место выдачи паспорта: У ФМС по Санкт-Петербургу	Тип документа: паспорт гражданина РФ Серия: 40 14 Номер: 785436 Дата выдачи паспорта: 27.08.2014 Код подразделения: 780-121 Место выдачи паспорта: УФМС по Санкт-Петербургу
3	EDIT ED	<a href="#">Правильное значение</a>	Данные о документе прошли все проверки (в т.ч. сверка даты выдачи паспорта с датой рождения лица), документ корректный	Тип документа: паспорт гражданина РФ Серия и номер паспорта: 4 014785436,, Дата выдачи паспорта: 2 7.08.2014 Код подразделения: 780-121 Место выдачи паспорта: У ФМС по Санкт-Петербургу	Тип документа: паспорт гражданина РФ Серия: 40 14 Номер: 785436 Дата выдачи паспорта: 27.08.2014 Код подразделения: 780-121 Место выдачи паспорта: УФМС по Санкт-Петербургу

				Дата рождения лица: 27.08.2000	
4	EMPT_Y_DATE	Пустая дата выдачи	Пустая дата выдачи документа		
<b>5 На ручную обработку</b>					
6	BAD_SERIES_NUMBER	Плохая серия или номер	Некорректный формат серии, недействительная серия, некорректный формат номера паспорта гражданина РФ	Тип документа: паспорт гражданина РФ Серия и номер паспорта: 000654343	Тип документа: паспорт гражданина РФ Серия: 0000 Номер: 654343
7	SER_ISSUE_ISMAT_CN	Дата выдачи не соответствует серии	Дата выдачи паспорта гражданина РФ (24.10.2007) не соответствует серии документа. Вторая пара цифр серии («?17»? ) — год выпуска бланка, который может быть на 3 года меньше или на 5 лет больше даты выдачи документа	Тип документа: паспорт гражданина РФ Серия и номер паспорта: 4017 766901 Дата выдачи паспорта: 24.10.2007 Код подразделения: 780-119 Место выдачи паспорта: УФМС по Санкт-Петербургу	Тип документа: паспорт гражданина РФ Серия: 4010 Номер: 766901 Дата выдачи паспорта: 24.10.2007 Код подразделения: 780-119 Место выдачи паспорта: УФМС по Санкт-Петербургу
8	BAD_DATE	Плохая дата выдачи	Дата выдачи сильно в прошлом или из будущего	Тип документа: паспорт гражданина РФ Серия и номер паспорта: 4014785436 Дата выдачи паспорта: 01.01.1900	Тип документа: паспорт гражданина РФ Серия: 40 14 Номер: 785436 Дата выдачи паспорта: 01.01.1900
9	BAD_CODE	Плохой код подразделения	Некорректный документ (устаревший код качества, с апреля 2017 не проставляется)	Тип документа: паспорт гражданина РФ	Тип документа: паспорт гражданина РФ Серия: 40 14 Номер: 785436

---

Серия и номер паспорта: 4014785436	Дата выдачи паспорта: 27.08.2014
Дата выдачи паспорта: 27.08.2014	Код подразделения: 123
Код подразделения: 123	

---

10	NOT_SURE	Сомнительное значение
----	----------	-----------------------

---



# ГОРЯЧИЕ КЛАВИШИ

- Списки контрагентов / дубликатов
- Карточка контрагента
- Группы дубликатов
- Прогноз слияния
- Дополнительные

## Списки контрагентов / дубликатов

Chrome / Firefox	Internet Explorer	Действие
<b>Alt + F</b>	<b>Ctrl + Shift + F</b>	Переход к строке поиска
<b>Alt + E</b>	<b>Ctrl + Shift + E</b>	Очистка строки поиска

## Карточка контрагента

Chrome / Firefox	Internet Explorer	Действие
<b>Alt + вправо</b>		Перейти к следующей карточке
<b>Alt + влево</b>		Перейти к предыдущей карточке
<b>Alt + вверх</b>		Вернуться к списку контрагентов
<b>Alt + E</b>	<b>Ctrl + Shift + E</b>	Очистить текущее поле формы
<b>Alt + Shift + E</b>	-	Очистить все поля формы
<b>Alt + C</b>	<b>Ctrl + Shift + C</b>	Стандартизировать атрибут
<b>Alt + A</b>	<b>Ctrl + Shift + A</b>	Подтвердить атрибут
<b>Alt + D</b>	<b>Ctrl + Shift + D</b>	Удалить атрибут
<b>Ctrl + клик</b>		Переход в режим копирования адресов (работает, когда более 1 адреса у контрагента)
<b>Tab</b>		Перевод курсора между свёрнутыми атрибутами/реквизитами или переход между полями раскрытого атрибута/реквизита
<b>Enter / Space</b>		При наведённом курсоре — раскрытие атрибута/реквизита

\* аналогичные комбинации работают и в **табличном режиме**

## Группы дубликатов

Сочетание клавиш	Действие
<b>Alt + вправо</b>	Перейти к следующей группе дубликатов
<b>Alt + влево</b>	Перейти к предыдущей группе дубликатов
<b>Alt + вверх</b>	Вернуться к списку групп дубликатов
<b>Alt + Y / N / L</b>	Подтвердить / Отклонить дубликат / Отложить принятие решения
<b>Alt + K</b>	Переключение фильтра <i>Все атрибуты</i> / <i>Конфликтующие</i> (работает на странице прогноза)

## Прогноз слияния

Сочетание клавиш	Действие
<b>Tab</b>	Перевод курсора между свёрнутыми атрибутами/реквизитами или переход между полями раскрытого атрибута/реквизита
<b>Enter / Space</b>	При наведённом курсоре — раскрытие атрибута/реквизита

## Дополнительные

Сочетание клавиш	Действие
<b>Alt + Del, затем F5</b>	Сброс локальных настроек браузера

# РУКОВОДСТВО ОФИЦЕРА БЕЗОПАСНОСТИ

# ОПИСАНИЕ СИСТЕМЫ «ЕДИНЬЙ КЛИЕНТ»

Назначение системы

Функциональные возможности

Архитектура системы

Перечень защищаемой информации, обрабатываемой в системе

Режимы функционирования

## УПРАВЛЕНИЕ ПАРАМЕТРАМИ ПОДСИСТЕМЫ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ СИСТЕМЫ «ЕДИНЫЙ КЛИЕНТ»

Конфигурирование параметров системы

Управление доступом пользователей

Перечень доступных ролей

Управление аудитом событий системы

Обеспечение целостности данных (резервное копирование данных)

Настройка SSL (https)

Шифрование паролей

# ЗАЩИТА СИСТЕМЫ «ЕДИНЫЙ КЛИЕНТ» ОТ ВОЗДЕЙСТВИЯ ВРЕДОНОСНОГО КОДА

Для защиты системы «Единый клиент» от вредоносного кода выполнены процедуры:

1. Заблокировано проксирование библиотек зависимостей, используемых при сборке ПО.
2. Ограничена загрузка зависимостей, которые до этого не использовались при сборке ПО, датой до 24.02.2022. Используемые версии библиотек не превышают эту дату.
3. Выполняется подсчет и хранение хэш-сумм дистрибутивов при сборке и контроль неизменности данного атрибута при деплоimente на стенды разработки и стенды заказчиков.
4. Устранены уязвимости безопасности компонентов ПО и ОС обновлениями с датой релиза до 24.02.2022.
5. Реализована процедура проверки OSS-компонентов, загружаемых из внешних репозиториях, на уязвимости и лицензионную чистоту.

## НАСТРОЙКА SSL (HTTPS)

«Единый клиент» поддерживает работу по защищенному https-соединению. Заказчику необходимо выпустить SSL-сертификат и обратиться в Поддержку.

Ключи могут быть переданы в виде:

- **jks** — Java Key Store. Лучший вариант.
- **p12** — PKCS#12. Стандартное хранилище цепочки сертификатов и закрытого ключа.
- **cer, p7b, key** — Закрытый и открытый ключи в виде отдельных файлов.

# РЕЖИМЫ ФУНКЦИОНИРОВАНИЯ

«Единый клиент» функционирует в следующих режимах:

- штатный режим, при котором обеспечивается выполнение всех функций системы;
- режим обслуживания, необходимый для проведения обслуживания, реконфигурации и пополнения технических и программных средств системы новыми компонентами;
- аварийный режим работы – режим работы системы, при котором она не способна выполнять все или часть возложенных на нее функций.

Основным режимом функционирования системы «Единый клиент» является штатный режим, в котором выполняются все функции, реализованные в системе.

В штатном режиме функционирования обеспечивается доступность системы 98%.



# РУКОВОДСТВО ПО СОПРОВОЖДЕНИЮ

# РЕЛИЗЫ «ЕДИНОГО КЛИЕНТА»

## *Периодичность*

Релизы выпускаются каждые 2 недели.

Работа над новым релизом начинается сразу после даты окончания предыдущего релиза.

Если не было специальных доработок, то заказчик получает новую версию каждые 3 месяца. В случае специальных доработок (CR) — чаще.

После выпуска Заказчик оповещается письмом с Release Notes и инструкцией по установке новой версии.

## *Что происходит в течение релиза?*

1. Разрабатываем новый релиз:
  - a. core-функционал;
  - b. запросы на изменения от Заказчиков
2. Проводим функциональное тестирование (покрываем автотестами).
3. Проводим стресс-регрессию нагрузки.
4. Проводим регрессионное тестирование.

После завершения тестирования выкладываем свежие сборки на <https://fs.hflabs.ru>. Уведомляем о выпуске релиза письмом.

## *Когда мы прекращаем брать фичи в релиз?*

За три дня до планирования, если ресурсы не закончились раньше.

## *Где можно посмотреть список изменений по выпущенному релизу?*

В разделе Release notes спейса заказчика в Confluence. О выпуске сообщаем отдельным письмом.

# ДИАГНОСТИКА ПРОБЛЕМ

Диагностика инцидентов производится при помощи специальной утилиты.

При необходимости может быть дополнительно собрана трассировка БД.

- Утилита диагностики
- Read-only пользователь СУБД для диагностики
- Диагностические SQL-запросы
- Трассировка на уровне СУБД
- Lucene. Медленно открываются страницы в UI
- Проблемы с жестким диском на Windows — сбор информации
- Автоматический мониторинг логов
- Диагностика «зависаний» JVM с помощью jHiccup
- Как снять диагностику выполнения web-запросов (access.log)
- Профилирование удаленного стенда
- Мониторинг производительности систем на Linux
- Снятие полной профилировки на стенде
- Диагностика запросов из браузера

# УТИЛИТА ДИАГНОСТИКИ

Для сбора необходимой информации при расследовании инцидентов существует специальная утилита диагностики **cdi-diagnostic**. Работа с ней описана ниже.

- Установка
  - Пользователь СУБД
  - Установка утилиты
- Сбор диагностической информации
  - Интерактивный режим
  - Режим командной строки
  - Параметризация профилей
  - Результаты работы утилиты

## Установка

### Пользователь СУБД

В СУБД ЕК необходимо создать отдельного **read-only** пользователя, которого будет использовать утилита для сбора информации из БД.

### Установка утилиты

Утилита поставляется в архиве **cdi-diagnostic-{version}.zip**. Устанавливать ее следует на сервере приложений Единого клиента.

Для установки достаточно распаковать архив, после чего задать настройки для соединения с ЕК в конфигурационном файле **CDI\_DIAG\_DIR/config/diagnostic.properties** (где **CDI\_DIAG\_DIR** — это корневой каталог утилиты):

```
# Строка соединения с БД
# Имеет вид jdbc:oracle:thin:@{db-hostname}:{port}:{sid}
# или jdbc:oracle:thin:@://{db-hostname}:{port}/{service-name}
# в зависимости от того, используется ли в Oracle SID или SERVICE NAME.
cdi.jdbc.url = jdbc:oracle:thin:@dbserver:1521:cdi

# Имя диагностического пользователя СУБД и его пароль
cdi.jdbc.name = cdi_diag
cdi.jdbc.password = cdi
```

Строка соединения с БД хранится в файле **CDI\_JBOSS\_HOME\standalone\deployments\cdi-oracle-ds.xml**

```
<connection-url>jdbc:oracle:thin:@{db-hostname}:{port}:{sid}</connection-url>
```

## Сбор диагностической информации

### Интерактивный режим

Для запуска утилиты в интерактивном режиме необходимо вызвать ее исполняемый файл без дополнительных параметров.

Linux:

```
$/diagnostic.sh
```

Windows:

```
diagnostic.bat
```

После запуска утилита распечатает список доступных профилей диагностики, и предложит выбрать интересующие:

```
Please choose diagnostic profile:
# Profile          Description
-----
0 all profiles
1 cdi.jvm          CDI JVM information
2 config          Application config files
3 db.connection   DB connection information
4 factor.jvm      Factor JVM information
5 logs            CDI and Factor logs
6 search          Search engine information
7 sql.adhoc       Ad-hoc queries
8 sql.common      Content of application system tables
9 sql.counts     Business objects (party, attribute etc) counts
10 sql.db         Database stats
11 sql.duplicate  Duplicate parties statistics
12 sql.metadata   Application DB metadata
13 sql.party      Particular party information
14 sql.quality.physical Data quality statistics
15 system        Operating system information
16 versions      CDI and Factor versions
To exit, press 'q'.

Choose profile number(s):
```

Можно указать номер конкретного профиля (например, **6** для профиля search) или несколько номеров через пробел (например, **1 4 15** для профилей cdi.jvm, factor.jvm, system).

### Режим командной строки

Для запуска утилиты в режиме командной строки необходимо вызвать ее исполняемый файл, передав перечень названий интересующих профилей через пробел.

Пример (Linux):

```
$/diagnostic.sh db.connection
```

---

В данном примере утилита соберет информацию по профилю db.connection.

Пример (Windows):

```
diagnostic.bat cdi.jvm sql.counts versions
```

В данном примере утилита соберет информацию по профилям cdi.jvm, sql.counts и versions.

### *Параметризация профилей*

Некоторые профили (такие, как sql.party) предусматривают параметризацию выполняющихся команд. Значения параметров задаются в файле **CDI\_DIAG\_DIR/config/{имя\_профиля}.params.properties**.

Например, для профиля sql.party это файл **CDI\_DIAG\_DIR/config/sql.party.params.properties**:

```
hid = 1024
```

### *Результаты работы утилиты*

Информацию по собранным профилям утилита размещает в архиве

```
CDI_DIAG_DIR/diag/info.{date}_{time}.zip
```

Этот архив необходимо направить в службу поддержки HFLabs для дальнейшего анализа.

# ДИАГНОСТИЧЕСКИЕ ПРОФИЛИ

Задачи диагностики предоставляют возможность сбора следующей информации:

Профиль	Параметры	Какую информацию собирает
<b>logs</b>	<b>lowerBound</b> <b>upperBound</b>	Журналы ЕК и Фактора в диапазоне дат [ <b>lowerBound</b> , <b>upperBound</b> ]. Оба параметра опциональные, задаются в формате <b>гггг-мм-дд</b> .
<b>logs-full</b>	<b>lowerBound</b> <b>upperBound</b>	Полные журналы ЕК и Фактора в диапазоне дат [ <b>lowerBound</b> , <b>upperBound</b> ]. Оба параметра опциональные, задаются в формате <b>гггг-мм-дд</b> .
<b>sql.party</b>	<b>hid</b>	Информация по конкретному контрагенту или по группе контрагентов, указанных через ";"
<b>sql.party.old</b>	<b>hid</b>	Устаревший профиль сбора диагностики по КА. Работает аналогично <b>sql.party</b> , но использует другой механизм.
<b>mapdb-state</b>	<b>hid</b>	Состояние mapdb-индекса по контрагенту. Обычно собирается в паре с <b>sql.party</b>
<b>graph-state</b>	<b>hid</b>	Состояние графа связей по контрагенту.
<b>cdi.jvm</b>		Информация о JVM Единого клиента: дампы потоков, параметры обработки HTTP-запросов, использование ресурсов ОС
<b>config</b>		Конфигурационные файлы
<b>db.connection</b>		Информация о соединении с БД
<b>factor.jvm</b>		Информация о JVM Фактора: дампы потоков, параметры обработки HTTP-запросов, использование ресурсов ОС.
<b>search</b>		Статистика по поисковому индексу
<b>sql.common</b>		Системные таблицы
<b>sql.counts</b>		Количество контрагентов
<b>sql.db</b>		Физические характеристики БД (нужны права <b>GRANT SELECT_CATALOG_ROLE TO &lt;user&gt;;</b> )
<b>sql.duplicate</b>		Общая статистика по дубликатам
<b>sql.duplicate.physical</b>		Статистика по дубликатам (физические лица)
<b>sql.metadata</b>		Метаданные БД
<b>sql.quality.physical</b>		Статистика по качеству данных (физические лица)
<b>sql.quality.phone</b>		Статистика по качеству данных (телефоны)
<b>sql.data.statistics.previous.month</b>		Статистика по качеству данных за прошлый месяц (общая)
<b>versions</b>		Версии ЕК, Фактора и справочников Фактора

<b>sql.checker</b>		Проверка целостности и согласованности бизнес-объектов системы
<b>sql.stat.task</b>	<b>task=task_key</b>	Информация обо всех запусках указанной задачи. <b>task_key</b> - ключ задачи как в админке, например, <b>findDuplicates</b>
<b>sql.buffer</b>		Сбор диагностики с таблицей <b>cdi_buffer_increments</b> . Выгружается вся таблица.
<b>user.env</b>		Информации о браузере
<b>user.event</b>	<b>lowerBound</b>  <b>upperBound</b>	Информации о событиях в веб-интерфейсе в диапазоне дат [ <b>lowerBound, upperBound</b> ]. Оба параметра опциональные, задаются в формате <b>гггг-мм-дд</b> .
<b>metadata</b>		Метаинформация о модели данных для автоматической генерации документации



# НАСТРОЙКА УТИЛИТЫ ПОД КОНКРЕТНОГО ЗАКАЗЧИКА

Не у всех Заказчиков есть юридические лица и ИП. Поэтому, перед тем, как отдавать утилиту Заказчику, необходимо убрать оттуда все лишнее:

## 1. `sql.counts.properties`

Удаляем лишние селекты, например, ИП:

```
# Количество исходных контрагентов (ИП) с разбиением по системам
COUNT_INDIVIDUAL_SRC = select substr(merge_author, 0, instr(merge_author, ':')-1)
author, count(*) party_count \
from individual_party \
where version = 0 and enddate is null and merged_status in (0,1) \
group by substr(merge_author, 0, instr(merge_author, ':')-1)

# Количество результирующих контрагентов (ИП)
COUNT_INDIVIDUAL = select count(*) party_count \
from individual_party \
where version = 0 and enddate is null and merged_status in (0,2)
```

## 2. `diagnostic-cdi-search.xml`

Тут выкидываем бины ненужные

```
<bean id="search.count.individual" parent="abstractInfoFileGenerator">
  <property name="infoProvider">
    <bean class="ru.hflabs.cdi.diagnostic.other.LuceneDocumentsCountProvider">
      <property name="indexDirectory">
        <bean class="java.io.File">
          <constructor-arg type="java.io.File" ref="cdi.searchFolder"/>
          <constructor-arg type="java.lang.String" value="IndividualParty"/>
        </bean>
      </property>
    </bean>
  </property>
</bean>
```

## 3. `diagnostic-cdi-search.xml`

Тут же выкидываем эти бины из начала файла:

```
<!-- Профиль: Поисковый индекс -->
<bean id="search" class="ru.hflabs.cdi.diagnostic.DiagnosticProfile" lazy-init="false">
  <constructor-arg>
    <util:list>
      <idref bean="search.dir"/>
      <idref bean="search.count.physical"/>
      <idref bean="search.count.legal"/>
      <idref bean="search.count.individual"/>
      <idref bean="search.count.lender"/>
    </util:list>
  </constructor-arg>
  <property name="description" value="Search engine information"/>
</bean>
```

# СБОР ДИАГНОСТИЧЕСКОЙ ИНФОРМАЦИИ

- Информация по контрагенту
- Сбор логов
- Сбор версий (ЕК, Фактор и справочники - КЛАДР, ЕГРЮЛ и другие)
- Диагностика поискового индекса
- Характеристики сервера приложения (ThreadDump, например)
- Диагностика по задаче
- Характеристики БД
- Сбор диагностики по дубликатам

## *Информация по контрагенту*

В файле **CDI\_DIAG\_DIR\config\sql.party.params.properties** вместо ? необходимо подставить идентификатор контрагента, который можно скопировать с карточки контрагента.

```
# HID искомого контрагента  
hid = ?
```

Также можно собрать информацию сразу по нескольким контрагентам, указав их hid-ы через запятую. После запятой могут быть пробелы, а могут и не быть.

```
# HID искомого контрагента  
hid = 1024, 1025, 1026
```

После этого необходимо собрать информацию по контрагенту с профилем **sql.party** из директории

## *Сбор логов*

В задаче *Сбор логов* необходимо установить значения параметров `lowerBound` (с какой даты собираем логи?) и `upperBound` (по какую дату?)

Пример. Сбор логов с 11 по 18 марта 2013 года

```
lowerBound = 2013-03-11  
upperBound = 2013-03-18
```

После этого необходимо собрать диагностическую информацию с профилем **logs** (из директории **CDI\_DIAG\_DIR**)

## Сбор версий (ЕК, Фактор и справочники - КЛАДР, ЕГРЮЛ и другие)

Для проверки версий справочников необходимо собрать диагностическую информацию с профилем **versions** (из директории **CDI\_DIAG\_DIR**)

### Диагностика поискового индекса

Для анализа поискового индекса необходимо собрать диагностическую информацию с профилем **search**. **Результаты см в файлах:**

- **search.count.physical.txt** - количество "золотых" клиентов ФЛ (аналогичные файлы есть и для других типов контрагентов)
- **search.dir.txt** - размер поискового индекса

### Характеристики сервера приложения (ThreadDump, например)

Для сбора характеристик сервера приложения (Параметры JVM и параметры JBOSS) необходимо запустить диагностику с профилями **cdi.jvm** и **factor.jvm** (из директории **CDI\_DIAG\_DIR**)

Информация находится в следующих файлах:

- Параметры JVM CDI - **cdi.jvm.runtime.txt**
- Параметры JBOSS CDI - **cdi.jvm.http.txt**
- Параметры JVM ФАКТОР - **factor.jvm.runtime.txt**
- Параметры JBOSS ФАКТОР - **factor.jvm.http.txt**

### Диагностика по задаче

Для анализа поискового индекса необходимо собрать диагностическую информацию, указав:

Название параметра	Значение	Комментарий
profiles	sql.stat.task	
params	task= <b>task_key</b> <b>task_key</b>	ключ задачи как в админке, например, <b>findDuplicates</b>

Например, для задачи поиска дублей

**Создать/Редактировать задачу**

Группа diagnostic  
 Название diagnosticProfiles

Параметры задачи

Название параметра	Значение	Обязательность
runInCluster	true	
profiles	sql.stat.task	
params	task=findDuplicates	

Сохранить  Отмена

## Характеристики БД

Для анализа характеристик БД необходимо собрать диагностическую информацию с профилем **sql.db** (из директории **CDI\_DIAG\_DIR**)

Пример (Windows):

```
diagnostic.bat sql.db
```

Пример (Linux):

```
$/diagnostic.sh sql.db
```

Результаты диагностики находятся в **CDI\_DIAG\_DIR/\_diag/info.{date}\_{time}.zip**

## Сбор диагностики по дубликатам

Для проверки версий справочников необходимо собрать диагностическую информацию с профилем **sql.duplicate** (из директории **CDI\_DIAG\_DIR**)

Пример (Windows):

```
diagnostic.bat sql.duplicate
```

Пример (Linux):

```
$/diagnostic.sh sql.duplicate
```

Результаты диагностики находятся в **CDI\_DIAG\_DIR/\_diag/info.{date}\_{time}.zip**

# READ-ONLY ПОЛЬЗОВАТЕЛЬ СУБД ДЛЯ ДИАГНОСТИКИ

Для расследования инцидентов, возникающих при работе системы в производственном окружении, рекомендуется использовать отдельного пользователя СУБД.

## Права

- **CREATE SESSION**
- Роль **SELECT\_CATALOG\_ROLE**
- **SELECT** из таблиц:

```
ADDRESS  
CONSENT  
DOCUMENT  
DUPLICATE  
EMAIL  
HISTORY  
INDIVIDUAL_PARTY  
LEGAL_PARTY  
LENDER_PARTY  
LOGGING  
MAPPING  
MERGED  
MERGED_ATTRIBUTE  
PHONE  
PHYSICAL_PARTY  
REFERENCE  
RELATION  
RELATION_TYPE  
FLYWAY_SCHEMA_HISTORY  
STAGING  
SYSTEM
```

- **SELECT** из прикладных view (перечень зависит от конфигурации заказчика)
- **EXECUTE** на прикладные пакеты и функции (перечень зависит от конфигурации заказчика)

## Синонимы

Для данного пользователя необходимо создать синонимы для всех таблиц, на которые у него есть SELECT-привилегии.

## Пример

Пример скрипта для создания пользователя см. в файле **schema\_oracle\_create\_user\_diag.sql** (скрипт должен выполняться под учетной записью с правами system).

# ДИАГНОСТИЧЕСКИЕ SQL-ЗАПРОСЫ

- Информация по контрагенту (без атрибутов)
- Информация по данным контрагента из буферной таблицы
- Информация по атрибутам конкретного контрагента
  - Адреса
  - Телефоны
  - Email
  - Документы (ДУЛ)
  - Сведения по слиянию
- Информация по атрибутам контрагентов
  - Хорошие адреса, не требующие проверки человеком
  - Адреса, требующие проверки человека
- Информация по дубликатам
  - Дубликаты контрагента
  - Дубликаты контрагента (полная информация)
  - Количество негарантированных пар дубликатов с разбивкой по сценариям
  - Количество дубликатов, слитых в определенных интервал времени, в разбивке по правилам слияния
- Метаданные
  - Объекты БД
  - Перечень индексов
  - Структура индексов
  - Перечень ограничений
- Физические характеристики БД
  - Размер БД
  - 20 наиболее крупных объектов БД
  - Размеры табличных пространств
  - Файлы данных
  - Redo логи

## ***Информация по контрагенту (без атрибутов)***

Вместо ? необходимо подставить идентификатор контрагента.

```
-- контрагент
select * from physical_party where hid_party = ?
union all
select * from physical_party where hid_party in (select hid_party from merged where
```

```

hid_final_scion = ?)
order by 2, 1;

-- сведения по слиянию
select * from merged where hid_party = ? or hid_direct_scion = ? or hid_final_scion = ?
order by id_merged;

-- история изменения
select id_history, history_type, history_party_event, history_event, hid_object, hid_party,
created, author
from history where hid_party = ?
union all
select id_history, history_type, history_party_event, history_event, hid_object, hid_party,
created, author
from history where hid_party in (select hid_party from merged where hid_final_scion = ?)
order by hid_party, id_history;

-- HID золотой карточки
select hid_final_scion from merged where hid_party = ? and canceled is null and transitional
= 0;

```

## *Информация по данным контрагента из буферной таблицы*

Вместо ? необходимо подставить значение **record\_id**

```

-- все контрагенты с диапазоне
select * from cdi_buffer where record_id > ?

and record_id < ?

```

## *Информация по атрибутам конкретного контрагента*

Вместо ? необходимо подставить идентификатор контрагента.

### *Адреса*

```

-- адреса
select * from address where hid_party = ?
union all
select * from address where hid_party in (select hid_party from merged where
hid_final_scion = ?)
order by 6, 2, 1;

```

### *Телефоны*

```

-- телефоны
select * from phone where hid_party = ?
union all
select * from phone where hid_party in (select hid_party from merged where hid_final_scion
= ?)
order by 6, 2, 1;

```

## Email

```
-- email
select * from email where hid_party = ?
union all
select * from email where hid_party in (select hid_party from merged where hid_final_scion =
?)
order by 6, 2, 1;
```

## Документы (ДУЛ)

```
-- документы (ДУЛ)
select * from document where hid_party = ?
union all
select * from document where hid_party in (select hid_party from merged where
hid_final_scion = ?)
order by 6, 2, 1;
```

## Сведения по слиянию

```
-- адреса
select * from merged_attribute
where final_hid = (select final_hid from merged_attribute
where attribute_type = 'ADDRESS'
and original_hid = ?);

-- телефоны
select * from merged_attribute
where final_hid = (select final_hid from merged_attribute
where attribute_type = 'PHONE'
and original_hid = ?);

-- email
select * from merged_attribute
where final_hid = (select final_hid from merged_attribute
where attribute_type = 'EMAIL'
and original_hid = ?);

-- документы (ДУЛ)
select * from merged_attribute
where final_hid = (select final_hid from merged_attribute
where attribute_type = 'DOCUMENT_PASSPORT'
and original_hid = ?);
```

## Информация по атрибутам контрагентов

Рандомная выборка по всей базе, снимает семпл (пример) на определенный % от всего объема.

Вместо ? необходимо подставить нужный процент выборки.

### *Хорошие адреса, не требующие проверки человеком*

```
-- 10% хороших адресов
select raw_source, qualitycode, validationcode, merge_author from address sample(?)
```



```
where qualitycode in ('GOOD', 'UNDEF_05')
and validationcode in ('VALIDATED', 'CONFIRMED_MANUALLY')
and version = 0 and enddate is null;
```

### *Адреса, требующие проверки человека*

```
-- 10% плохих адресов
select raw_source, qualitycode, validationcode, merge_author from address sample(?)
where (qualitycode not in ('GOOD', 'UNDEF_05')
or validationcode not in ('VALIDATED', 'CONFIRMED_MANUALLY'))
and version = 0 and enddate is null;
```

## *Информация по дубликатам*

### *Дубликаты контрагента*

```
select hid_party_2 from duplicate WHERE hid_party_1 = ?
and version = 0
and enddate is null
union
select hid_party_1 from duplicate WHERE hid_party_2 = ?
and version = 0
and enddate is null;
```

### *Дубликаты контрагента (полная информация)*

```
select * from duplicate WHERE hid_party_1 = ?
union
select * from duplicate WHERE hid_party_2 = ?
order by 7, 2, 1
```

### *Количество негарантированных пар дубликатов с разбивкой по сценариям*

```
select match_scope as "Козф. соответствия", rule as "ID сценария", count(*) as "Кол-во пар"
from duplicate
where version = 0 and merged_status = 0
group by match_scope, rule
order by match_scope desc, rule;
```

### *Количество дубликатов, слитых в определенных интервал времени, в разбивке по правилам слияния*

Данный пример показывает все слияния, начиная от даты 14.01.2013 и выше.

Чтобы изменить дату, надо в 2 местах поменять условие **trunc(startdate) >= '20.10.2013'**

Чтобы установить интервал дат, надо заменить условие на условие вида **trunc(startdate) >= '20.10.2013' and trunc(startdate) <= '20.10.2013'**

```
select distinct trunc(startdate), rule, count(*)
from duplicate
where trunc(startdate) >= '20.10.13'
and version = 0
and enddate is null
and merged_status = 1
group by trunc(startdate), rule;
```

## Метаданные

### Объекты БД

```
select object_type, object_name from user_objects order by object_type, object_name;
```

### Перечень индексов

```
select table_name, index_name, tablespace_name, index_type, partitioned, logging,
compression, uniqueness, status
from user_indexes order by table_name, index_name;
```

### Структура индексов

```
select i.table_name, i.index_name, ic.column_position, ic.column_name, ie.
column_expression
from user_indexes i join user_ind_columns ic on i.index_name = ic.index_name
left join user_ind_expressions ie on ic.index_name = ie.index_name and ic.column_position
= ie.column_position
order by i.table_name, i.index_name, ic.column_position;
```

### Перечень ограничений

```
select table_name, constraint_type, constraint_name, status, r_constraint_name,
index_name
from user_constraints
where constraint_type in ('P', 'R', 'U')
order by table_name, constraint_type, constraint_name;
```

## Физические характеристики БД

### Размер БД

```
select
round(sum(used.bytes)/1024/1024/1024) || ' GB' "Database Size",
round(sum(used.bytes)/1024/1024/1024) - round(free.p/1024/1024/1024) || ' GB' "Used space",
round(free.p/1024/1024/1024) || ' GB' "Free space"
from
(select bytes from v$datafile
union all
select bytes from v$tempfile
union all
select bytes from v$log
```

```
) used,  
(select sum(bytes) as p from dba_free_space) free  
group by free.p;
```

## 20 наиболее крупных объектов БД

```
select owner, segment_name, segment_type, mb  
from  
(select owner, segment_name, segment_type, bytes/1024/1024 "MB"  
from dba_segments  
order by bytes desc  
)  
where rownum < 21;
```

## Размеры табличных пространств

```
select tsu.tablespace_name,  
ceil(tsu.used_mb) "size MB",  
decode(ceil(tsf.free_mb), NULL, 0, ceil(tsf.free_mb)) "free MB",  
decode(100 - ceil(tsf.free_mb/tsu.used_mb*100), NULL, 100, 100 - ceil(tsf.free_mb/tsu.  
used_mb*100)) "% used"  
from  
(select tablespace_name, sum(bytes)/1024/1024 used_mb  
from dba_data_files group by tablespace_name  
union all  
select tablespace_name || ' **TEMP**', sum(bytes)/1024/1024 used_mb  
from dba_temp_files group by tablespace_name  
) tsu,  
(select tablespace_name, sum(bytes)/1024/1024 free_mb  
from dba_free_space group by tablespace_name  
) tsf  
where tsu.tablespace_name = tsf.tablespace_name (+)  
order by 1;
```

## Файлы данных

```
select tablespace_name, file_name, ceil(bytes/1024/1024) "size MB", ceil(maxbytes/1024  
/1024) "max size MB"  
from dba_data_files  
order by tablespace_name, file_name;
```

## Redo логи

```
select lf.member "Redo log", ceil(lg.bytes / 1024 / 1024) "Size MB"  
from v$logfile lf, v$log lg  
where lg.group# = lf.group#  
order by 1;
```

# ДИАГНОСТИЧЕСКИЕ SQL-ЗАПРОСЫ (ПОИСК ПРОБЛЕМ)

- *Старая версия атрибута лучше, чем новая*
- *В золотой карточке не слились одинаковые телефоны*
- *1 исходный КА влился в 2 результирующих*
- *Есть неуникальные данные в индексе в МА (original\_hid, attribute\_type, canceled)*

## Старая версия атрибута лучше, чем новая

В данном примере мы ищем все адреса, у которых

- Старая версия - хорошая
- Новая версия - плохая, но у нее изменилось 1 поле (district)

```
insert /*+ APPEND */ into TMP$PROBLEMS (hid_party, problem_type, hid_attr)
SELECT /*+ ORDERED FULL(angel) FULL(ma) */ a4.hid_party, 'UPDATE_DISTRICT', a4.
hid_address FROM
(SELECT * FROM
(
SELECT a2.*,
row_number() over (partition by a2.hid_address order by a2.startdate desc) as
address_number
FROM
address a1 join address a2 on (a2.hid_address = a1.hid_address and a2.startdate < a1.
startdate)
WHERE a1.district is not null and
(a1.validationcode not in ('VALIDATED', 'CONFIRMED_MANUALLY') or a1.qualitycode not in
('GOOD', 'UNDEF_05', 'UNDEF_07'))
) a
WHERE a.address_number = 1 and a.district is null and
(a.validationcode in ('VALIDATED', 'CONFIRMED_MANUALLY') and a.qualitycode in ('GOOD',
'UNDEF_05', 'UNDEF_07')) a3
join address a4 on (a3.hid_address = a4.hid_address)
where a4.version = 0 and a4.validationcode not in ('VALIDATED', 'CONFIRMED_MANUALLY');
```

## В золотой карточке не слились одинаковые телефоны

В данном примере мы ищем все телефоны, которые идентичны по значимым полям, находятся в "золотой" карточке, но не слиты (hidden = 0).

Использовалось, чтобы найти телефоны, которые не слились из-за разного поля *Оператор*.

```
insert /*+ APPEND */ into TMP$PROBLEMS (hid_party, problem_type, hid_attr)
select /*+ ORDERED FULL(p) FULL(pp) */ p.hid_party, 'MERGE_OPERATOR', p.
```

```

phone_type||'_'||p.countrycode||p.citycode||p.telephone from phone p
join physical_party pp on p.hid_party = pp.hid_party
where pp.version = 0 and pp.enddate is null and pp.merged_status = 2
and p.version = 0 and p.enddate is null and p.hidden = 0
group by p.hid_party, p.phone_type||'_'||p.countrycode||p.citycode||p.telephone
having count(*) > 1;

```

## 1 исходный КА влился в 2 результирующих


Если **check\_8** находит проблемы, дальше надо диагностировать - 1 КА влит в 2 результирующих или больше?

Если только в два, то можно далее выполнить этот скрипт.

Скрипт для того, чтобы правильно выбрать исходные записи, которые должны быть удалены. Основной принцип: в одной группе с проблемой есть две результирующих записи. Количество исходных у них не равно, при этом исходные одной полностью входят в исходные другой (если это не так, т.е. когда неполное пересечение по исходным, то всё сильно сложнее). Удалять надо ту, у которой список исходных меньше. Тогда все записи останутся влитыми в записи с БОльшими списками, ничего не потеряется.

Проблема в том, что нет чёткой зависимости типа "больше хид - больше список".

Если списки исходных совпадают (идеальный случай), удалять можно любую из результирующих.

 Скрипт ниже работает только для тех случаев, когда максимальное количество результатов слияния равно двум. Если в БД есть БОльшее количество неоднозначных записей, скрипт выполнять строго нельзя!

```

with dist as (
  -- выбрать все варианты вхождения
  select distinct min_final, max_final
  from (
    -- для каждой исходной от проблемных выяснить минимальный и максимальный
    хиды результирующих
    select hid_party, min(hid_final_scion) min_final, max(hid_final_scion) max_final
    from merged
    where canceled is null and hid_party in (
      -- выбрать исходные для неоднозначных результатов
      select hid_party
      from merged
      where canceled is null and hid_final_scion in (
        -- выбрать все результирующие по проблемным исходным (т.е. все неоднозначные
        результаты)
        select distinct hid_final_scion
        from merged
        where canceled is null and hid_party in (
          -- выбрать все исходные, для которых больше одного результирующего
          select hid_party source_hid
          from merged
          where canceled is null group by hid_party having count(*) = 2
        )))
    group by hid_party
  ),
  dist_dif as (

```

```

-- вхождение в две разных результирующих записи
select * from dist where min_final != max_final
),
dist_same as (
-- вхождение только в одну результирующую запись
select min_final same_final from dist where min_final = max_final
)
select
-- отладочные поля, при выборке только хидов на удаление закомментировать
d1.min_final, d1.max_final, d2.same_final same_as_min, d3.same_final same_as_max,
-- выбор той, которую удалить.
-- если есть уникальные вхождения только в одну из проблемных записей (т.е. по
включению исходных), вернётся другая запись на удаление
-- если есть уникальные вхождения в обе проблемных записи (т.е. неполное
пересечение исходных), вернётся null
-- если уникальных вхождений нет ни в одну из записей (т.е. полное пересечение
исходных), вернётся результирующая запись с максимальным hid
nvl(
  nvl2(d2.same_final, d1.max_final, null),
  nvl2(d3.same_final, d1.min_final, null)
) to_del
from
  dist_dif d1
  left join dist_same d2 on (d1.min_final = d2.same_final)
  left join dist_same d3 on (d1.max_final = d3.same_final)
;

```

## *Есть неуникальные данные в индексе в МА (original\_hid, attribute\_type, canceled)*

Как найти эти атрибуты:

```

select original_hid, attribute_type, canceled, count(original_hid) from
merged_attribute
group by original_hid, attribute_type, canceled
having count(original_hid) > 1;

```

Как найти hid\_party с проблемами, чтобы диагностику собрать:

```

with ma as (
select original_hid, attribute_type, canceled, count(original_hid) from merged_attribute
group by original_hid, attribute_type, canceled
having count(original_hid) > 1)
(
select hid_party from document where hid_document in (select original_hid from ma where
attribute_type = 'DOCUMENT_PASSPORT')
) union (
select hid_party from document where hid_document in (select original_hid from ma where
attribute_type = 'ADDRESS')
) union (
select hid_party from document where hid_document in (select original_hid from ma where
attribute_type = 'EMAIL')
) union (
select hid_party from document where hid_document in (select original_hid from ma where
attribute_type = 'PHONE')
)
;

```



# ТРАССИРОВКА НА УРОВНЕ СУБД

При диагностике проблем производительности может потребоваться трассировка действий, выполняемых на уровне СУБД.

Порядок действий при сборе трассировки:

1. Убедиться, что у пользователя **sys** создан пакет **dbms\_monitor**.  
Если он отсутствует, то можно создать с помощью файла **{ORACLE\_HOME}/rdbms/admin/dbmsmntr.sql**
2. Убедиться, что у пользователя СУБД есть права на execute пакета **dbms\_monitor**.
3. Запустить операцию ЕК, которую необходимо профилировать.
4. Запустить трассировку для всех сессий ЕК:

```
exec sys.dbms_monitor.serv_mod_act_trace_enable('SYS$USERS', 'HFLabs-CDI');
```

Вместо **HFLabs-CDI** необходимо подставить значение свойства **v\$session.program** в настройках соединения с БД.

5. После окончания выполнения профилируемой операции ЕК, остановить трассировку:

```
exec sys.dbms_monitor.serv_mod_act_trace_disable('SYS$USERS', 'HFLabs-CDI');
```

Вместо **HFLabs-CDI** необходимо подставить значение свойства **v\$session.program** в настройках соединения с БД.

6. Собрать агрегированную статистику с помощью утилиты **tkprof**:

```
tkprof <путь_к_файлу_трассировки> <результатирующий_файл>
```

Каталог с файлами трассировки находится на сервере СУБД. Путь к нему хранится в параметре СУБД **user\_dump\_dest**:

```
select value from v$parameter where name = 'user_dump_dest';
```

## Пример - трассировка во время поиска дубликатов

Убедились, что права у пользователя есть.

1. Определяем значение свойства **v\$session.program**. На Linux выполняем запрос:

```
grep "session.program" /home/cdi/jboss/standalone/deployments/cdi-oracle-ds.xml
```

Получаем ответ:

```
<connection-property name="v$session.program">CDIDatasource</connection-property>
```



2. Запускаем задачу *Поиск дубликатов* в АРМ Администратора.
3. Запускаем трассировку для всех сессий ЕК:

```
exec sys.dbms_monitor.serv_mod_act_trace_enable('SYS$USERS', 'CDIDatasource');
```

4. После окончания выполнения поиска дубликатов (или через час, если задача выполняется слишком долго), остановить трассировку:

```
exec sys.dbms_monitor.serv_mod_act_trace_disable('SYS$USERS', 'CDIDatasource');
```

5. Узнаем путь к трассировочному файлу

```
select value from v$parameter where name = 'user_dump_dest';  
-- Например, получаем  
/xdata/11g/diag/rdbms/cdi/cdi/trace  
  
-- Переходим в данный каталог и выполняем  
ls -ltr  
  
-- Допустим, получили последний файл  
cdi_ora_16661.trc
```

6. Заходим на сервер БД и выполняем следующую команду

```
tkprof /xdata/11g/diag/rdbms/cdi/cdi/trace/cdi_ora_16661.trc trace.prf;
```

# LUCENE. МЕДЛЕННО ОТКРЫВАЮТСЯ СТРАНИЦЫ В UI

Если резко упала скорость отображения данных (отрисовка списка контрагентов в UI долго открывается, отрисовка отдельно взятой карточки, скролирование и т.д.), необходимо диагностировать проблему.

Для этого:

1. Подключаем логирование

- War файл - **WEB-INF\classes\logback.xml (но подхватится только после рестарта)**
- Без рестарта - **JBOSS\_CDI\standalone\tmp\vfs\tmp\*\*\* (ищем самую свежую папку temp)\cdi-web-customer-version-SNAPSHOT.war-\*\*\*\WEB-INF\classes\logback.xml**

```
<logger name="ru.hflabs.cdi.engine.search" level="DEBUG"/>
```

Теперь в **cdi.log** будет логироваться время поисковых запросов.

2. Выполнить медленное действие в UI, при запуске приложения на клиенте.

3. Выполнить медленное действие в UI, при запуске приложения на сервере (если есть такая возможность).

4. Передать сформированный cdi.log команде сопровождения CDI.

# ПРОБЛЕМЫ С ЖЕСТКИМ ДИСКОМ НА WINDOWS — СБОР ИНФОРМАЦИИ

Выгрузка из индекса или отображение списка контрагентов в CDM начали спонтанно занимать очень много времени. При этом сжатие [HTTP](#) трафика включено.

Проблема может быть связана с софтом, может с железом. По предварительным данным тормозит чтение с диска (анализ Thread dump показывает, что время уходит на операции чтения из поискового индекса Lucene).

Сервер перегружать нежелательно (например, это прод-сервер на территории Заказчика)

Информация, которую можно получить без перезагрузки, и которая может помочь для выявления источника проблемы:

1. Дата последнего перестроения поискового индекса и дата последней оптимизации с указанием времени, которое заняла эта операция.
2. Uptime для службы CDI
3. Результаты SMART-диагностики дисков
4. Логи Windows - если что-то нехорошее творится с самим диском, это можно увидеть по логам.
5. Посмотреть uptime windows. Это можно сделать командой

```
net statistics server
```

Смотреть дату "Статистика после ..."

6. Собрать информацию о скорости записи/чтения на диск с помощью [iozone](#). Необходимо обратить внимание, что тестовый файл должен быть больше размера оперативной памяти (если позволяет свободное пространство на диске), например, для машины с 24Гб оперативки надо использовать 30Гб файл для тестирования чтения-записи

```
iozone -i 0 -i 1 -i 2 -i 3 -g 30G -s 30G
```

# АВТОМАТИЧЕСКИЙ МОНИТОРИНГ ЛОГОВ

## Зачем нужен

Как только произошла ошибка в работе ЕК, автоматически сообщает о ней в техподдержку по почте.

## Как подключить

1. В АРМ Администратора зайдите на вкладку *Конфигурация*, раздел *Параметры отправки email*.
2. Включите мониторинг: **mail.logging.enabled = true**
3. Укажите, с какого e-mail адреса рассылать сообщения: **mail.from = cdi@ваша\_компания.ru**
4. Задайте параметры SMTP-сервера (уточните их у системного администратора):
  - a. **mail.host** = имя SMTP-сервера (*smtp.ваша\_компания.ru*);
  - b. **mail.port** = порт SMTP-сервера (обычно 25)
  - c. **mail.user** = логин для авторизации на SMTP-сервере;
  - d. **mail.password** = пароль для авторизации на SMTP-сервере;
5. Перечислите, кому слать сообщения об ошибках: **mail.logging.to = ваш\_адрес@hflabs.ru, ваш\_сотрудник@ваша\_компания.ru**  
*ваш\_адрес@hflabs.ru* узнайте в техподдержке.
6. Сохраните настройки.
7. Перезапустите сервис ЕК.

Убедитесь, что доступ от ЕК до SMTP-сервера открыт:

Источник запроса			Получатель запроса			Протокол	Назначение и описание информационного потока (какой тип данных передается)
Имя (DNS-имя)	IP-адрес(а)	Порт	Имя (DNS-имя)	IP-адрес(а)	Порт		
Сервер приложений ЕК	XXX.XXX.XXX.XXX	>=1024	SMTP-сервер	XXX.XXX.XXX.XXX	XX	smtp /smtps	Отправка почтовых оповещений с диагностикой и информацией об ошибках

# ДИАГНОСТИКА «ЗАВИСАНИЙ» JVM С ПОМОЩЬЮ JHICCUP

Если есть подозрения, что JVM из-за GC или по другим причинам «зависает» на какое-то время, то тут пригодится утилита `jHiccup`.

Вкратце, все что она делает, это регулярно засыпает на 1мс, перед тем как заснуть запоминает время, а после пробуждения смотрит, сколько она реально спала.

Данные сплёвывает в файлики в формате гистограмм.

Умеет подключаться к уже работающим процессам. Не влияет на производительность.

## Сбор данных

Сама утилита поставляется с sh-скриптом, так что для винды [прикладываю](#) отдельно jar-ник и пример команды, как его запустать:

```
"%JAVA_HOME%\bin\java" -cp "%JAVA_HOME%\lib\tools.jar";"C:\jHiccup\jHiccup.jar" org.jhiccup.HiccupMeterAttacher -j "C:\jHiccup\jHiccup.jar" -l C:\jHiccupOut\cdi.hlog -p 1234
```

где "-p 1234" указывает PID процесса, к которому подключиться (нужно через JVisualVM или Диспетчер Задач заранее его узнать)

Если все ОК, команда молча выполнится и JVM теперь начнет срыгивать данные в указанный "cdi.hlog" файл.

UPD: для аттача к процессу нужно чтобы команда была запущена от имени того юзера, который запустил и сам wildfly. Если это невозможно (wildfly запускается как служба от имени особого юзера без пароля), проще всего включить jHiccup, изменив standalone.conf.bat:

в самом начале файла нужно написать

```
:: JBOSS options for HF Labs products
set _JAVA_OPTIONS=-javaagent:D:\jhiccup\jHiccup.jar="-l D:\jhiccup\cdi.hlog"
....
```

## Анализ

Для анализа данных нужно их конвертировать а затем анализировать сторонними средствами.

Но на Github уже есть [удобная GUI-тулза](#) для визуализации. К сожалению автор в ней чуть-чуть накосячил, и она запускается только на английской локали. [Прикладываю](#) исправленный исполняемый JAR-файл.

# КАК СНЯТЬ ДИАГНОСТИКУ ВЫПОЛНЕНИЯ WEB-ЗАПРОСОВ (ACCESS.LOG)

Данная настройка помогает включить фиксацию всех запросов к приложению в access\_log.log

## Для включения полного логирования необходимо:

1. Зайти в директорию `{JBOSS_CDI}/bin`
2. Запустить консоль управления `./jboss-cli.sh --connect`
3. Ввести логин пароль `admin / cdi`
4. В открывшейся консоли выполнить команды:

```
connect
batch
/subsystem=undertow/server=default-server/https-listener=https:/:undefine-attribute
(name=enable-http2)
/subsystem=undertow/server=default-server/http-listener=default:/write-attribute
(name=record-request-start-time, value=true)
/subsystem=undertow/server=default-server/https-listener=https:/write-attribute
(name=record-request-start-time, value=true)
/subsystem=undertow/server=default-server/host=default-host/setting=access-log:add
(pattern="%%{time,yyyy-MM-dd}      %{time,HH:mm:ss} %h %T \"%r\" %s \"%{i,User-
Agent}\"")
run-batch
```

Можно выполнять команды как последовательно, так и сразу же указать все.

5. Передать вывод консоли команде сопровождения CDI.
6. После проверки, что на шаге 4 все выполнено успешно выйти из консоли управления. Перезапуск CDI не требуется.

После этого появится файл `{JBOSS_CDI}/standalone/log/access_log.log`, в котором вайлдфлай пишет все http-запросы с временем обработки на сервере (полное время включая обвязку Wildfly).

Чтобы стало заполняться время выполнения запроса, надо перезагрузить ЕК.

## После включения полного логирования web-запросов

1. Выполнить вход на страницы, которые обрабатывают медленно
2. Передать сформировавшийся `access_log.log` команде сопровождения CDI.

## Отключение логирования

1. Зайти в директорию `{${JBoss_CDI}/bin`
2. Запустить консоль управления `./jboss-cli.sh --connect`
3. В открывшейся консоли выполнить:

```
/subsystem=undertow/server=default-server/host=default-host/setting=access-log:remove
```

Перезапуск CDI не требуется.

### *Примечание*

В логе данные фиксируются согласно заданной маске `%t %h %T \"%r\" %s \"%i,User-Agent`, где:

`%t` – Время получения запроса (стандартный английский формат);

`%h` – Удалённый хост;

`%T` – Время, затраченное на обслуживание запроса в секундах;

`%r` – Первая строка запроса;

`%s` – Статус. Для запросов, которые получили внутреннюю переадресацию, это статус опционального запроса — `%>s` для последнего;

`%i` – Содержимое строки (строк) заголовка в отправленном на сервер запросе в параметре `User-Agent`.

# ПРОФИЛИРОВАНИЕ УДАЛЕННОГО СТЕНДА

- Подготовка к запуску
- Настройка утилиты JVVM
- Настройка и запуск в Linux
- Запуск снятия статистики JVVM
- Перерыв на кофе
- Снятие профилировки по CPU через JVVM
- Возможные проблемы и решения

Проблема 1. При попытке профилировки выдается сообщение No Data.

Проблема 2. В консоли, откуда запущена jvvm, при попытке профилировки появляется ошибка: OOM:JavaHeapSpace

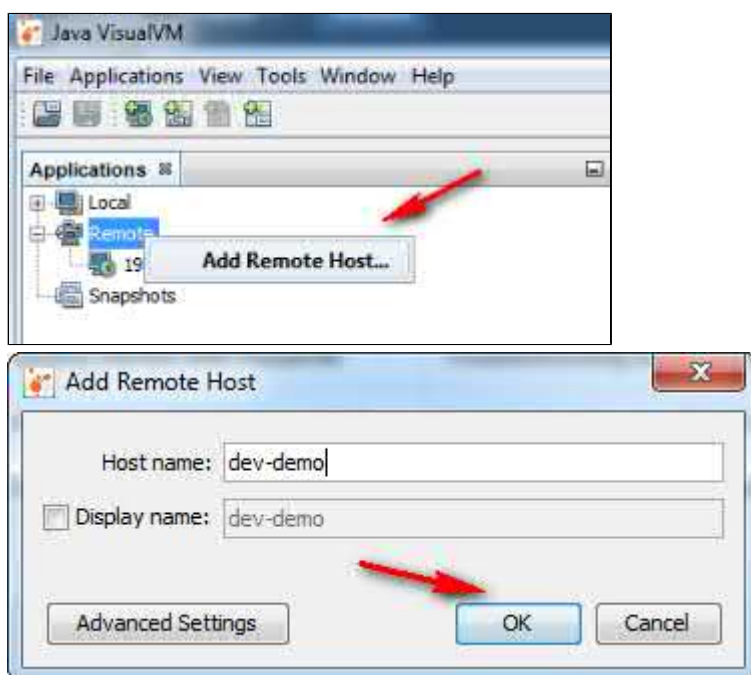
## *Подготовка к запуску*

1. Для сбора данных нужна рабочая станция, которая имеет доступ через порты 9990 и 19990 до сервера(ов) приложения, с которых планируется снимать профилировку.
2. Должен быть установлен OpenJdk11.  
Установить можно AdoptOpenJDK с сайта проекта (ссылка: <https://adoptopenjdk.net/archive.html?variant=openjdk11&jvmVariant=hotspot>).
3. Должна быть установлена последняя версия VisualVM (перейти по ссылке: <https://visualvm.github.io/?DownloadStandalone> ? скачать предложенный архив под вашу ОС). Полученный архив распаковать в любую директорию на вашем ПК.

## *Настройка утилиты JVVM*

1. Скачать и распаковать архив **jvvm\_16.zip**.
2. Отредактировать файл **jvvm16.bat**, указав в нем правильный путь до visualvm.exe и путь до OpenJdk.
3. Запустить утилиту двойным кликом по bat-нику **jvvm16.bat**
4. В запущенном JVVM сначала добавить удаленный хост:





5. Создать JMX-соединение для профилирования ЕК (CDI)



6. Задать параметры для профилирования ЕК (cdi):

- Connection-string — **service:jmx:http-remoting-jmx://<hostname>:<port>**, где  
**hostname** — имя или ip-адрес сервера cdi  
**port** — порт для профилировки — 9990 для cdi

Пример строки для подсоединения: **service:jmx:http-remoting-jmx://dev-demo:9990**

- Display name — понятное название создаваемого профиля, например, DEV-CDI
- Логин: admin
- Пароль: cdi

- Поставить галочку "Save security credentials"

7. Создать второе JMX-соединение для профилирования Фактора.

- Connection-string — **service:jmx:http-remoting-jmx://<hostname>:<port>**, где **hostname** — имя или ip-адрес сервера Фактора.  
**port** — порт для профилировки. Обычно 19990 для Фактора . (Если для Фактора в конфигурации jboss задано значение port.offset не 10000, а иное, то и порт для профилировки будет иной, а именно: port.offset + 9990).

Пример строки для подсоединения: **service:jmx:http-remoting-jmx://dev-demo:19990**

- Display name — понятное название создаваемого профиля, например, dev-factor
- Логин: admin
- Пароль: cdi
- Поставить галочку "Save security credentials"

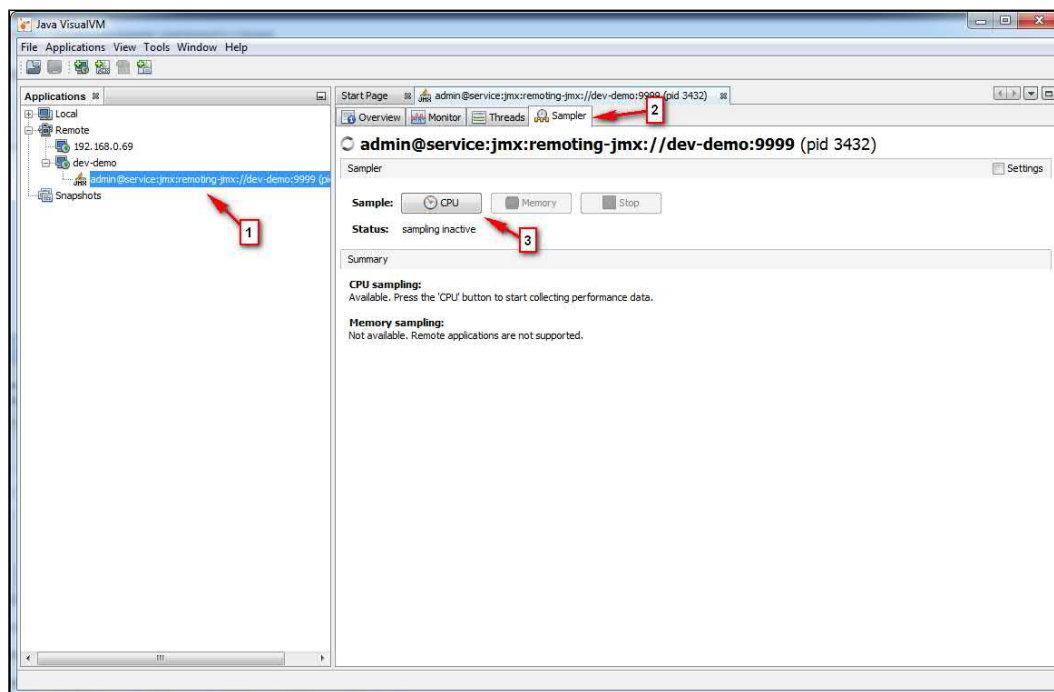
## ***Запуск снятия статистики JVVM***

**!** Если нужно снять профилировку не только с cdi, но и с factor, то повторить шаги из этого и следующих разделов для JMX-соединения к Фактору (порт 19990).

Основная статистика собирается по использованию процессора (CPU):

1. Открыть соединение, дважды щелкнув по нему в левой навигационной панели.

2. Открыть вкладку Sampler.
3. Нажать кнопку CPU.



## Перерыв на кофе

После старта профилировки надо 15 минут подождать пока накопится статистика.

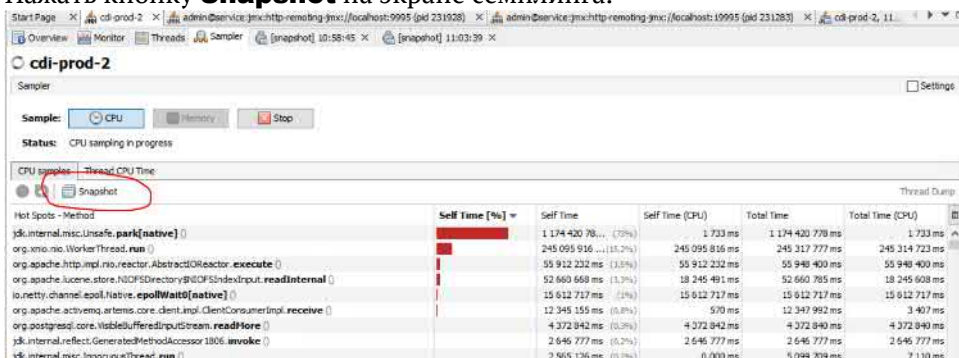


Далее снять первый снейпшот (см. следующий раздел).

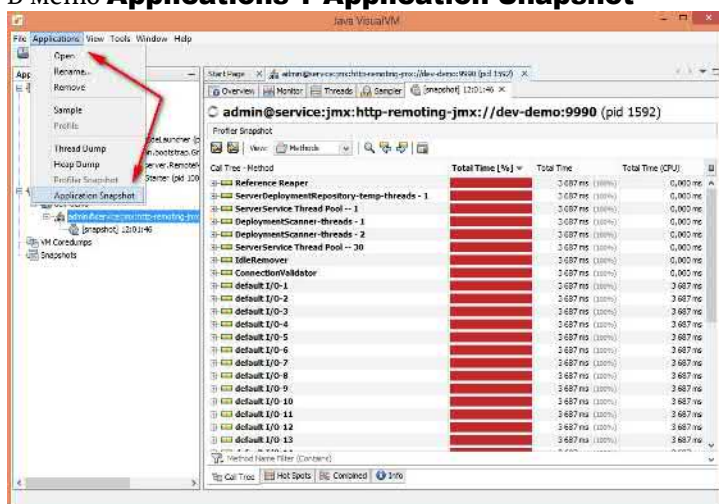
⚠️ Через 15-30 минут после первого снейпшота снять второй снейпшот.

## Снятие профилировки по CPU через JVVM

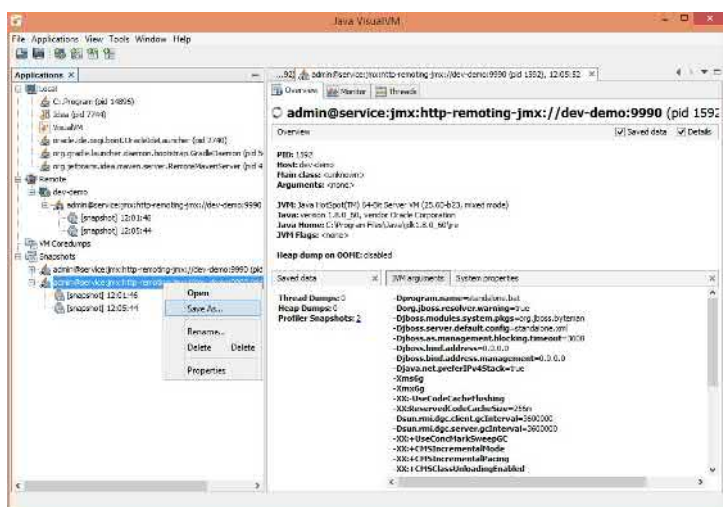
1. Когда накопилась статистика по использованию CPU (не ранее, чем через 15 минут после запуска) необходимо сохранить ее.
2. Нажать кнопку **Snapshot** на экране семплинга.



3. Создать **Application Snapshot**:
  - a. Выбрать в левом меню JMX-соединение к cdi или factor, который профилируем.
  - b. В меню **Applications ? Application Snapshot**



4. Созданный на шаге 6 снейпшот отобразится в левой навигационной панели:



5. Сохранить снимок: нажать на него правой кнопкой мыши и выбрать "Save As...". Сохранить обязательно в том формате, которое предлагает приложение.
6. Собранные файлы отправить коллегам HFLabs по почте или прикрепить к задаче в Jira

## ***Возможные проблемы и решения***

### **Проблема 1. При попытке профилировки выдается сообщение No Data.**

Проверить, что

1. на клиенте, откуда запущена jvisualvm, версия java >= версии java сервера, который хотим профилировать
2. jboss-client.jar, с которым ты запускаешь jvisualvm, от правильного jboss-a, т.е. от той же версии JBOSS, что и на сервере, который профилируем.  
При расхождении версий установить правильную джаву и jboss-client.jar, после чего повторить попытку.

### **Проблема 2. В консоли, откуда запущена jvmt, при попытке профилировки появляется ошибка: OOM:JavaHeapSpace**

Увеличить параметр `-J-Xmx` в файле `visualvm.conf`, который находится в `<JAVA_HOME>\lib\visualvm\etc`.

Обычно по умолчанию там 256Мб, рекомендуется поставить 2Гб:

`-J-Xmx2g`

# НАСТРОЙКА УТИЛИТЫ JVVM

## Подготовка к запуску

1. Для сбора данных нужна рабочая станция, которая имеет доступ через порты 9990 и 19990 до сервера(ов) приложения, с которых планируется снимать профилировку.
2. Должен быть установлен OpenJdk11.  
Установить можно AdoptOpenJDK с сайта проекта (ссылка: <https://adoptopenjdk.net/archive.html?variant=openjdk11&jvmVariant=hotspot>).
3. Должна быть установлена последняя версия VisualVM (перейти по ссылке: <https://visualvm.github.io/?DownloadStandalone> ? скачать предложенный архив под вашу ОС). Полученный архив распаковать в любую директорию на вашем ПК.

## Настройка утилиты JVVM

1. Скачать и распаковать архив **jvvm\_16.zip**.
2. Отредактировать файл **jvvm16.bat**, указав в нем правильный путь до **visualvm.exe** и путь до OpenJdk.
3. Запустить утилиту двойным кликом по bat-нику **jvvm16.bat**
4. В запущенном JVVM сначала добавить удаленный хост:



5. Создать JMX-соединение для профилирования ЕК (CDI)

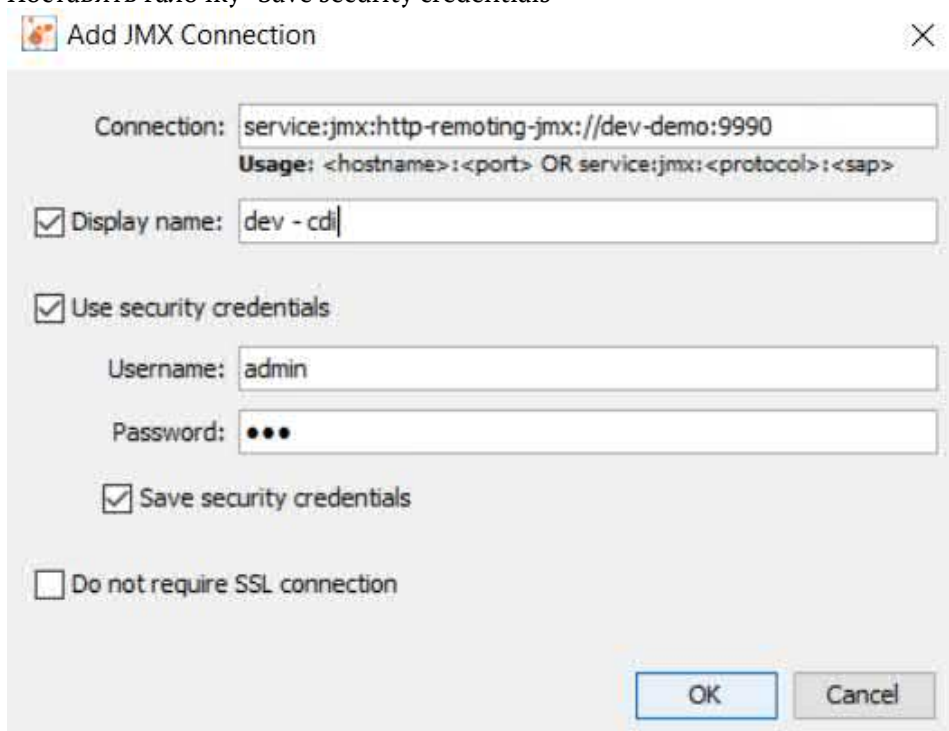


6. Задать параметры для профилирования ЕК (cdi):

- Connection-string — **service:jmx:http-remoting-jmx://<hostname>:<port>**, где  
**hostname** — имя или ip-адрес сервера cdi  
**port** — порт для профилировки — 9990 для cdi

Пример строки для подключения: **service:jmx:http-remoting-jmx://dev-demo:9990**

- Display name — понятное название создаваемого профиля, например, DEV-CDI
- Логин: admin
- Пароль: cdi
- Поставить галочку "Save security credentials"



## 7. Создать второе JMX-соединение для профилирования Фактора.

- Connection-string — **service:jmx:http-remoting-jmx://<hostname>:<port>**, где **hostname** — имя или ip-адрес сервера Фактора.  
**port** — порт для профилировки. Обычно 19990 для Фактора . (Если для Фактора в конфигурации jboss задано значение port.offset не 10000, а иное, то и порт для профилировки будет иной, а именно: port.offset + 9990).

Пример строки для подсоединения: **service:jmx:http-remoting-jmx://dev-demo:19990**

- Display name — понятное название создаваемого профиля, например, dev-factor
- Логин: admin
- Пароль: cdi
- Поставить галочку "Save security credentials"



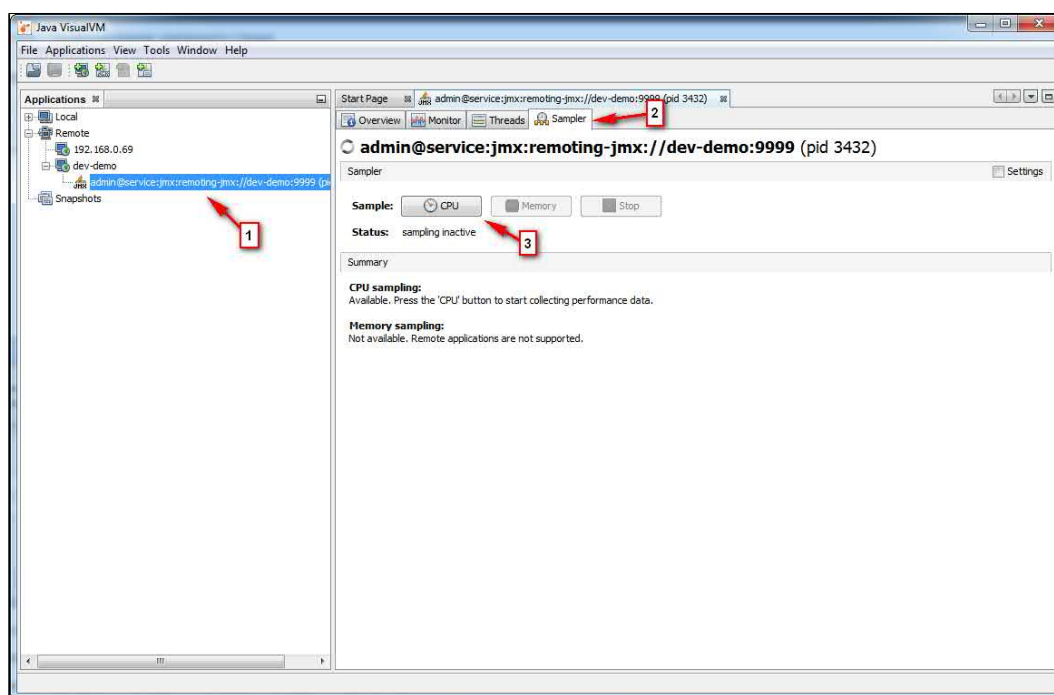
# ЗАПУСК СНЯТИЯ СТАТИСТИКИ JVVM

## Запуск снятия статистики JVVM

❗ Если нужно снять профилировку не только с cdi, но и с factor, то повторить шаги из этого и следующих разделов для JMX-соединения к Фактору (порт 19990).

Основная статистика собирается по использованию процессора (CPU):

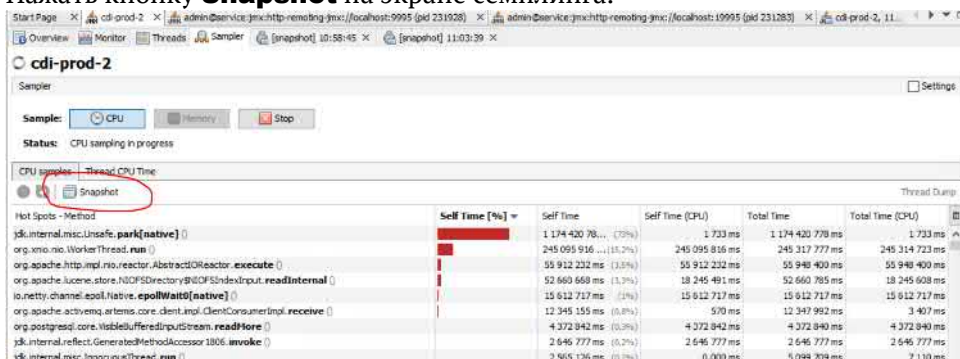
1. Открыть соединение, дважды щелкнув по нему в левой навигационной панели.
2. Открыть вкладку Sampler.
3. Нажать кнопку CPU.



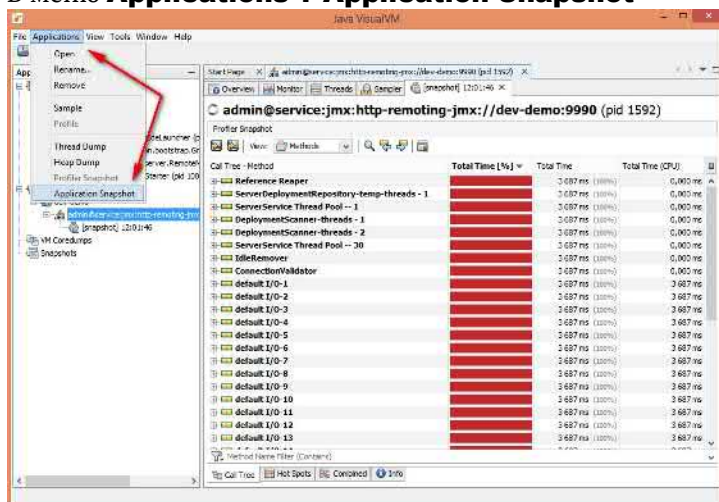
# СНЯТИЕ ПРОФИЛИРОВКИ ПО CPU ЧЕРЕЗ JVVM

## Снятие профилировки по CPU через JVVM

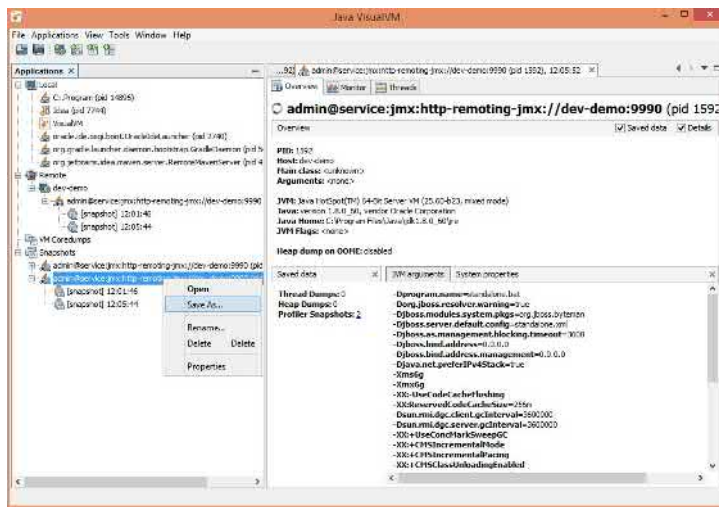
1. Когда накопилась статистика по использованию CPU (не ранее, чем через 15 минут после запуска) необходимо сохранить ее.
2. Нажать кнопку **Snapshot** на экране семплинга.



3. Создать **Application Snapshot**:
  - a. Выбрать в левом меню JMX-соединение к cdi или factor, который профилируем.
  - b. В меню **Applications ? Application Snapshot**



4. Созданный на шаге 6 снелшот отобразится в левой навигационной панели:



5. Сохранить снимок: нажать на него правой кнопкой мыши и выбрать "Save As...". Сохранить обязательно в том формате, которое предлагает приложение.
6. Собранные файлы отправить коллегам HFLabs по почте или прикрепить к задаче в Jira

# МОНИТОРИНГ ПРОИЗВОДИТЕЛЬНОСТИ СИСТЕМ НА LINUX

- Установка утилиты `nmon`
- Запуск утилиты `nmon`
- Профилирование производительности сервера

## Установка утилиты `nmon`

Данная утилита собирает статистику работы машины за временной интервал, при анализе этих данных можно сделать вывод о узком месте в тот или иной промежуток времени.

1. Подключиться по `ssh` к серверу, на котором необходимо выполнить мониторинг производительности.
2. Установить утилиту `nmon`, если не была установлена ранее: **`yum install nmon`** или **`apt-get install nmon`**

## Запуск утилиты `nmon`

1. Подключиться по `ssh` к серверу, на котором необходимо выполнить мониторинг производительности.
  2. Перейти в каталог, куда будет складываться отчет.
  3. Запустить сбор отчета
- При проблеме периодических проблем с производительностью лучше снимать суточный `nmon`

```
nmon -f -s 10 -c 9000
```

- При точечной проблеме (зависании системы достаточно снять метрики за 15 минут)

```
nmon -f -s 1 -c 900
```

где:

- f** — работа в фоне;
- s** — интервал опроса в секундах;
- c** — количество опросов.

## *Профилирование производительности сервера*

В директорию, из которой была запущена утилита **nmom**, сохранен файл вида **<hostname>\_date\_time.nmon** с результатом ее выполнения.

Нужно скопировать данный файл к себе на машину и передать коллегам из поддержки HFLabs.

# УСТАНОВКА УТИЛИТЫ NMON

## *Установка утилиты nmon*

Данная утилита собирает статистику работы машины за временной интервал, при анализе этих данных можно сделать вывод о узком месте в тот или иной промежуток времени.

1. Подключиться по ssh к серверу, на котором необходимо выполнить мониторинг производительности.
2. Установить утилиту nmon, если не была установлена ранее: **yum install nmon** или **apt-get install nmon**

# ЗАПУСК УТИЛИТЫ NMON

## *Запуск утилиты nmon*

1. Подключиться по ssh к серверу, на котором необходимо выполнить мониторинг производительности.
2. Перейти в каталог, куда будет складываться отчет.
3. Запустить сбор отчета

- При проблеме периодических проблем с производительностью лучше снимать суточный nmon

```
nmon -f -s 10 -c 9000
```

- При точечной проблеме (зависании системы достаточно снять метрики за 15 минут)

```
nmon -f -s 1 -c 900
```

где:

- f — работа в фоне;
- s — интервал опроса в секундах;
- c — количество опросов.

# ПРОФИЛИРОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ СЕРВЕРА

## *Профиллирование производительности сервера*

В директорию, из которой был запущена утилита **nmom**, сохранен файл вида **<hostname>\_date\_time.nmon** с результатом ее выполнения.

Нужно скопировать данный файл к себе на машину и передать коллегам из поддержки HFLabs.



# СНЯТИЕ ПОЛНОЙ ПРОФИЛИРОВКИ НА СТЕНДЕ

- Шаг 1. Подготовить утилиты для снятия профилировки
  - Подготовка к запуску
  - Настройка утилиты JVVM
  - Настройка и запуск в Linux
  - Установка утилиты nmon
- Шаг 2. Запустить утилиты для снятия профилировки по нагрузке на сервере приложений
  - Запуск снятия статистики JVVM
  - Запуск утилиты nmon
- Шаг 3. Подождать 15 минут
- Шаг 4. Собрать данные профилировок
  - Снятие профилировки по CPU через JVVM
  - Профилирование производительности сервера

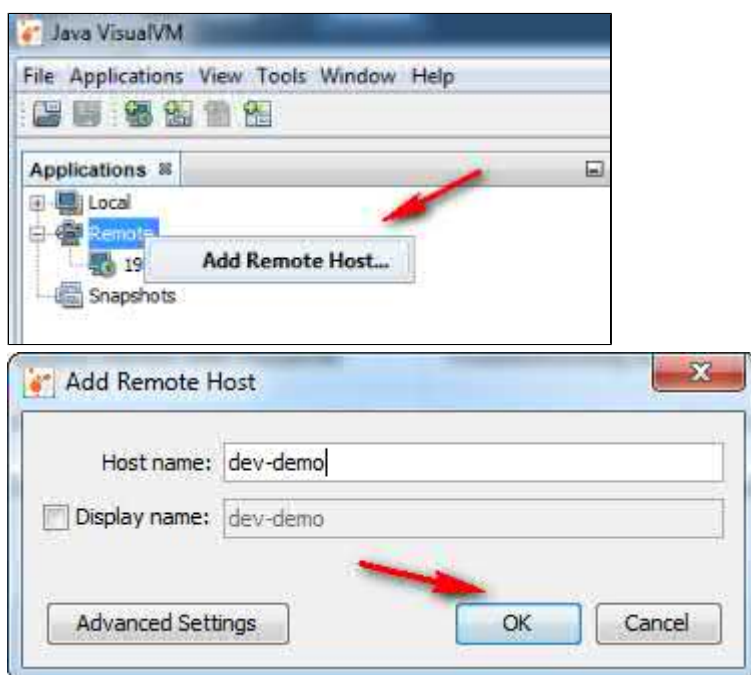
## Шаг 1. Подготовить утилиты для снятия профилировки

### *Подготовка к запуску*

1. Для сбора данных нужна рабочая станция, которая имеет доступ через порты 9990 и 19990 до сервера(ов) приложения, с которых планируется снимать профилировку.
2. Должен быть установлен OpenJdk11.  
Установить можно AdoptOpenJDK с сайта проекта (ссылка: <https://adoptopenjdk.net/archive.html?variant=openjdk11&jvmVariant=hotspot>).
3. Должна быть установлена последняя версия VisualVM (перейти по ссылке: <https://visualvm.github.io/?DownloadStandalone> ? скачать предложенный архив под вашу ОС). Полученный архив распаковать в любую директорию на вашем ПК.

### *Настройка утилиты JVVM*

1. Скачать и распаковать архив **jvvm\_16.zip**.
2. Отредактировать файл **jvvm16.bat**, указав в нем правильный путь до **visualvm.exe** и путь до **OpenJdk**.
3. Запустить утилиту двойным кликом по **bat**-нику **jvvm16.bat**
4. В запущенном JVVM сначала добавить удаленный хост:



5. Создать JMX-соединение для профилирования ЕК (CDI)



6. Задать параметры для профилирования ЕК (cdi):

- Connection-string — **service:jmx:http-remoting-jmx://<hostname>:<port>**, где  
**hostname** — имя или ip-адрес сервера cdi  
**port** — порт для профилировки — 9990 для cdi

Пример строки для подсоединения: **service:jmx:http-remoting-jmx://dev-demo:9990**

- Display name — понятное название создаваемого профиля, например, DEV-CDI
- Логин: admin
- Пароль: cdi

- Поставить галочку "Save security credentials"

Connection:   
**Usage:** <hostname>:<port> OR service:jmx:<protocol>:<sap>

Display name:

Use security credentials

Username:   
 Password:

Save security credentials

Do not require SSL connection

OK Cancel

7. Создать второе JMX-соединение для профилирования Фактора.

- Connection-string — **service:jmx:http-remoting-jmx://<hostname>:<port>**, где **hostname** — имя или ip-адрес сервера Фактора.  
**port** — порт для профилировки. Обычно 19990 для Фактора . (Если для Фактора в конфигурации jboss задано значение port.offset не 10000, а иное, то и порт для профилировки будет иной, а именно: port.offset + 9990).

Пример строки для подсоединения: **service:jmx:http-remoting-jmx://dev-demo:19990**

- Display name — понятное название создаваемого профиля, например, dev-factor
- Логин: admin
- Пароль: cdi
- Поставить галочку "Save security credentials"

## Установка утилиты ntop

Данная утилита собирает статистику работы машины за временной интервал, при анализе этих данных можно сделать вывод о узком месте в тот или иной промежуток времени.

1. Подключиться по ssh к серверу, на котором необходимо выполнить мониторинг производительности.

2. Установить утилиту nmon, если не была установлена ранее: **yum install nmon** или **apt-get install nmon**

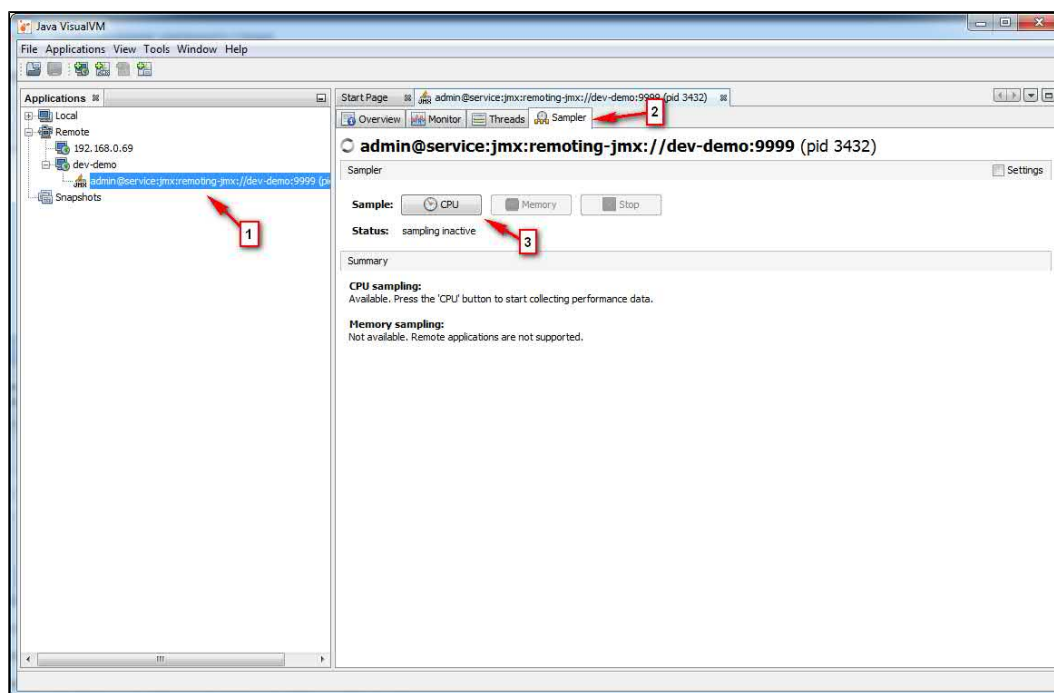
## Шаг 2. Запустить утилиты для снятия профилировки по нагрузке на сервере приложений

### Запуск снятия статистики JVVM

❗ Если нужно снять профилировку не только с cdi, но и с factor, то повторить шаги из этого и следующих разделов для JMX-соединения к Фактору (порт 19990).

Основная статистика собирается по использованию процессора (CPU):

1. Открыть соединение, дважды щелкнув по нему в левой навигационной панели.
2. Открыть вкладку Sampler.
3. Нажать кнопку CPU.



## Запуск утилиты nmon

1. Подключиться по ssh к серверу, на котором необходимо выполнить мониторинг производительности.
2. Перейти в каталог, куда будет складываться отчет.
3. Запустить сбор отчета

- При проблеме периодических проблем с производительностью лучше снимать суточный nmon

```
nmon -f -s 10 -c 9000
```

- При точечной проблеме (зависании системы достаточно снять метрики за 15 минут)

```
nmon -f -s 1 -c 900
```

где:

- f — работа в фоне;
- s — интервал опроса в секундах;
- c — количество опросов.

## Шаг 3. Подождать 15 минут

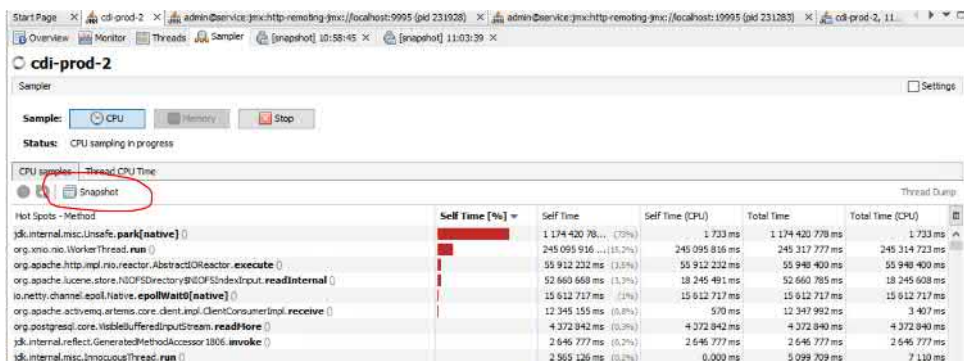
После того как утилиты для сбора данных запущены необходимо засечь 15 минут и дать системам и утилитами поработать в течение этого времени.

Нужно снять 2 снэпшота — через 15 и через 30 минут.

## Шаг 4. Собрать данные профилировок

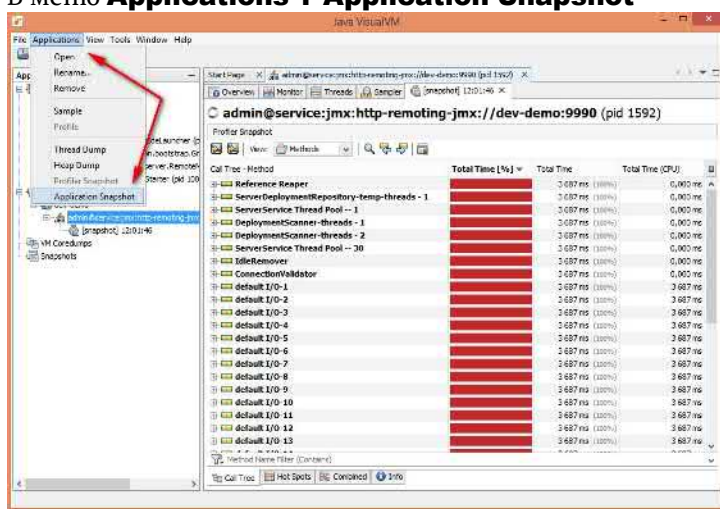
### Снятие профилировки по CPU через JVVM

1. Когда накопилась статистика по использованию CPU (не ранее, чем через 15 минут после запуска) необходимо сохранить ее.
2. Нажать кнопку **Snapshot** на экране семплинга.

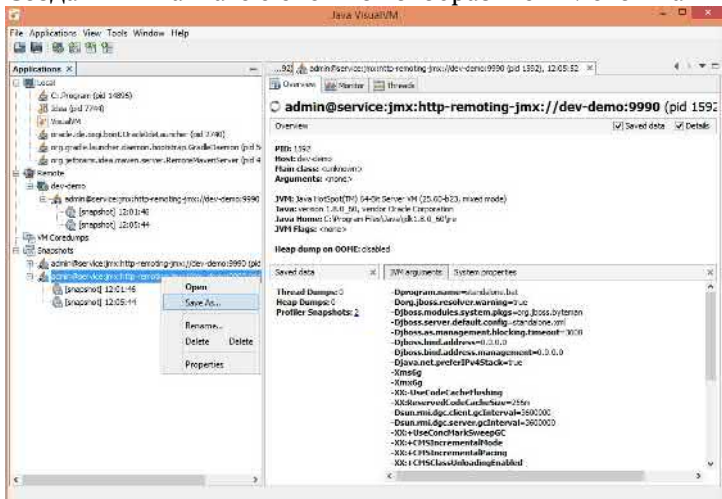


### 3. Создать **Application Snapshot**:

- a. Выбрать в левом меню JMX-соединение к cdi или factor, который профилируем.
- b. В меню **Applications ? Application Snapshot**



### 4. Созданный на шаге 6 снэпшот отобразится в левой навигационной панели:



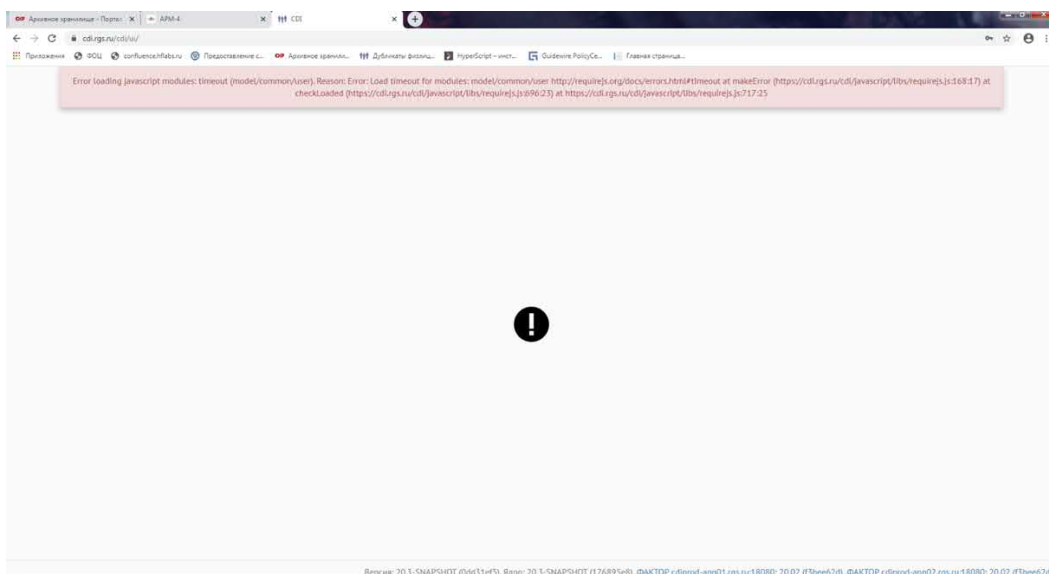
5. Сохранить снэпшот: нажать на него правой кнопкой мыши и выбрать "Save As...". Сохранить обязательно в том формате, которое предлагает приложение.
6. Собранные файлы отправить коллегам HFLabs по почте или прикрепить к задаче в Jira

## *Профилирование производительности сервера*

В директорию, из которой была запущена утилита **nmom**, сохранен файл вида **<hostname>\_date\_time.nmon** с результатом ее выполнения.

Нужно скопировать данный файл к себе на машину и передать коллегам из поддержки HFLabs.

# ДИАГНОСТИКА ЗАПРОСОВ ИЗ БРАУЗЕРА



При ошибке получения ресурсов на клиенте полезно снимать диагностику запросов из браузера:

1. Зайти в Google Chrome на проблемную страницу.
2. Открыть панель разработчика (Ctrl+Shift+I), перейти на вкладку Network.
3. Поставить галочку Preserve log.
4. Обновить страницу и начать вводить данные в поле, к которому подключен CDI.
5. В панели разработчика появятся несколько запросов. По любому из них ткнуть правой кнопкой мыши, выбрать пункт Save as HAR with content, сохранить в файл.
6. Выслать файл нам.



## ТИПОВЫЕ ПРОБЛЕМЫ

- АРМ Менеджера данных долго загружает список контрагентов
- Недоступен CleanService при отсутствии доступа к сервисам ФМС
- Ошибка автоматической миграции

# АРМ МЕНЕДЖЕРА ДАННЫХ ДОЛГО ЗАГРУЖАЕТ СПИСОК КОНТРАГЕНТОВ

Открываю АРМ Менеджера данных, но он виснет на загрузке списка контрагентов. Почему?

1. Скорее всего, в JBoss не подключено [сжатие HTTP-трафика](#). Необходимо проверить и подключить сжатие, если оно до сих пор не было подключено.
2. Если сжатие подключено, то возможно проблемы с поисковым индексом — необходимо посмотреть статистику выгрузки записей на дедубликацию, а также получить информацию, когда индекс последний раз перестраивался и оптимизировался, и сколько времени это заняло.
3. Если индекс в порядке, то необходимо подключить [логирование статистики выполнения поисковых запросов](#)

# НАСТРОЙКА СЖАТИЯ HTTP НА СЕРВЕРЕ ПРИЛОЖЕНИЙ ЕДИНОГО КЛИЕНТА

## *Настройка сжатия HTTP на сервере приложений Единого Клиента*

1. Остановить сервис Единого клиента (если он еще не остановлен).
2. В директории **HOME\_JBOSS\_CDI/standalone/configuration/standalone.xml** добавить раздел

```
<system-properties>
  <property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION" value="on"/>
  <property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION_MIME_TYPES"
value="text/javascript,text/css,text/html,text/xml,text/json,application/x-java-serialized-
object"/>
</system-properties>
```

Между разделами **extensions** и **management**. В итоге получаем:

```
....
<extension module="org.jboss.as.webservices"/>
<extension module="org.jboss.as.weld"/>
</extensions>

<system-properties>
  <property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION" value="on"/>
  <property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION_MIME_TYPES"
value="text/javascript,text/css,text/html,text/xml,text/json,application/x-java-serialized-
object"/>
</system-properties>

<management>
  <security-realms>
    <security-realm name="ManagementRealm">
      <authentication>
        ....
```

# НЕДОСТУПЕН CLEANSERVICE ПРИ ОТСУТСТВИИ ДОСТУПА К СЕРВИСАМ ФМС

**Q:** При недоступности сервиса ФМС по проверке паспортов на действительность CleanService не отвечал на запросы. Как исправить?

**A:** Необходимо в конфигурационном файле /factor/jboss/server/default/deploy/jboss-web.deployer/server.xml найти следующий тег

```
<Connector port="8080"  
  address="{jboss.bind.address}"  
  maxThreads="250" maxHttpHeaderSize="8192"  
  emptySessionPath="true" protocol="HTTP/1.1"  
  enableLookups="false" redirectPort="28443" acceptCount="100"  
  connectionTimeout="20000" disableUploadTimeout="true"  
  restrictedUserAgents="^.*Microsoft-ATL-Native.*$" />
```

И поменять значение параметра maxThreads="250" на 1500. Данный параметр увеличивает максимальное количество потоков, но негативно сказывается на быстродействии. Выбирать его нужно исходя из конкретной нагрузки.

# ОШИБКА АВТОМАТИЧЕСКОЙ МИГРАЦИИ

## *Как определить*

В случае если автоматическая миграция завершилась с ошибкой в логе cdi.log будут записи следующего вида:

```
2018-08-23 13:26:51,424 [ServerService Thread Pool -- 92] INFO DbMigrate - Current version of schema "CDI_NAME": 18.9.0.0
```

...

```
2018-08-23 13:26:51,722 [ServerService Thread Pool -- 92] ERROR DbMigrate - Migration of schema "CDI_NAME" to version 18.10.0.0 - cdi-1024 failed! Please restore backups and roll back database and code!
```

```
2018-08-23 13:26:51,746 [ServerService Thread Pool -- 92] INFO DbRepair - Successfully repaired metadata table "CDI_NAME"."FLYWAY_SCHEMA_HISTORY" (execution time 00:00.012s).
```

```
2018-08-23 13:26:51,747 [ServerService Thread Pool -- 92] INFO DbRepair - Manual cleanup of the remaining effects the failed migration may still be required.
```

```
2018-08-23 13:26:51,752 [ServerService Thread Pool -- 92] ERROR ContextLoader - Context initialization failed
```

```
java.lang.RuntimeException: Failed to update database schema to version 18.10.0.0. Please contact support team.
```

```
Error executing statement at line 36: ALTER TABLE Address ADD district NVARCHAR2 ( 50 ) NULL
```

При этом БД будет находиться в неконсистентном состоянии и запуск приложения будет блокироваться.

## *Что делать*

Собрать [диагностику по логам](#) и обратиться в поддержку HFLabs, приложив собранные логи.

# СТАНДАРТНЫЕ ХАРАКТЕРИСТИКИ СЕРВЕРА ПРИЛОЖЕНИЯ

- CDI
  - Параметры JVM CDI
  - Параметры JBOSS CDI
  - Параметры datasource CDI
  - Прочие параметры
- ФАКТОР
  - Параметры JVM ФАКТОР
  - Параметры JBOSS ФАКТОР

## CDI

### Параметры JVM CDI

```
-server
-ea
-XX:PermSize=128m
-XX:MaxPermSize=256m
-XX:+UseConcMarkSweepGC
-XX:+CMSIncrementalMode
-XX:+CMSIncrementalPacing
-XX:+CMSClassUnloadingEnabled
-XX:CMSIncrementalDutyCycleMin=0
-XX:CMSIncrementalDutyCycle=10
-Xms5g
-Xmx5g
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Duser.timezone=GMT+4
```

Параметр **-Dcdi.root.folder** настраивается отдельно.

### Параметры JBOSS CDI

```
port=8080
address=${jboss.bind.address}
protocol=HTTP/1.1
enableLookups=false
redirectPort=8433
maxPostSize=2097152
maxSavePostSize=4096
maxTime=54148
name=http
scheme=http
secure=false
socketBinding=http
```

## Параметры datasource CDI

```
<connection-property name="v$$session.program">CDIDatasource</connection-property>
<connection-property name="useFetchSizeWithLongColumn">>true</connection-property>
<connection-property name="disableDefineColumnType">>true</connection-property>
<pool>
  <min-pool-size>2</min-pool-size>
  <max-pool-size>50</max-pool-size>
</pool>
<validation>
  <valid-connection-checker class-name="org.jboss.jca.adapters.jdbc.extensions.oracle.
OracleValidConnectionChecker"/>
  <check-valid-connection-sql>select 1 from dual</check-valid-connection-sql>
  <validate-on-match>>false</validate-on-match>
  <background-validation>true</background-validation>
  <background-validation-millis>120000</background-validation-millis>
  <exception-sorter class-name="org.jboss.jca.adapters.jdbc.extensions.oracle.
OracleExceptionSorter"/>
</validation>
<timeout>
  <blocking-timeout-millis>30000</blocking-timeout-millis>
  <idle-timeout-minutes>1</idle-timeout-minutes>
</timeout>
```

## Прочие параметры

```
engine.rebuild.pageSize=10000
engine.rebuild.poolSize=16
engine.search.bufferSize=1000
engine.search.maxBufferedDocs=1000
engine.search.commitInterval=5000
engine.search.needOptimizeWhenOpen=true
engine.search.queueSize=10000

factor.clean.packetSize=200
factor.clean.timeout=600000
factor.dedup.incremental.queueSize=10000
factor.dedup.timeout=600000
factor.factor.timeout=600000

jdbc.batchSize=1000
jdbc.timeout=900

services.cache.refreshIntervalInMin=60
services.deduplication.autoMatchScope=100
services.merge.findCycles=true
services.transform.updateOverMerge=true
services.transform.closeMissedAttributes=false
services.relation.enabled=false
services.actualityDate.truncateTime=true

task.deduplication.pageSize=10000
task.deduplication.poolSize=-1
task.merge.pageSize=1000
task.merge.poolSize=4
task.merge.threshold=100
task.multiclean.timeout=10
task.transform.poolSize=-1
task.reMerge.maxOriginalPartiesCount=1000
```

## **ФАКТОР**

### **Параметры JVM ФАКТОР**

```
-ea  
-Xms6g  
-Xmx8g  
-XX:PermSize=128m  
-XX:MaxPermSize=256m  
-Dsun.rmi.dgc.client.gcInterval=3600000  
-Dsun.rmi.dgc.server.gcInterval=3600000  
-XX:+UseConcMarkSweepGC  
-XX:+CMSIncrementalMode  
-XX:+CMSIncrementalPacing  
-XX:+CMSClassUnloadingEnabled  
-XX:CMSIncrementalDutyCycleMin=0  
-XX:CMSIncrementalDutyCycle=10  
-Duser.timezone=GMT+4  
-Djboss.bind.address=0.0.0.0  
-Djboss.bind.address.management=0.0.0.0  
-Djava.io.charset=Cp1251  
-Dfile.encoding=Cp1251  
-Djboss.socket.binding.port-offset=10000
```

### **Параметры JBOSS ФАКТОР**

```
port=18080  
address=${jboss.bind.address}  
protocol=HTTP/1.1  
maxPostSize = 2097152  
maxSavePostSize = 4096  
maxTime = 483  
name = http  
redirectPort = 8433  
requestCount = 87014  
scheme = http  
secure = false  
socketBinding = http
```



# КАК СОЗДАТЬ ЗАПРОС В ПОДДЕРЖКУ И УПРАВЛЯТЬ ИМ

- [Инструкция по работе с JIRA](#)
- [Собрать диагностику](#)

# ИНСТРУКЦИЯ ПО РАБОТЕ С JIRA

Все вопросы лучше писать в JIRA, а не на почту / в скайп сотрудникам ХФЛабс

Это позволяет лучше отслеживать статус запросов, вести переписку по запросу в одном месте и пользоваться прочими преимуществами системы учета запросов.

- [Что писать](#)
- [Куда писать](#)
- [Статусы задач](#)
- [Что делать с задачей](#)
- [Что такое dashboard?](#)
- [Доступ к системе](#)

## Что писать

Что можно и нужно писать в нашу систему поддержки:

- ошибки
- запросы на доработки
- вопросы по функционированию системы

Для этого необходимо создать задачу типа *Support Request*, а сотрудник ХФЛабс при необходимости сам поменяет тип и(или) приоритет задачи.

## Куда писать

1. Нажать на кнопку *Создать запрос*
2. Заполнить поля *Тема* и *Описание* (аналогично полям в электронном письме)
3. Обязательно приложить [диагностику по контрагенту](#) или логи за сегодняшний день. При необходимости можно добавить скриншот и другие артефакты, необходимые для воспроизведения проблемы

Здесь выложен ролик, который показывает как создавать и работать с запросами в поддержку.

## Статусы задач

Статус	Описание
<i>Open</i>	Задача открыта, по ней проводятся работы
<i>Reopen</i>	
<i>Delayed</i>	Работы по задаче приостановлены.

---

Задача не закрыта, но и не в работе (например, требуется согласование требований или завершение другой задачи)

---

*Resolved* Задача выполнена (Если это вопрос по функциональности - ответ на вопрос дан)

---

*Closed* Задача закрыта, все работы по ней завершены

---

## Что делать с задачей

1. Если была создана **ошибка**, сотрудники HFLabs решают ее и переводят в статус *Resolved*.

Вы проверяете на своем окружении, что ошибка действительно исправлена, и закрываете задачу.

2. Если был создан **запрос на доработку**, статус задачи меняют сотрудники HFLabs.

Мы на своей стороне разрабатываем функционал, переводя в статус *Resolved*, а потом тестируем его, закрывая задачу.

Задача указывается в Release Notes и, если у Вас возникнут вопросы по реализации после тестирования на своей платформе, можно будет создать отдельную заявку на это.

Если Вы назначены исполнителем по доработке, Вам необходимо ответить на вопрос в комментарии и нажать на кнопку "*Назначить*", выбрав ответственного за поддержку.

3. Если был создан **вопрос по функционированию системы**, сотрудники HFLabs отвечают на него и переводят в статус *Resolved*.

Вы закрываете задачу, если вопросов по данной функциональности больше нет.

## Что такое dashboard?

Dashboard - это страница, на которую вы попадаете сразу после ввода логина и пароля.

Для удобства использования в правой части dashboard-а расположены 2 фильтра:

- *Assign to me* - запросы в статусе "Open", требующие консультации. Необходимо ответить на вопрос в комментарии и назначить его обратно на сотрудника HFLabs.
- *I am closing* - запросы в статусе "Resolved". Это вопросы в поддержку, на которые был дан ответ. Если ответ вас устраивает, закройте задачу, нажав на кнопку "Close Issue".

Таким образом, зайдя в систему, можно легко и быстро оценить, ждет ли сотрудник ХФЛабс от Вас ответа по какой-либо задаче.

## Доступ к системе

Логины и пароли к системе учета запросов выдаются сотрудниками ХФЛабс по запросу.

# СОБРАТЬ ДИАГНОСТИКУ

## Когда использовать

По просьбе сотрудника техподдержки.

## Как собрать диагностику

1. В АРМ Администратора зайдите на вкладку *Задачи*, раздел *Диагностика*.

Действия	Название	Описание
Детали   Редактировать   Выполнить	diagnosticLogs	Сбор логов
Детали   Редактировать   Выполнить	diagnosticMetrics	Сбор метрик
Детали   Редактировать   Выполнить	diagnosticParty	Сбор диагностики по контрагенту
Детали   Редактировать   Выполнить	diagnosticProfiles	Сбор диагностики с указанным профилем
Детали   Редактировать   Выполнить	diagnosticTaskStat	Сбор диагностики по задаче
Детали   Редактировать   Выполнить	mergedPartiesCheck	Найти проблемы в золотых карточках
Детали   Редактировать   Выполнить	mergedPartiesCheckAndRemerge	Найти и исправить проблемы в золотых карточках
Детали   Редактировать   Выполнить	runDBCheckers	Проверка БД на непротиворечивость

2. Соберите диагностику:

- Логи (**diagnosticLogs**).
- Информация по контрагенту (**diagnosticParty**).
- Диагностика по профилю (**diagnosticProfiles**).
- Проверка БД на непротиворечивость (**runDBCheckers**).
- Проверка на непротиворечивость и исправление проблем (**mergedPartiesCheckAndRemerge**).

3. Архив собранной информации ЕК автоматически отправит по электронной почте администратору и в техподдержку (после [настройки](#)).

Копию архива сохранит на сервере во временной директории (**{TEMP}/diagnostic.{date}\_{time}**).

4. Иногда случается, что логи увеличиваются и архив не помещается во вложение письма. Тогда «Единый клиент» выбирает все файлы до превышения размера в 75Мб и этот архив отправляет в письме. Полный архив диагностики сохраняется во временной директории {TEMP} с постфиксом **\_full**.

5. В папке {TEMP} «Единый клиент» хранит диагностики только за последние три дня (начиная с версии 18.14). Все что старше — удаляется, чтобы беречь ресурсы.

6. Название архива и его месторасположение можно посмотреть в панели администратора

— **diagnosticClusterTaskPerformer: processed 0 entities; 0 with errors; took 7 seconds or 7.39 sec; diagnostic saved to C:\cdi\tmp\diag\_10.0.18.47\_181008\_2355.zip; diagnostic sent to cdi\_qa@hflabs.ru**

# МОНИТОРИНГ РАБОТЫ СИСТЕМЫ

Проверить работоспособность системы:

- Сетевая доступность
- Доступность БД
- Доступность «Фактора»
- Доступность «Единого клиента»
- Работоспособность задач пакетной обработки данных
- Работоспособность интерфейса пользователя
- Мониторинг журналов
- Метрики для мониторинга

Рекомендуем проверять в автоматическом режиме и оповещать администратора системы по email / SMS в случае проблем.

# СЕТЕВАЯ ДОСТУПНОСТЬ

Убедитесь в следующем:

1. Открыт доступ к серверу СУБД по порту, на котором слушает Oracle (обычно 1521) или PostgreSQL (обычно 5432).
2. Открыт доступ к серверу LDAP по соответствующему порту.
3. На сервере приложений открыты порты
  - 8080 / 8443 (HTTP/HTTPS-порты Единого клиента)
  - 18080 / 18443 (HTTP/HTTPS-порты ФАКТОР)

# ДОСТУПНОСТЬ БД

- Порт
- Использование процессора
- Свободное место на диске (табличное пространство CDI)
- Свободное место на диске (табличное пространство CDI\_IDX)
- Свободное место на диске (табличное пространство TEMP)
- Использование TEMP

## Порт

### *Периодичность*

1 раз в минуту

### *Проверка*

На сервере СУБД открыт и прослушивается порт:

- для Oracle — 1521;
- для PostgreSQL — 5432;
- для MariaDB — 3306.

Если используется альтернативный порт — уточните его у администратора СУБД.

### *Признак аварии*

Порт не прослушивается.

### *Что делать*

Убедиться, что СУБД запущена, а порт не закрыт брандмауэром или промежуточным сетевым оборудованием.

## Использование процессора

### *Периодичность*

1 раз в минуту



## *Проверка*

Процент использования процессора.

## *Признак аварии*

Больше 95% в течение 15 минут.

## *Что делать*

Собрать **диагностику**. Результат отправить в ХФ Лабс.

# Свободное место на диске (табличное пространство CDI)

## *Периодичность*

1 раз в минуту

## *Проверка*

Выполнить запрос:

```
select tsu.tablespace_name,  
       ceil(tsu.used_mb) "size MB",  
       decode(ceil(tsf.free_mb), NULL, 0, ceil(tsf.free_mb)) "free MB",  
       decode(100 - ceil(tsf.free_mb/tsu.used_mb*100), NULL, 100, 100 - ceil(tsf.free_mb/tsu.  
used_mb*100)) "% used"  
from  
  (select tablespace_name, sum(bytes)/1024/1024 used_mb  
   from dba_data_files group by tablespace_name  
  ) tsu,  
  (select tablespace_name, sum(bytes)/1024/1024 free_mb  
   from dba_free_space group by tablespace_name  
  ) tsf  
where tsu.tablespace_name = tsf.tablespace_name (+)  
and tsu.tablespace_name = 'CDI'  
order by 1;
```

У пользователя, который выполняет запрос, должны быть права на чтение системных таблиц dba\_free\_space, dba\_temp\_files и dba\_data\_files.

## *Признак аварии*

Значение "free MB" меньше 2000.

## *Что делать*

Выделить дополнительное дисковое пространство для табличного пространства CDI.

## Свободное место на диске (табличное пространство CDI\_IDX)

### Периодичность

1 раз в минуту

### Проверка

Выполнить запрос:

```
select tsu.tablespace_name,  
       ceil(tsu.used_mb) "size MB",  
       decode(ceil(tsf.free_mb), NULL, 0, ceil(tsf.free_mb)) "free MB",  
       decode(100 - ceil(tsf.free_mb/tsu.used_mb*100), NULL, 100, 100 - ceil(tsf.free_mb/tsu.  
used_mb*100)) "% used"  
from  
  (select tablespace_name, sum(bytes)/1024/1024 used_mb  
   from dba_data_files group by tablespace_name  
  ) tsu,  
  (select tablespace_name, sum(bytes)/1024/1024 free_mb  
   from dba_free_space group by tablespace_name  
  ) tsf  
where tsu.tablespace_name = tsf.tablespace_name (+)  
and tsu.tablespace_name = 'CDI_IDX'  
order by 1;
```

У пользователя, который выполняет запрос, должны быть права на чтение системных таблиц dba\_free\_space, dba\_temp\_files и dba\_data\_files.

### Признак аварии

Значение "free MB" меньше 1000.

### Что делать

Выделить дополнительное дисковое пространство для табличного пространства CDI\_IDX.

## Свободное место на диске (табличное пространство TEMP)

### Периодичность

1 раз в минуту

### Проверка

Выполнить запрос:

```
SELECT  
D.mb_total "size MB",
```

```

SUM (A.used_blocks * D.block_size) / 1024 / 1024 "used MB",
D.mb_total - SUM (A.used_blocks * D.block_size) / 1024 / 1024 "free MB"
FROM v$sort_segment A,
(SELECT B.name, C.block_size, SUM (C.bytes) / 1024 / 1024 mb_total
FROM v$tablespace B, v$tempfile C
WHERE B.ts#= C.ts#
GROUP BY B.name, C.block_size) D
WHERE A.tablespace_name = D.name
GROUP by A.tablespace_name, D.mb_total;

```

У пользователя, который выполняет запрос, должны быть права на чтение системных таблиц dba\_free\_space, dba\_temp\_files и dba\_data\_files.

### *Признак аварии*

Значение "free MB" меньше 1000.

### *Что делать*

Выделить дополнительное дисковое пространство для табличного пространства TEMP.

## Использование TEMP

### *Периодичность*

1 раз в 12 часов

### *Проверка*

Выполнить запрос:

```

select (select tablespace_size / 1024 / 1024 from dba_temp_free_space) - temp_mb from (
select max(temp_mb) temp_mb from (
select sample_time, sum(temp_mb) temp_mb from (
select session_id, to_char(sample_time, 'YYYY-MM-DD HH24:MI') sample_time, max
(temp_space_allocated / 1024 / 1024) temp_mb
from DBA_HIST_ACTIVE_SESS_HISTORY
where
sample_time > sysdate - 0.5
group by session_id, to_char(sample_time, 'YYYY-MM-DD HH24:MI')
order by sample_time desc
) group by sample_time
));

```

У пользователя, который выполняет запрос, должны быть права на чтение системных таблиц dba\_free\_space, dba\_temp\_files, dba\_data\_files и dba\_hist\_active\_sess\_history.

### *Признак аварии*

Значение меньше 1000 — предупреждение.

## *Что делать*

Сообщить в ХФ Лабс.

# ДОСТУПНОСТЬ «ФАКТОРА»

- Порт
- Использование процессора
- Свободное место на диске
- Доступность SOAP-сервиса стандартизации
- Корректная работа SOAP-сервиса стандартизации
- Ошибки в журнале

## Порт

### *Периодичность проверки*

1 раз в минуту

### *Проверка*

Система слушает на порту 18080 (если JBOSS работает в режиме HTTP) или на порту 8443 (если JBOSS работает в режиме HTTPS).

Пример (Linux):

```
$ netstat -an | grep 18080
tcp    0    0 0.0.0.0:18080      0.0.0.0:*        LISTEN

$ netstat -an | grep 8443
tcp    0    0 0.0.0.0:8443      0.0.0.0:*        LISTEN
```

Пример (Windows):

```
C:\>netstat -an | findstr 18080
TCP    0.0.0.0:18080      0.0.0.0:0        LISTENING

C:\>netstat -an | findstr 8443
TCP    0.0.0.0:8443      0.0.0.0:0        LISTENING
```

### *Признак аварии*

Порт не прослушивается.

### *Что делать*

Убедиться, что Фактор запущен, а порт не закрыт брандмауэром или промежуточным сетевым оборудованием.

## Использование процессора

### *Периодичность*

1 раз в минуту

### *Проверка*

Процент использования процессора.

### *Признак аварии*

Больше 95% в течение 15 минут.

### *Что делать*

Собрать [диагностику](#). Результат отправить в ХФ Лабс.

## Свободное место на диске

### *Периодичность*

1 раз в минуту

### *Проверка*

Объем свободного места в файловой системе, где установлен Фактор.

### *Признак аварии*

Свободно меньше 10%.

### *Что делать*

Выделить дополнительное дисковое пространство.

## Доступность SOAP-сервиса стандартизации

### *Периодичность проверки*

1 раз в 5 минут

### *Проверка*

HTTP-запрос по следующему адресу возвращает код 200 ОК:

- Если используется HTTP: **http://{hostname}:18080/factor-service-{customer}/services/CleanService?wsdl**
- Если используется HTTPS: **https://{hostname}:8443/factor-service-{customer}/services/CleanService?wsdl**

Где **{hostname}** — доменное имя сервера приложения, а **{customer}** — код заказчика.

Пример (Linux):

```
$ wget --no-cache --spider http://factor.example.com:18080/factor-service-acme/services/CleanService?wsdl

--15:36:45-- http://factor.example.com:18080/factor-service-acme/services/CleanService?wsdl
Connecting to factor.example.com:18080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6790 (6.6K) [text/xml]
200 OK
```

### *Признак аварии*

Код ответа, отличный от 200 OK.

### *Что делать*

Убедиться, что Фактор запущен. Проверить использование процессора и место на диске.

## Корректная работа SOAP-сервиса стандартизации

### *Периодичность проверки*

1 раз в 15 минут

### *Проверка*

Тестовый SOAP-запрос (тело запроса см. в любом примере [CleanService](#)) к SOAP-сервису **Clean Service**.

Обращение к SOAP-сервису производится по следующему URL:

- Если используется HTTP: **http://{hostname}:18080/factor-service-{customer}/services/CleanService**
- Если используется HTTPS: **https://{hostname}:8443/factor-service-{customer}/services/CleanService**

### *Признак аварии*

- Код ответа отличается от 200 OK.

- Ответ не соответствует эталонному (эталонный ответ см. в примерах).
- Время отклика больше 30 секунд.

Время отклика от 10 до 30 секунд — предупреждение.

### *Что делать*

Убедиться, что Фактор запущен. Проверить использование процессора и место на диске.

## Ошибки в журнале

### *Периодичность проверки*

1 раз в 5 минут

### *Проверка*

В журнале **JBOSS\_FACTOR\_HOME/standalone/log/server.log** отсутствуют ошибки (строки, содержащие **ERROR**).

Пример (Linux):

```
$ grep ERROR "$JBOSS_FACTOR_HOME/standalone/log/server.log"
```

Пример (Windows):

```
C:\>findstr /L ERROR "%JBOSS_FACTOR_HOME%\standalone\log\server.log"
```

### *Признак аварии*

Наличие ошибок — предупреждение.

### *Что делать*

Диагностировать причину. При необходимости сообщить в ХФ Лабс.



# ДОСТУПНОСТЬ «ЕДИНОГО КЛИЕНТА»

- Порт
- Использование процессора
- Свободное место на диске
- Доступность APM Администратора
- Доступность SOAP-сервисов

## Порт

### *Периодичность проверки*

1 раз в минуту

### *Проверка*

Система слушает на порту 8080 (если WildFly работает в режиме HTTP) или на порту 8443 (если WildFly работает в режиме HTTPS).

Пример (Linux):

```
$ netstat -an | grep 8080
tcp    0    0 0.0.0.0:8080      0.0.0.0:*        LISTEN

$ netstat -an | grep 8443
tcp    0    0 0.0.0.0:8443      0.0.0.0:*        LISTEN
```

Пример (Windows):

```
C:\>netstat -an | findstr 8080
TCP    0.0.0.0:8088      0.0.0.0:0        LISTENING

C:\>netstat -an | findstr 8443
TCP    0.0.0.0:8443      0.0.0.0:0        LISTENING
```

### *Признак аварии*

Порт не прослушивается.

### *Что делать*

Убедиться, что Единый клиент запущен, а порт не закрыт брандмауэром или промежуточным сетевым оборудованием.

## Использование процессора

### *Периодичность*

1 раз в минуту

### *Проверка*

Процент использования процессора.

### *Признак аварии*

Больше 95% в течение 15 минут, при условии, что не выполняются задачи пакетной обработки данных.

Выполнение задач проверять SQL-запросом:

```
select count(*) from task_execution  
where task_key in  
('findDuplicatesAndMerge','findDuplicates','incrementDuplicatesAndMerge','importExternalID  
uplicateWithMerge')  
and end_date is null  
and status != 'SKIPPED'
```

Если задачи не выполняются, запрос возвращает 0.

### *Что делать*

Собрать диагностику. Результат отправить в ХФ Лабс.

## Свободное место на диске

### *Периодичность*

1 раз в минуту

### *Проверка*

Объем свободного места в файловой системе, где установлен Единый клиент.

### *Признак аварии*

Свободно меньше 10% — авария

Свободно меньше 20% при условии, что не выполняется задача перестроения поисковых индексов — предупреждение.

Выполнение задач проверять SQL-запросом:

```
select count(*) from task_execution
where (task_key like '%FullRebuild' or task_key like 'allEnginesRebuild%')
and end_date is null
and status != 'SKIPPED'
```

Если задачи не выполняются, запрос возвращает 0.

### *Что делать*

Выделить дополнительное дисковое пространство.

## Доступность АРМ Администратора

### *Периодичность проверки*

1 раз в 15 минут

### *Проверка*

HTTP-запрос по следующему адресу возвращает код 200 ОК:

- Если используется HTTP: **http://{hostname}:8080/cdi/login.html**
- Если используется HTTPS: **https://{hostname}:8443/cdi/login.html**

Где **{hostname}** — доменное имя сервера приложения.

Пример (Linux):

```
$ wget --no-cache --spider http://cdi.example.com:8080/cdi/login.html
--15:36:45-- http://cdi.example.com:8080/cdi/login.html
Connecting to cdi.example.com:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
200 OK
```

### *Признак аварии*

Код ответа, отличный от 200 ОК.

### *Что делать*

Убедиться, что Единый клиент запущен.

## Доступность SOAP-сервисов

### *Периодичность проверки*

1 раз в 5 минут

### *Проверка*

HTTP-запрос по следующему адресу возвращает код 200 OK:

- Если используется HTTP:

```
http://{hostname}:8080/cdi/soap/services/2_13/PartyWS?wsdl  
http://{hostname}:8080/cdi/soap/services/15_3/TaskWS?wsdl
```

- Если используется HTTPS:

```
http://{hostname}:8443/cdi/soap/services/2_13/PartyWS?wsdl  
http://{hostname}:8443/cdi/soap/services/15_3/TaskWS?wsdl
```

Где **{hostname}** — доменное имя сервера приложения.

Пример (Linux):

```
$ wget --no-cache --spider http://cdi.example.com:8080/cdi/soap/services/2_13/PartyWS?wsdl  
  
--15:36:45-- http://cdi.example.com:8080/cdi/soap/services/2_13/PartyWS?wsdl  
Connecting to factor.example.com:8080... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 6790 (6.6K) [text/xml]  
200 OK
```

### *Признак аварии*

Код ответа, отличный от 200 OK.

### *Что делать*

Убедиться, что Единый клиент запущен. Проверить использование процессора и место на диске.

## Ошибки в журнале

### *Периодичность проверки*

1 раз в 5 минут

## *Проверка*

В журнале **JBOSS\_CDI\_HOME/standalone/log/server.log** отсутствуют ошибки (строки, содержащие **ERROR**).

Пример (Linux):

```
$ grep ERROR "$JBOSS_CDI_HOME/standalone/log/server.log"
```

Пример (Windows):

```
C:\>findstr /L ERROR "%JBOSS_CDI_HOME%\standalone\log\server.log"
```

## *Признак аварии*

Наличие ошибок — предупреждение.

## *Что делать*

Диагностировать причину. При необходимости сообщить в ХФ Лабс.

# РАБОТОСПОСОБНОСТЬ ЗАДАЧ ПАКЕТНОЙ ОБРАБОТКИ ДАННЫХ

- Наличие входящих инкрементов
- Зависшие задачи
- Задачи не запускаются

## Наличие входящих инкрементов

### *Периодичность проверки*

1 раз в сутки (перед тем, как ЕК начинает забирать инкремент)

### *Проверка*

Выполнить SQL-запрос:

```
select count(*) from БУФЕРНАЯ_ТАБЛИЦА where created between sysdate-3 and sysdate;
```

### *Признак аварии*

Запрос вернул 0.

### *Что делать*

Выяснить, почему инкремент из исходных систем не доходит до буферной таблицы.

## Зависшие задачи

### *Периодичность*

1 раз в сутки

### *Проверка*

Выполнить SQL-запрос:

```
select count(*) from task_execution  
where start_date between sysdate-2 and sysdate-1  
where end_date is null;
```

### *Признак аварии*

Запрос не вернул 0.

### *Что делать*

Собрать *диагностику*. Результат отправить в ХФ Лабс.

## **Задачи не запускаются**

### *Периодичность*

1 раз в сутки

### *Проверка*

Выполнить SQL-запрос:

```
select count(*) from task_execution  
where start_date > sysdate-2  
AND STATUS != 'SKIPPED';
```

### *Признак аварии*

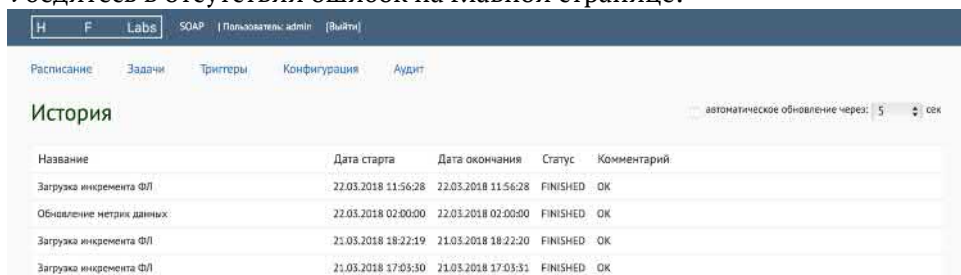
Запрос вернул 0.

### *Что делать*

Собрать *диагностику*. Результат отправить в ХФ Лабс.

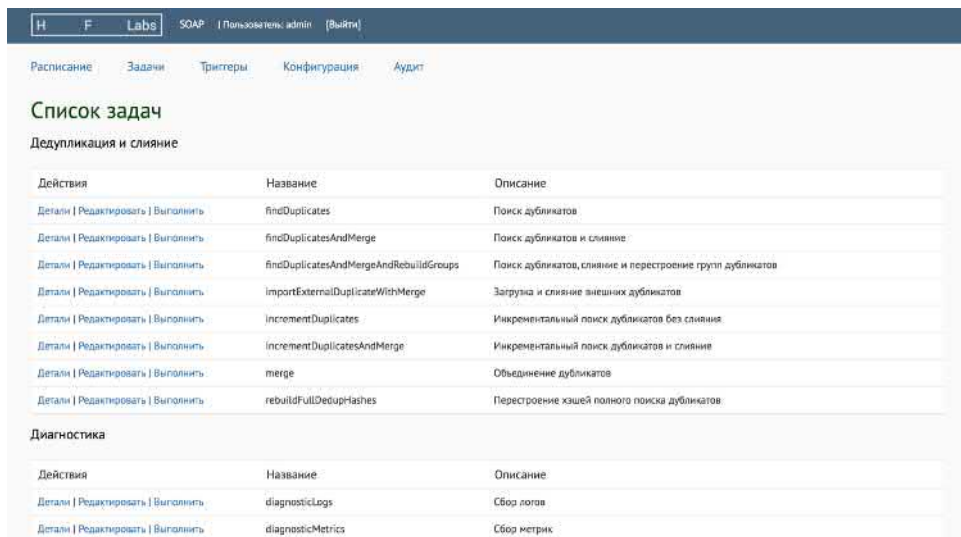
# РАБОТОСПОСОБНОСТЬ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

1. Перейдите в браузере по адресу `http://{hostname}:8080/cdi`, и авторизуйтесь в системе, как описано в разделе [Вход в систему \(администратор\)](#).
2. Убедитесь в отсутствии ошибок на главной странице:



Название	Дата старта	Дата окончания	Статус	Комментарий
Загрузка инкремента ФЛ	22.03.2018 11:56:28	22.03.2018 11:56:28	FINISHED	OK
Обновление метрик данных	22.03.2018 02:00:00	22.03.2018 02:00:00	FINISHED	OK
Загрузка инкремента ФЛ	21.03.2018 18:22:19	21.03.2018 18:22:20	FINISHED	OK
Загрузка инкремента ФЛ	21.03.2018 17:03:30	21.03.2018 17:03:31	FINISHED	OK

3. Перейдите на страницу *Задачи* и убедитесь в наличии задач в списке:



Действия	Название	Описание
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	findDuplicatess	Поиск дубликатов
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	findDuplicatessAndMerge	Поиск дубликатов и слияние
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	findDuplicatessAndMergeAndRebuildGroups	Поиск дубликатов, слияние и перестроение групп дубликатов
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	importExternalDuplicateWithMerge	Загрузка и слияние внешних дубликатов
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	IncrementDuplicatess	Инкрементальный поиск дубликатов без слияния
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	IncrementDuplicatessAndMerge	Инкрементальный поиск дубликатов и слияние
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	merge	Объединение дубликатов
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	rebuildFullDedupHashes	Перестроение хэшей полного поиска дубликатов

Действия	Название	Описание
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	diagnosticLogs	Сбор логов
<a href="#">Детали</a>   <a href="#">Редактировать</a>   <a href="#">Выполнить</a>	diagnosticMetrics	Сбор метрик



# МОНИТОРИНГ ЖУРНАЛОВ

Для мониторинга ошибок, возникающих в ходе работы системы, необходимо отслеживать журналы системы на наличие записей с уровнем протоколирования ERROR.

## Статистика по пакетным задачам

Для сбора статистики по пакетным задачам обработки данных (загрузка, поиск дубликатов, и т. п.) можно настроить сбор информации из журналов системы на наличие строк вида:

```
... {статус} task '{имя_задачи}'; processed {количество} entities ({доп_информация});  
{количество_ошибок} errors; took {время_выполнения} seconds
```

Где:

- {статус} --- может принимать одно из значений:
  - FINISHED — если задача завершилась (даже если в ходе выполнения возникали отдельные ошибки);
  - ERROR — если выполнение задачи завершилось глобальной ошибкой;
  - CANCELED — если выполнение задачи прервано пользователем;
- {имя\_задачи} --- наименование задачи;
- {количество} --- общее количество обработанных сущностей (как правило, контрагентов);
- {доп\_информация} --- детальная информация по обработанным сущностям (заполняется только для некоторых задач);
- {количество\_ошибок} --- количество ошибок, произошедших во время выполнения задачи;
- {время\_выполнения} --- время выполнения задачи в секундах.

Сбор информации из журналов должен производиться внешними по отношению к ЕК средствами (например, с помощью специализированных систем мониторинга).

### Примеры сообщений в журналах:

Для задачи загрузки данных:

```
2012-12-27 17:49:18,098 [TaskManagerExecutor-1] INFO interceptor.  
TaskPerformerStatsInterceptor - FINISHED task 'bufferUploadTaskPerformer'; processed  
163 entities (update: 0, skip: 0, create: 163, close: 0); 0 errors; took 12 seconds
```

Для задачи поиска дубликатов:

```
2012-12-27 18:04:28,128 [TaskManagerExecutor-1] INFO interceptor.  
TaskPerformerStatsInterceptor - FINISHED task 'deDuplicationTaskPerformer'; processed  
115 entities (update: 0, skip: 0, create: 115, close: 0); 0 errors; took 2 seconds
```

Для задачи слияния дубликатов:

```
2012-12-27 18:04:32,036 [TaskManagerExecutor-1] INFO interceptor.  
TaskPerformerStatsInterceptor - FINISHED task 'mergeTaskPerformer'; processed 59  
entities (); 0 errors; took 3 seconds
```



# МЕТРИКИ ДЛЯ МОНИТОРИНГА

Метрики для настройки в сторонних системах мониторинга (prometheus+grafana). Подключаются по запросу.

Метрика	Описание	Тэги (измерения)	С какого релиза
<b><i>Единый клиент</i></b>			
cdi_api_requests_count	Общее количество запросов к внешнему API	<ul style="list-style-type: none"><li>service - название Endpoint-a</li><li>operation - название метода</li><li>username - учетка</li><li>ip - ip сервера, с которого был вызов.</li><li>system - исходная система</li><li>quantile - 0.9 / 0.95 / 0.99</li></ul>	21.21
cdi_api_requests_max	Максимальное время выполнения вызова (сек)		
cdi_api_requests_90	Значение 90/95/99 перцентиля на запрос (сек)		
cdi_api_errors_total	Количество ошибочных запросов		
<hr/>			
cdi_ui_api_requests_count	Общее количество запросов к UI контроллерам	<ul style="list-style-type: none"><li>service - название контроллера</li><li>operation - название метода</li><li>quantile - 0.9 / 0.95 / 0.99</li></ul>	21.23
cdi_ui_api_requests_max	Максимальное время выполнения вызова (сек)		
cdi_ui_api_requests_90	Значение 90/95/99 перцентиля на запрос (сек)		
cdi_ui_api_errors_total	Количество ошибочных запросов		
cdi_ui_api_errors_cancelled_total	Количество отмененных запросов		
<hr/>			
cdi_sync_queue_size_max	Размер очереди синхронизации между нодами в ГР		
<hr/>			
cdi_sync_latency_max	Макс. задержка в доставке сообщений синхронизации между нодами в ГР (мс)		
<hr/>			
data_source_ping	Пинг до базы данных (мс)		21.22

## ***Общий мониторинг сервера и нагрузки***

disk_total	Общее количество места (байт)	
disk_free	Количество свободного места (байт)	<p><b>name</b> {НазваниеПараметра} — идентификатор содержимого, возможные значения:</p> <ul style="list-style-type: none"> <li>• <b>root</b> — папка, указанная в cdi.root.folder;</li> <li>• <b>tmp</b> — папка с временными файлами;</li> </ul>
		<p><b>path</b> {Путь} — путь к директории</p>
system_cpu_ticks	Нагрузка на ЦПУ	<p><b>metric</b> {Процесс} — компонент загрузки ЦПУ.</p> <p><b>options</b> Возможные значения:</p> <ul style="list-style-type: none"> <li>• <b>system</b> — нагрузка на сри на обслуживание переключений между задачами;</li> <li>• <b>steal</b> — нагрузка на сри на обслуживание виртуализации (только для виртуальных машин linux);</li> <li>• <b>iowait</b> — нагрузка на сри на обслуживание ввода-вывода;</li> <li>• <b>soft_irq</b> — нагрузка на сри на прерывания;</li> <li>• <b>user</b> — нагрузка на сри от всех приложений, работающих на сервере;</li> <li>• <b>idle</b> — отсутствие нагрузки на сри (сри не используется ни одним приложением);</li> <li>• <b>hard_irq</b> — нагрузка на сри на обработку аппаратных прерываний;</li> <li>• <b>nice</b> — нагрузка на сри на процессы с низким приоритетом</li> </ul>
process_cpu_usage	Нагрузка на ЦПУ от «ЕК»	
process_files_open	Количество открытых файловых дескрипторов «ЕК»	
process_threads	Количество активных потоков, которые использует «ЕК»	
jvm_gc_pause	Максимальные паузы GC (сек)	<p><b>cause</b> <b>G1 Evacuation Pause</b> — фаза GC</p>
jvm_memory_used	Использование памяти приложения (байт)	<p><b>area</b> <b>heap</b> — область памяти</p> <p><b>id</b> {ОбластьПамяти} — идентификатор области внутри <b>area</b>. Возможные значения:</p>

- **G1 Eden Space** — динамическая память, которая используется «Фактором»;
- **G1 Old Gen** — память, занятая под долгосрочные объекты (справочники, фильтры, классы и т.д.)

process\_memory\_gss Кол-во памяти, выделенное приложению, которое реально находится в оперативной памяти (байт)

process\_memory\_vss Кол-во используемой виртуальной памяти.  
Используется для работы со структурами данных через ММАР (байт)

process\_memory\_swap Количество памяти приложения, которое было перенесено ОС из RAM в swap (байт)

logback\_events\_total Количество операций логирования

**{Уровень Логирования}** — уровень логирования для которого необходимо посчитать количество сообщений Фактора. Возможные значения:

- **warn** — предупреждения;
- **debug** — сообщения на уровне дебага;
- **error** — сообщения об ошибках;
- **trace** — сообщения на уровне трассировки;
- **info** — информационные сообщения.

Метрики логируем по коду. Т.е. если это будет с точки зрения функционала 1 ошибка, но залогировается в 2 лога, то в метрике отдадим 2 ошибки.

## Метрики обратного потока

jms_messages_error_total	Общее количество ошибок при отправке сообщений с момента старта	23.18
jms_messages_pending	Количество сообщений со статусом PENDING на текущий момент	
jms_messages_sent_seconds_max	Максимальное время успешно отправленного сообщения за период	
	Количество успешно отправленных сообщений с момента старта	

---

jms\_messag      Сумма времени затраченного на все  
es\_sent\_sec      сообщения за период  
onds\_count

jms\_messag  
es\_sent\_sec  
onds\_sum

---

# РУКОВОДСТВО ПО РЕЗЕРВНОМУ КОПИРОВАНИЮ

 Далее по тексту используются [Условные обозначения](#).

## Резервное копирование БД

Резервное копирование БД системы производится стандартными средствами СУБД. Конкретная технология резервного копирования остается на усмотрение администратора СУБД.

Ориентировочный объем БД — 50 Гб на каждый 1 млн контрагентов. Объем и время создания резервной копии БД зависят от выбранного инструмента бэкапирования.

Необходимо бэкапировать схему **{cdi-db}** (табличные пространства **{cdi-db}** и **{cdi-db}\_idx**), включая все объекты и данные.

## Резервное копирование сервера приложений

В состав резервной копии сервера приложений должны входить следующие каталоги со всем содержимым:

### **CDI\_ROOT\_DIR/conf**

- Ориентировочный размер архива: не более 1 Мб.
- Ориентировочное время создания архива: не более 10 сек.

Пример (Linux):

```
$ ./etc/sysconfig/cdi  
$ tar cvzf cdi_conf.tar.gz $CDI_ROOT_DIR/conf
```

Пример (Windows):

```
7z.exe a -tzip cdi_conf.zip %CDI_ROOT_DIR%\conf*
```

### **JBOSS\_CDI\_HOME**

- При архивировании следует исключить содержимое каталогов  
**JBOSS\_CDI\_HOME/standalone/log**  
**JBOSS\_CDI\_HOME/standalone/work**  
**JBOSS\_CDI\_HOME/standalone/tmp**

- Ориентировочный размер архива: не более 500 Мб.
- Ориентировочное время создания архива: не более 1 мин.

Пример (Linux):

```
$ ./etc/sysconfig/cdi
$ tar cvzf jboss_cdi.tar.gz \
--exclude "$JBOSS_CDI_HOME/standalone/log/*" \
--exclude "$JBOSS_CDI_HOME/standalone/tmp" \
--exclude "$JBOSS_CDI_HOME/standalone/work" \
$JBOSS_CDI_HOME
```

Пример (Windows):

```
7z.exe a -tzip ^
-x!%JBOSS_CDI_HOME%\standalone\log\* ^
-x!%JBOSS_CDI_HOME%\standalone\work\* ^
-x!%JBOSS_CDI_HOME%\standalone\tmp\* ^
jboss_cdi.zip %JBOSS_CDI_HOME%/*
```

## **JBOSS\_FACTOR\_HOME**

- При архивировании следует исключить содержимое каталогов
  - JBOSS\_FACTOR\_HOME/standalone/data/dedup-data.ws**
  - JBOSS\_FACTOR\_HOME/standalone/log**
  - JBOSS\_FACTOR\_HOME/standalone/work**
  - JBOSS\_FACTOR\_HOME/standalone/tmp**
- Ориентировочный размер архива: не более 500 Мб.
- Ориентировочное время создания архива: не более 1 мин.

Пример (Linux):

```
$ ./etc/sysconfig/factor
$ tar cvzf jboss_factor.tar.gz \
--exclude "$JBOSS_FACTOR_HOME/standalone/data/dedup-data.ws" \
--exclude "$JBOSS_FACTOR_HOME/standalone/log" \
--exclude "$JBOSS_FACTOR_HOME/standalone/tmp" \
--exclude "$JBOSS_FACTOR_HOME/standalone/work" \
$JBOSS_FACTOR_HOME
```

Пример (Windows):

```
7z.exe a -tzip ^
-x!%JBOSS_FACTOR_HOME/server/default/data/dedup-data.ws\* ^
-x!%JBOSS_FACTOR_HOME%\standalone\log\* ^
-x!%JBOSS_FACTOR_HOME%\standalone\work\* ^
-x!%JBOSS_FACTOR_HOME%\standalone\tmp\* ^
jboss_factor.zip %JBOSS_FACTOR_HOME%/*
```



# РУКОВОДСТВО ПО ВОССТАНОВЛЕНИЮ ПОСЛЕ СБОЯ

1. Условные обозначения
2. Восстановление БД
3. Восстановление сервера приложений
  - a. Восстановление ПО на сервере приложений
  - b. Запуск системы Фактор
  - c. Запуск системы Единый клиент
  - d. Восстановление поискового индекса

# ВОССТАНОВЛЕНИЕ БД

1. Остановите сервис Единого клиента.
2. Восстановите БД системы из резервной копии (схема **{cdi-db}**, табличные пространства **{c di-db}** и **{c di-db}\_idx**).
3. Запустите сервис Единого клиента.
4. Почистить кэш фактора **{JBOSS\_FACTOR\_HOME}jboss/standalone/data/dedup-data.ws**.
5. Запустить сбор статистики в БД (**EXEC DBMS\_STATS.gather\_schema\_stats (ownname => UPPER('ИМЯ\_ПОЛЬЗОВАТЕЛЯ'), cascade => true, estimate\_percent => dbms\_stats.auto\_sample\_size);**)
6. Откройте закладку *Триггеры* в АРМ Администратора и выполните триггер `afterDatabaseBackupTrigger`.
7. Проверьте [доступность БД](#)

# ВОССТАНОВЛЕНИЕ ПО НА СЕРВЕРЕ ПРИЛОЖЕНИЙ

## Связность с другими серверами

Проверьте [связность](#) сервера приложений с другими серверами.

## Восстановление системного и специального ПО

1. Если была нарушена целостность ОС, то после ее переустановки:
  - a. создайте [пользователя JBOSS](#);
  - b. восстановите [системное ПО](#).
2. Если была нарушена целостность ПО Java, переустановите его в соответствии с [инструкцией по установке](#).

## Восстановление прикладного ПО системы

### 1. Если была нарушена целостность ПО JBOSS ФАКТОР

Восстановите соответствующий каталог (**JBOSS\_FACTOR\_HOME**) из резервной копии. Ориентировочное время восстановления каталога из zip/gzip-архива (без учета времени копирования архива на сервер) — не более 1 мин.

Пример (Linux):

```
# ./etc/sysconfig/factor  
# tar xvzf jboss_factor.tar.gz -C /
```

Пример (Windows):

```
mkdir %JBOSS_FACTOR_HOME%  
7z.exe x -o%JBOSS_FACTOR_HOME% jboss_factor.zip
```

### 2. Если была нарушена целостность ПО JBOSS CDI

1. Создайте дополнительные каталоги, которые будут использоваться при работе системы, в соответствии с разделом [Установка системы Единый клиент до 23.17](#) (секция *Дополнительные каталоги*).
2. Восстановите соответствующий каталог (**JBOSS\_CDI\_HOME**) из резервной копии. Ориентировочное время восстановления каталога из zip/gzip-архива (без учета времени копирования архива на сервер) — не более 1 мин.

Пример (Linux):

```
# ./etc/sysconfig/cdi  
# tar xvzf jboss_cdi.tar.gz -C /
```

Пример (Windows):

```
mkdir %JBOSS_CDI_HOME%  
7z.exe x -o%JBOSS_CDI_HOME% jboss_cdi.zip
```

### ***3. Если была нарушена целостность каталога с дополнительными настройками (CDI\_ROOT\_DIR/conf)***

Восстановите его из резервной копии.

Пример (Linux):

```
# ./etc/sysconfig/cdi  
# tar xvzf cdi_conf.tar.gz -C /
```

Пример (Windows):

```
7z.exe x -o%CDI_ROOT_DIR%\conf cdi_conf.zip
```

## ***Очистка базы инкрементальных дубликатов***

Если производилось восстановление БД из резервной копии, удалите базу инкрементальных дубликатов, которая находится в каталоге **JBOSS\_FACTOR\_HOME/server/default/data/dedup-data.ws**.

# ЗАПУСК СИСТЕМЫ ФАКТОР

1. Если была нарушена работа системы Фактор, запустите ее, как описано в разделе [Запуск системы](#).
2. Проверьте, что Фактор корректно стартовал (ожидаемое время старта — 1-5 минут), как описано ниже.

## Сообщение о старте в журнале

В журнале **JBOSS\_FACTOR\_HOME/server/default/server.log** появилась строка вида:

```
... INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel) ... Started in ...
```

Пример проверки (Linux):

```
$ grep '[org.jboss.system.server.Server]' "$JBOSS_FACTOR_HOME/server/default/log/server.log"

2011-12-12 12:24:47,424 INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel) [4.2.3.GA (build: SVNTag=JBoss_4_2_3_GA date=200807181439)] Started in 52s:817ms
```

Пример проверки (Windows):

```
C:\>findstr /L [org.jboss.system.server.Server] "%JBOSS_FACTOR_HOME%\server\default\log\server.log"

2011-12-12 12:24:47,424 INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel) [4.2.3.GA (build: SVNTag=JBoss_4_2_3_GA date=200807181439)] Started in 52s:817ms
```

## Прочие проверки

Выполните все проверки [доступности Фактора](#).

# ЗАПУСК СИСТЕМЫ ЕДИНЫЙ КЛИЕНТ

1. Если была нарушена работа системы Единый клиент, запустите ее, как описано в разделе [Запуск системы](#).
2. Проверьте, что Единый клиент корректно стартовал (ожидаемое время старта — 2-5 минут, кроме случаев, когда [поврежден поисковый индекс](#)), как описано ниже.

## Сообщение о старте в журнале

В журнале **JBOSS\_CDI\_HOME/server/default/server.log** появилась строка вида:

```
... INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel) ... Started in ...
```

Пример проверки (Linux):

```
$ grep '[org\.jboss\.system\.server\.Server\]' "$JBOSS_CDI_HOME/server/default/log/server.log"

2011-12-12 12:24:47,424 INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel)
[4.2.3.GA (build: SVNTag=JBoss_4_2_3_GA date=200807181439)] Star
ted in 52s:817ms
```

Пример проверки (Windows):

```
C:\>findstr /L [org.jboss.system.server.Server] "%JBOSS_CDI_HOME%\server\default\log\server.log"

2011-12-12 12:24:47,424 INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel)
[4.2.3.GA (build: SVNTag=JBoss_4_2_3_GA date=200807181439)] Star
ted in 52s:817ms
```

## Прочие проверки

Выполните все проверки доступности Единого клиента.

## Инкрементальный поиск дубликатов

Если производилось восстановление БД из резервной копии, то после запуска системы необходимо произвести полный поиск дубликатов. В противном случае инкрементальный поиск дубликатов может работать некорректно.

# ВОССТАНОВЛЕНИЕ ПОИСКОВОГО ИНДЕКСА

Как указано в разделе [Запуск системы Единый клиент](#), время старта системы обычно составляет 2-5 минут. Исключением из этого правила являются те случаи, когда при старте обнаруживается, что поврежден поисковый индекс. Поисковый индекс — это каталог в файловой системе на сервере приложений (**CDI\_ROOT\_DIR/search**). Система использует его для отображения списка контрагентов в АРМ Менеджера данных, а также для всех операций поиска, отбора и фильтрации по списку контрагентов. Возможны две ситуации повреждения поискового индекса, они описаны ниже.

## *Поисковый индекс поврежден*

Если поисковый индекс поврежден, то при старте в журнале **JBOSS\_CDI\_HOME/server/default/cdi.log** появится строка вида:

```
... WARN engine.SearchEngine - Index for '...' is corrupted. Need FULL rebuild.
```

Сразу после записи данного сообщения в журнал система начинает полное перестроение индекса. В данном случае старт системы завершится только после того, как поисковый индекс будет успешно перестроен, поэтому время старта будет существенно превышать обычные 2-5 минут. Время перестроения поискового индекса составляет 15-25 минут на каждый 1 млн. контрагентов.

По окончании перестроения поискового индекса в журнале **JBOSS\_CDI\_HOME/server/default/cdi.log** появится строка вида:

```
... INFO engine.EnginesRebuildProcessor - Finish FULL rebuild
```

После чего запуск системы продолжится. Как и при обычном сценарии запуска, он должен закончиться появлением в журнале **JBOSS\_CDI\_HOME/server/default/server.log** строки вида:

```
... INFO [org.jboss.system.server.Server] JBoss (MX MicroKernel) ... Started in ...
```

## *Поисковый индекс неактуален*

Если при старте система обнаруживает, что количество контрагентов в поисковом индексе не соответствует количеству контрагентов в БД (что косвенно свидетельствует о неактуальности индекса), то при старте в журнале **JBOSS\_CDI\_HOME/server/default/cdi.log** появится строка вида:

```
... WARN engine.SearchEngineValidator - '...' count in index (...) is not equal to actual count (...). Index may be corrupted. Please rebuild ASAP.
```

В данной ситуации система не будет самостоятельно перестраивать поисковый индекс, и продолжит запуск как в обычном сценарии. Однако, администратору системы рекомендуется перестроить поисковый индекс при первой удобной возможности (когда нагрузка на систему

не слишком большая). Для перестроения поискового индекса можно воспользоваться задачей **enginesFullRebuild**, которую можно запустить через [APM Администратора](#).



# SOAP И REST (ОПИСАНИЕ ОНЛАЙН-ИНТЕРФЕЙСОВ)

- Объектная модель онлайн-интерфейсов
  - Примеры — объектная модель (онлайн)
  - Элементы — объектная модель (онлайн)
  - Общее описание объектной модели
- PartyWS и PartyRA
  - Party.save — Создание нового контрагента или обновление существующего.
  - Party.saveAndMerge — Сохранить или обновить одного и более контрагентов с проверкой на дубликаты и слиянием.
  - Party.getByRawID — Метод позволяет получить контрагента по идентификатору исходной клиентской записи.
  - Party.getByHID — Метод позволяет получить контрагента по его идентификатору в Едином клиенте.
  - Party.search — Метод позволяет найти всех контрагентов, удовлетворяющих заданным критериям.
  - Party.fuzzySearch — Метод позволяет найти всех контрагентов, являющихся дубликатами для заданной записи.
  - Party.findInFns — Метод позволяет получить ИНН от ФНС по набору ПДн без сохранения карточки в CDI ЕК (кредитный конвейер, скоринг и т.д.).
  - Party.searchRelatedParties — Метод позволяет найти связанных контрагентов.
  - Party.checkBlacklist — Метод позволяет проверить контрагента по черным спискам и перечню недействительных паспортов.
  - Party.checkBlacklistByRawID — Метод проверяет контрагента по черным спискам по rawId и коду исходной системы и сразу создает связь между искомым КА и записью ЧС.
  - Party.checkBlackListByHid — Метод проверяет контрагента по черным спискам по хиду и сразу создает связь между искомым КА и записью ЧС.
  - Party.close — Метод позволяет закрыть контрагента по его идентификатору в Едином Клиенте (HID) или по идентификатору в исходной системе (RawId).
  - Party.closeAttribute — Закрыть атрибут клиента по его hid.
  - Party.closeAttributeByRawId — Метод позволяет закрыть атрибут клиента по его rawId.
  - Party.getReference — Метод позволяет получить все значения из справочника Классификаторов <http://confluence.hflabs.ru/pages/viewpage.action?pageId=19433287> по выбранному классификатору.
  - Party.saveRelations — Метод позволяет создавать и закрывать взаимосвязи между объектами.
  - Party.getRelations — Метод позволяет получить взаимосвязи объекта с другими объектами по идентификатору в «Едином клиенте». Возвращает связи между золотыми контрагентами.
  - Party.getSourceAttributes — Метод позволяет получить по идентификатору атрибута в Едином клиенте информацию об атрибутах, из которых он был составлен .

Party.mergeSource — Устарел, больше не поддерживается.

PartyWS и PartyRA (WSDL, WADL и схемы)

- TaskWS и TaskRA

Task.execute — Метод позволяет запустить задачу с указанием ее параметров.

Task.status — Метод позволяет получить статус запущенной задачи по ее идентификатору.

TaskWS и TaskRA (WSDL и схемы)

- CleanService

CleanService.address — Стандартизация адреса

CleanService.email — Стандартизация адреса электронной почты

CleanService.legal — Стандартизация реквизитов контрагента юридического лица

CleanService.passport — Стандартизация ДУЛ

CleanService.phone — Стандартизация телефона

CleanService.physical — Стандартизация реквизитов контрагента

CleanService.soap-name-declension — Падежные формы ФИО

- Безопасность

- Особенности сервисов

# ОБЪЕКТНАЯ МОДЕЛЬ ОНЛАЙН-ИНТЕРФЕЙСОВ

Объектная модель онлайн-интерфейсов использует 3 основных типа:

1. **WParty** — контрагент;
2. **WAttribute** — составной атрибут контрагента, который может состоять из нескольких полей — например, адрес, телефон;
3. **WField** — отдельное поле контрагента или атрибута.

## *Контрагенты и атрибуты*

Каждый контрагент может содержать произвольное количество полей и атрибутов. Каждый атрибут, в свою очередь, может содержать произвольное количество полей.

Каждое поле идентифицируется именем. Перечень разрешенных имен полей и их типы определяется конфигурацией заказчика. Общие для всех заказчиков поля описаны ниже по тексту.

Для контрагентов и атрибутов обязательно указание их типа (**//party@type** и **//party/attribute@type**). Список разрешенных типов указан в разделе [Классификаторы](#), секции [Тип контрагента](#) и [Тип атрибута](#) соответственно.

## *Идентификаторы*

Каждый контрагент содержит уникальный идентификатор **NID** (**//party@hid**). Также исходные контрагенты (контрагенты, которые не были ни с кем объединены) содержат исходный идентификатор — пару **//party@sourceSystem** + **//party@rawId**.

Атрибуты также имеют уникальные идентификаторы — **//party/attribute@hid**, и помимо это дополнительно содержат исходный идентификатор — **//party/attribute@rawId**.

## *Исходные и объединенные контрагенты*

Объединенные контрагенты (контрагенты, образовавшиеся в результате слияния двух или более исходных контрагентов) хранят информацию о привязке к своим исходным контрагентам в элементах вида **//party/source**. Каждый **//party/source** соответствует одному исходному контрагенту, и содержит его идентификаторы.

У объединенных контрагентов, в отличие от исходных, отсутствуют **//party@sourceSystem** и **/party@rawId**.

*Примеры — объектная модель (онлайн)*

*Элементы — объектная модель (онлайн)*

*Общее описание объектной модели*

# ПРИМЕРЫ — ОБЪЕКТНАЯ МОДЕЛЬ (ОНЛАЙН)

- Исходный контрагент
- Объединенный контрагент

## *Исходный контрагент*

```
<party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1" xmlns="http://hflabs.ru/cdi/soap/2_12">
  <field name="actualityDate">22.02.2012</field>
  <field name="surname">МАМОНОВ</field>
  <field name="name">СЕРГЕЙ</field>
  <field name="patronymic">ВИКТОРОВИЧ</field>
  <field name="fullNameAuthor">AL:1</field>
  <field name="fullNameQC">EDITED</field>
  <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
  <field name="birthdate">03.12.1987</field>
  <field name="birthdateAuthor">AL:1</field>
  <field name="birthdateQC">EDITED</field>
  <field name="birthdateRawSource">03.12.1987</field>
  <attribute type="ADDRESS" hid="1024" rawId="AL.1.ADDR.1">
    <field name="primary">true</field>
    <field name="actualityDate">12.01.2012</field>
    <field name="type">HOME</field>
    <field name="postalCode">129090</field>
    <field name="kladrPostalCode">129090</field>
    <field name="country">РОССИЯ</field>
    <field name="regionType">Г</field>
    <field name="region">МОСКВА</field>
    <field name="rayonType"/>
    <field name="rayon"/>
    <field name="cityType">Г</field>
    <field name="city">МОСКВА</field>
    <field name="settlementType"/>
    <field name="settlement"/>
    <field name="streetType">ПП-КТ</field>
    <field name="street">МИРА</field>
    <field name="houseNumber">22</field>
    <field name="militaryNumber"/>
    <field name="postalBox"/>
    <field name="korpus"/>
    <field name="stroenie"/>
    <field name="vladenie"/>
    <field name="hostel"/>
    <field name="floor"/>
    <field name="entrance"/>
    <field name="section"/>
    <field name="flat">4</field>
    <field name="flat2"/>
    <field name="office"/>
    <field name="office2"/>
    <field name="room"/>
    <field name="room2"/>
    <field name="kladrCode">7700000000018560125</field>
    <field name="okatoCode">45286570000</field>
  </attribute>
</party>
```

```

<field name="validationExtension"/>
<field name="author">AL:1</field>
<field name="qualityCode">GOOD</field>
<field name="validationCode">VALIDATED</field>
<field name="rawSource">,,Москва,,,,, Проспект мира,22,,,4</field>
<field name="additionalInfo"/>
</attribute>
<attribute type="DOCUMENT_PASSPORT" hid="1024" rawId="AL.1.DOC.1">
  <field name="primary">true</field>
  <field name="actualityDate">23.08.2011</field>
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="documentSeries">45 07</field>
  <field name="documentNumber">672184</field>
  <field name="issueDate">15.06.2007</field>
  <field name="issueAuthority">ОВД Хамовники гор. Москвы</field>
  <field name="departmentCode">662-533</field>
  <field name="expiryDate"/>
  <field name="author">AL:1</field>
  <field name="invalidStatus">ACTUAL</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource">4509 672184</field>
  <field name="additionalInfo"/>
</attribute>
<attribute type="EMAIL" hid="1024" rawId="AL.2.EM.1">
  <field name="primary">false</field>
  <field name="actualityDate">23.08.2011</field>
  <field name="type">WORK</field>
  <field name="email">MAMON@GAZPROM.RU</field>
  <field name="author">AL:1</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource">mamon@gazprom.ru</field>
  <field name="additionalInfo"/>
</attribute>
<attribute type="PHONE" hid="1024" rawId="AL.1.PH.1">
  <field name="primary">false</field>
  <field name="actualityDate">23.08.2011</field>
  <field name="type">HOME</field>
  <field name="countryCode">7</field>
  <field name="cityCode">495</field>
  <field name="number">3542344</field>
  <field name="extension"/>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
  <field name="qualityCode">GOOD</field>
  <field name="rawSource">8-495-354-23-44</field>
  <field name="additionalInfo"/>
</attribute>
</party>

```

## Объединенный контрагент

```

<party type="PHYSICAL" hid="1026" xmlns="http://hflabs.ru/cdi/soap/2_12">
  <field name="actualityDate">22.02.2012</field>
  <field name="surname">КУРКО</field>
  <field name="name">ОЛЕГ</field>
  <field name="patronymic">ИГОРЕВИЧ</field>
  <field name="fullNameAuthor">AL:2</field>
  <field name="fullNameQC">EDITED</field>
  <field name="fullNameRawSource">Курко Олег Игоревич</field>

```

```
<field name="birthdate"/>
<field name="birthdateAuthor">AL:2</field>
<field name="birthdateQC">NOT_SURE</field>
<field name="birthdateRawSource"/>
<field name="maritalStatus">SINGLE_MALE</field>
<field name="maritalStatusAuthor">AL:2</field>
<attribute type="PHONE" hid="1025" rawId="AL.2.PH.1">
  <field name="actualityDate"/>
  <field name="type">MOBILE</field>
  <field name="countryCode">7</field>
  <field name="cityCode">916</field>
  <field name="number">5345622</field>
  <field name="extension"/>
  <field name="primary"/>
  <field name="additionalInfo"/>
  <field name="author">AL:2</field>
  <field name="qualityCode">GOOD</field>
  <field name="rawSource">9165345622</field>
</attribute>
<attribute type="PHONE" hid="1026" rawId="BT.3.PH.1">
  <field name="actualityDate"/>
  <field name="type">HOME</field>
  <field name="countryCode">7</field>
  <field name="cityCode">495</field>
  <field name="number">2283412</field>
  <field name="extension"/>
  <field name="primary"/>
  <field name="additionalInfo"/>
  <field name="author">BT:3</field>
  <field name="qualityCode">GOOD</field>
  <field name="rawSource">4952283412</field>
</attribute>
<attribute type="DOCUMENT_PASSPORT" hid="1025" rawId="AL.2.DOC.1">
  <field name="actualityDate"/>
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="documentSeries">45 09</field>
  <field name="documentNumber">672184</field>
  <field name="issueDate"/>
  <field name="issueAuthority"/>
  <field name="departmentCode"/>
  <field name="expiryDate"/>
  <field name="invalidStatus">ACTUAL</field>
  <field name="author">AL:2</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource">4509 672184</field>
</attribute>
<source hid="1024" sourceSystem="AL" rawId="2"/>
<source hid="1025" sourceSystem="BT" rawId="3"/>
</party>
```

# ЭЛЕМЕНТЫ — ОБЪЕКТНАЯ МОДЕЛЬ (ОНЛАЙН)

- Атрибуты контрагента

Адрес

Документ, удостоверяющий личность

Документ юридического лица (DOCUMENT\_LEGAL\_PARTY)

Лицензия (DOCUMENT\_LICENSE)

Лицензия юридического лица (CERTIFICATE)

Документ субъектов малого и среднего предпринимательства (DSMP)

Телефон

Адрес электронной почты

Аккаунт

Согласие

- Прочие элементы

Классификатор (WReference)

Значение классификатора (WReferenceEntry)

Контрагент-дубликат (WMatchedParty)

Взаимосвязь (WRelation)

Связанный объект (relatedEssences)

Тип информации о контрагенте

Тип обновления контрагента




## Атрибуты контрагента

### Адрес

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного адреса
actualityDate	нет	string	Дата актуальности
type	нет	Тип адреса (литера)	Тип адреса
postalCode	нет	string	Почтовый индекс
kladrPostalCode	нет	string	Почтовый индекс по КЛАДР
country	нет	string	Наименование страны
district	да	string	Федеральный округ



regionType	нет	string	Тип региона
region	нет	string	Наименование региона
rayonType	нет	string	Тип района в регионе
rayon	нет	string	Наименование района в регионе
cityType	нет	string	Тип города
city	нет	string	Наименование города
cityAreaType	нет	string	Тип района города
cityArea	нет	string	Район города
settlementType	нет	string	Тип населенного пункта
settlement	нет	string	Наименование населенного пункта
streetType	нет	string	Тип улицы
street	нет	string	Наименование улицы
houseNumber	нет	string	Номер дома (включает номер, литеру, дробь, литеру дробки)
militaryNumber	нет	string	Номер военной части
postalBox	нет	string	Номер абонентского ящика
korpus	нет	string	Корпус (включает номер, литеру, дробь, литеру дробки)
stroenie	нет	string	Строение (включает номер, литеру, дробь, литеру дробки)
vladenie	нет	string	Владение (включает номер, литеру, дробь, литеру дробки)
hostel	нет	string	Общежитие (номер или просто указание на то, что это общежитие)
floor	нет	string	Этаж
entrance	нет	string	Номер подъезда
section	нет	string	Секция
flat	нет	string	Первая квартира (включает номер, литеру, дробь, литеру дробки)
flat2	нет	string	Вторая квартира (включает номер, литеру, дробь, литеру дробки)
office	нет	string	Первый офис (включает номер, литеру, дробь, литеру дробки)
office2	нет	string	Второй офис (включает номер, литеру, дробь, литеру дробки)
room	нет	string	Первая комната (включает номер, литеру, дробь, литеру дробки)
room2	нет	string	Вторая комната (включает номер, литеру, дробь, литеру дробки)
kladrCode	нет	string	Код КЛАДР адреса
okatoCode	нет	string	Код ОКАТО
validationExtension	да	string	Расширение валидатора (сомнительная часть адреса)
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код полноты адреса	Код полноты адреса

		(литера)	
validationCode	да	Код проверки адреса (литера)	Код проверки адреса
rawSource	нет	string	Исходное значение адреса (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)
kladrActualityDate	да	string	Дата актуальности КЛАДР
fiasCode	да	string	Идентификатор ФИАС
fiasLevel	да	Код соответствия ФИАС (литера)	Уровень ФИАС
popularity	да	Популярность атрибута (литера)	Популярность
oksmCode	да	string	Код по общероссийскому классификатору стран мира (ОКСМ)  Подключается по запросу. Если у вас нет данного поля, но вы хотите его добавить, напишите, пожалуйста, нам в поддержку
oktmoCode	да	string	Код по общероссийскому классификатору территорий муниципальных образований (ОКТМО)  Подключается по запросу. Если у вас нет данного поля, но вы хотите его добавить, напишите, пожалуйста, нам в поддержку
timezone	да	Часовые пояса России (литера)	Часовой пояс, в котором находится адрес  Подключается по запросу. Если у вас нет данного поля, но вы хотите его добавить, напишите, пожалуйста, нам в поддержку

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## Документ, удостоверяющий личность

Поле	Readonly ? *	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
qualifierType	нет	Тип ДУЛ (литера)	Тип документа, удостоверяющего личность
type	да	Тип документа (литера)	Тип документа
documentSeries	нет	string	Серия документа
documentNumber	нет	string	Номер документа
issueDate	нет	string	Дата выдачи документа
issueAuthority	нет	string	Орган, выдавший документ
departmentCode	нет	string	Код подразделения органа, выдавшего документ

expiryDate	нет	string	Дата истечения срока действия
additionalInfo	нет	string	Дополнительная информация (примечание)
invalidStatus	да	Статус недействительности для документа (литера)	Статус недействительности для документа
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества документа (литера)	Код качества документа
documentNSReasons	да	string	Причины некорректности документа
rawSource	да	string	Исходное значение документа (одной строкой)
popularity	да	Код популярности (литера)	Популярность

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## **Документ юридического лица (DOCUMENT\_LEGAL\_PARTY)**

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
qualifierType	нет	Тип документа ЮЛ (литера)	Тип документа юридического лица
documentSeries	нет	string	Серия документа
documentNumber	нет	string	Номер документа
registrationDate	нет	string	Дата регистрации
issueAuthority	нет	string	Орган, выдавший документ
additionalInfo	нет	string	Дополнительная информация (примечание)
expiryDate	нет	string	Дата истечения срока действия
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества документа (литера)	Код качества документа
rawSource	да	string	Исходное значение документа (одной строкой)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## **Лицензия (DOCUMENT\_LICENSE)**

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности

qualifierType	нет	Тип лицензии (литера)	Тип лицензии
documentSeries	нет	string	Серия лицензии
documentNumber	нет	string	Номер лицензии
issueDate	нет	string	Дата выдачи
validityPeriod	нет	string	Срок действия
licensedActivity	нет	string	Лицензируемая деятельность
issueAuthority	нет	string	Орган, выдавший документ
additionalInfo	нет	string	Дополнительная информация (примечание)
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества документа (литера)	Код качества лицензии
rawSource	да	string	Исходное значение лицензии (одной строкой)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## Лицензия юридического лица (CERTIFICATE)

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основной лицензии
actualityDate	нет	string	Дата актуальности
series	нет	string	Серия лицензии
number	нет	string	Номер лицензии
issueDate	нет	string	Дата выдачи лицензии
issueAuthority	нет	string	Орган, выдавший лицензию
suspendDate	нет	string	Дата приостановления действия лицензии
suspendAuthority	нет	string	Орган, приостановивший действие лицензии
activityDateBegin	нет	string	Дата начала действия лицензии
activityDateEnd	нет	string	Дата окончания действия лицензии
activityValue	нет	string	Наименование деятельности, на которую рассчитана лицензия
activityPlace	нет	string	Место осуществления деятельности
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества (литера)	Код качества лицензии
rawSource	нет	string	Исходное значение лицензии (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## Документ субъектов малого и среднего предпринимательства (DSMP)

Поле	Readonly?*	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
type	нет	Категория предприятия документа МСП (литера)	Категория предприятия
dateFrom	нет	string	Дата включения в реестр
dateTo	нет	string	Дата исключения из реестра
author	да	string	Автор версии
rawSource	нет	string	Исходное значение документа МСП (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## Телефон

Поле	Readonly?*	Тип	Описание
primary	нет	boolean	Признак основного телефона
actualityDate	нет	string	Дата актуальности
type	нет	Тип телефона (литера)	Тип телефона
operator	да	string	Оператор связи
operatorAddress	да	string	Район/город/НП, определенный по номеру телефона
operatorRegion	да	string	Регион, определенный по номеру телефона
countryCode	нет	string	Код страны
cityCode	нет	string	Код города
number	нет	string	Номер телефона
extension	нет	string	Добавочный номер
author	да	string	Ссылка на источник атрибута
qualityCode	да	string	Код качества телефона
rawSource	нет	string	Исходное значение телефона (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)
timezone	да	Часовой пояс (литера)	Часовой пояс
numberProfile	да	Характеристика номера телефона (литера)	Характеристика номера телефона

Поле	Readonly? *	Тип	Описание
popularity	да	Код популярности (литера)	Популярность

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## Адрес электронной почты

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного email
actualityDate	нет	string	Дата актуальности
type	нет	Тип email (литера)	Тип email
email	нет	string	Значение email
author	да	string	Автор версии
qualityCode	да	Код качества реквизита (литера)	Код качества email
rawSource	нет	string	Исходное значение email
additionalInfo	нет	string	Дополнительная информация (примечание)
popularity	да	Код популярности (литера)	Популярность

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

## Аккаунт

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного аккаунта
actualityDate	нет	string	Дата актуальности
type	нет	Тип аккаунта	Тип аккаунта (литера)
account	нет	string	Значение аккаунта
author	да	string	Автор версии
qualityCode	да	Код качества реквизита (литера)	Код качества аккаунта
rawSource	нет	string	Исходное значение аккаунта
additionalInfo	нет	string	Дополнительная информация (примечание)

## Согласие

Поле	Readonly? *	Тип	Описание
actualityDate	нет	string	Дата актуальности
type	нет	Тип согласия	Тип согласия (литера)

consentValue	нет	Значение согласия	Значение согласия (литера)
rawSource	да	string	Исходное значение согласия
primary	да	boolean	Признак основного согласия (всегда FALSE)
qualityCode	да	string	Код качества согласия
author	да	string	Автор версии
additionalInfo	да	string	Дополнительная информация (примечание)

\* Признак, можно ли указывать поле в запросе, или оно фигурирует только в ответе.

## Прочие элементы

### Классификатор (*WReference*)

Поле	Тип	Описание
type	WReferenceType	Тип классификатора
referenceEntry	WReferenceEntry[]	Значения классификатора

### Значение классификатора (*WReferenceEntry*)

Поле	Тип	Описание
partyType	string	Тип контрагента. Определяет для какого типа контрагентов доступно данное значение. Если поле не заполнено, то данное значение доступно всем типам контрагентов.
code	string	Литера значения в классификаторе.
actual	boolean	Признак актуальности значения.
ordinal	int	Код значения в классификаторе.
label	string	Описание в классификаторе.
description	string	Дополнительное расширенное описание в классификаторе.

### Контрагент-дубликат (*WMatchedParty*)

Поле	Тип	Описание
matchRule	string	Номер правила идентификации дубликатов, по которому был найден контрагент-дубликат

matchScore	int	Коэффициент соответствия для правила идентификации дубликатов, по которому был найден контрагент-дубликат
party	WParty	Информация о контрагенте-дубликате.

## Взаимосвязь (WRelation)

Имя параметра	Тип	Обязательный	Значение по умолчанию	Описание
type	Код связи	Да	—	Тип взаимосвязи между объектами
first	Связанный объект	Нет	—	Первый объект (сторона А связи)
second	Связанный объект	Нет	—	Второй объект (сторона Б связи)
deleted	boolean	Нет	False	Признак закрытия связи
hashcode	string	Нет	—	Контрольная сумма

## Связанный объект (relatedEssences)

Имя параметра	Тип	Обязательный	Описание
type	WPartyType	Да	Тип объекта
hid	long	Нет	Уникальный идентификатор объекта в Едином клиенте (HID)
sourceSystem	string	Нет	Исходная система связываемого объекта
rawId	string	Нет	Идентификатор связываемого объекта в исходной системе

## Тип информации о контрагенте

Литера	Описание
REQUISITE	Реквизиты
ATTRIBUTE	Атрибуты
SUBATTRIBUTE	Податрибуты. Работает только вместе с ATTRIBUTE.
SOURCE	Для объединенного контрагента информация об исходных контрагентах.
RELATION	Взаимосвязи
PAST_ATTRIBUTE	Прошлые значения



## Тип обновления контрагента

Литера	Описание
PARTIAL	Частичное. В запросе передаются только изменившиеся данные. <b>Обратите внимание:</b> значения для полей, которые не были переданы в запросе, останутся без изменений – то есть останется то значение, которое было установлено ранее. Если необходимо сбросить значение какого-либо поля в «пусто», то есть очистить значения поля, необходимо в запросе явно указать данное поле с пустым значением.  Для атрибутов и связанных полей должны передавать все поля атрибута и вся группа связанных полей. То есть нельзя передать только изменение номера паспорта без остальных полей. Также нельзя передать изменение только имени клиента без передачи полей фамилии и отчества.
FULL	Полное. В запросе передается полная карточка контрагента

# АДРЕС

## Адрес

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного адреса
actualityDate	нет	string	Дата актуальности
type	нет	Тип адреса (литера)	Тип адреса
postalCode	нет	string	Почтовый индекс
kladrPostalCode	нет	string	Почтовый индекс по КЛАДР
country	нет	string	Наименование страны
district	да	string	Федеральный округ
regionType	нет	string	Тип региона
region	нет	string	Наименование региона
rayonType	нет	string	Тип района в регионе
rayon	нет	string	Наименование района в регионе
cityType	нет	string	Тип города
city	нет	string	Наименование города
cityAreaType	нет	string	Тип района города
cityArea	нет	string	Район города
settlementType	нет	string	Тип населенного пункта
settlement	нет	string	Наименование населенного пункта
streetType	нет	string	Тип улицы
street	нет	string	Наименование улицы
houseNumber	нет	string	Номер дома (включает номер, литеру, дробь, литеру дробь)
militaryNumber	нет	string	Номер военной части
postalBox	нет	string	Номер абонентского ящика
korpus	нет	string	Корпус (включает номер, литеру, дробь, литеру дробь)
stroenie	нет	string	Строение (включает номер, литеру, дробь, литеру дробь)
vladenie	нет	string	Владение (включает номер, литеру, дробь, литеру дробь)
hostel	нет	string	Общежитие (номер или просто указание на то, что это общежитие)
floor	нет	string	Этаж

entrance	нет	string	Номер подъезда
section	нет	string	Секция
flat	нет	string	Первая квартира (включает номер, литеру, дробь, литеру дроби)
flat2	нет	string	Вторая квартира (включает номер, литеру, дробь, литеру дроби)
office	нет	string	Первый офис (включает номер, литеру, дробь, литеру дроби)
office2	нет	string	Второй офис (включает номер, литеру, дробь, литеру дроби)
room	нет	string	Первая комната (включает номер, литеру, дробь, литеру дроби)
room2	нет	string	Вторая комната (включает номер, литеру, дробь, литеру дроби)
kladrCode	нет	string	Код КЛАДР адреса
okatoCode	нет	string	Код ОКАТО
validationExtension	да	string	Расширение валидатора (сомнительная часть адреса)
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код полноты адреса (литера)	Код полноты адреса
validationCode	да	Код проверки адреса (литера)	Код проверки адреса
rawSource	нет	string	Исходное значение адреса (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)
kladrActualityDate	да	string	Дата актуальности КЛАДР
fiasCode	да	string	Идентификатор ФИАС
fiasLevel	да	Код соответствия ФИАС (литера)	Уровень ФИАС
popularity	да	Популярность атрибута (литера)	Популярность
oksmCode	да	string	Код по общероссийскому классификатору стран мира (ОКСМ) ⚠ Подключается по запросу. Если у вас нет данного поля, но вы хотите его добавить, напишите, пожалуйста, нам в поддержку
oktmoCode	да	string	Код по общероссийскому классификатору территорий муниципальных образований (ОКТМО) ⚠ Подключается по запросу. Если у вас нет данного поля, но вы хотите его добавить, напишите, пожалуйста, нам в поддержку
timezone	да	Часовые пояса России (литера)	Часовой пояс, в котором находится адрес ⚠ Подключается по запросу. Если у вас нет данного поля, но вы хотите его добавить, напишите, пожалуйста, нам в поддержку

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# АДРЕС ЭЛЕКТРОННОЙ ПОЧТЫ

## *Адрес электронной почты*

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного email
actualityDate	нет	string	Дата актуальности
type	нет	Тип email (литера)	Тип email
email	нет	string	Значение email
author	да	string	Автор версии
qualityCode	да	Код качества реквизита (литера)	Код качества email
rawSource	нет	string	Исходное значение email
additionalInfo	нет	string	Дополнительная информация (примечание)
popularity	да	Код популярности (литера)	Популярность

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# АККАУНТ

## Аккаунт

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного аккаунта
actualityDate	нет	string	Дата актуальности
type	нет	Тип аккаунта	Тип аккаунта (литера)
account	нет	string	Значение аккаунта
author	да	string	Автор версии
qualityCode	да	Код качества реквизита (литера)	Код качества аккаунта
rawSource	нет	string	Исходное значение аккаунта
additionalInfo	нет	string	Дополнительная информация (примечание)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# ВЗАИМОСВЯЗЬ

## Взаимосвязь (*WRelation*)

Имя параметра	Тип	Обязательный	Значение по умолчанию	Описание
type	Код связи	Да	—	Тип взаимосвязи между объектами
first	Связанный объект	Нет	—	Первый объект (сторона А связи)
second	Связанный объект	Нет	—	Второй объект (сторона Б связи)
deleted	boolean	Нет	False	Признак закрытия связи
hashcode	string	Нет	—	Контрольная сумма

## Связанный объект (*relatedEssences*)

Имя параметра	Тип	Обязательный	Описание
type	WPartyType	Да	Тип объекта
hid	long	Нет	Уникальный идентификатор объекта в Едином клиенте (HID)
sourceSystem	string	Нет	Исходная система связываемого объекта
rawId	string	Нет	Идентификатор связываемого объекта в исходной системе

# ВЗАИМОСВЯЗЬ (WRELATION)

## *Взаимосвязь (WRelation)*

Имя параметра	Тип	Обязательный	Значение по умолчанию	Описание
type	Код связи	Да	—	Тип взаимосвязи между объектами
first	Связанный объект	Нет	—	Первый объект (сторона А связи)
second	Связанный объект	Нет	—	Второй объект (сторона Б связи)
deleted	boolean	Нет	False	Признак закрытия связи
hashcode	string	Нет	—	Контрольная сумма

# СВЯЗАННЫЙ ОБЪЕКТ

## *Связанный объект (relatedEssences)*

Имя параметра	Тип	Обязательный	Описание
type	WPartyType	Да	Тип объекта
hid	long	Нет	Уникальный идентификатор объекта в Едином клиенте (HID)
sourceSystem	string	Нет	Исходная система связываемого объекта
rawId	string	Нет	Идентификатор связываемого объекта в исходной системе



# ДОКУМЕНТ, УДОСТОВЕРЯЮЩИЙ ЛИЧНОСТЬ

## Документ, удостоверяющий личность

Поле	Readonly ? *	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
qualifierType	нет	Тип ДУЛ (литера)	Тип документа, удостоверяющего личность
type	да	Тип документа (литера)	Тип документа
documentSeries	нет	string	Серия документа
documentNumber	нет	string	Номер документа
issueDate	нет	string	Дата выдачи документа
issueAuthority	нет	string	Орган, выдавший документ
departmentCode	нет	string	Код подразделения органа, выдавшего документ
expiryDate	нет	string	Дата истечения срока действия
additionalInfo	нет	string	Дополнительная информация (примечание)
invalidStatus	да	Статус недействительности для документа (литера)	Статус недействительности для документа
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества документа (литера)	Код качества документа
documentNSReasons	да	string	Причины некорректности документа
rawSource	да	string	Исходное значение документа (одной строкой)
popularity	да	Код популярности (литера)	Популярность

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# ДОКУМЕНТ МСП

## Документ субъектов малого и среднего предпринимательства (DSMP)

Поле	Readonly?*	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
type	нет	Категория предприятия документа МСП (литера)	Категория предприятия
dateFrom	нет	string	Дата включения в реестр
dateTo	нет	string	Дата исключения из реестра
author	да	string	Автор версии
rawSource	нет	string	Исходное значение документа МСП (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# ДОКУМЕНТ ЮРИДИЧЕСКОГО ЛИЦА

## Документ юридического лица (DOCUMENT\_LEGAL\_PARTY)

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
qualifierType	нет	Тип документа ЮЛ (литера)	Тип документа юридического лица
documentSeries	нет	string	Серия документа
documentNumber	нет	string	Номер документа
registrationDate	нет	string	Дата регистрации
issueAuthority	нет	string	Орган, выдавший документ
additionalInfo	нет	string	Дополнительная информация (примечание)
expiryDate	нет	string	Дата истечения срока действия
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества документа (литера)	Код качества документа
rawSource	да	string	Исходное значение документа (одной строкой)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# КЛАССИФИКАТОР

## Классификатор (*WReference*)

Поле	Тип	Описание
type	<a href="#">WReferenceType</a>	Тип классификатора
referenceEntry	<a href="#">WReferenceEntry[]</a>	Значения классификатора

## Значение классификатора (*WReferenceEntry*)

Поле	Тип	Описание
partyType	string	Тип контрагента. Определяет для какого типа контрагентов доступно данное значение. Если поле не заполнено, то данное значение доступно всем типам контрагентов.
code	string	Литера значения в классификаторе.
actual	boolean	Признак актуальности значения.
ordinal	int	Код значения в классификаторе.
label	string	Описание в классификаторе.
description	string	Дополнительное расширенное описание в классификаторе.

# КОНТРАГЕНТ-ДУБЛИКАТ

## *Контрагент-дубликат (WMatchedParty)*

Поле	Тип	Описание
matchRule	string	Номер правила идентификации дубликатов, по которому был найден контрагент-дубликат
matchScore	int	Коэффициент соответствия для правила идентификации дубликатов, по которому был найден контрагент-дубликат
party	WParty	Информация о контрагенте-дубликате.

# КОНТРАГЕНТ ИЗ ЧЕРНОГО СПИСКА

Поле	Readonly? *	Тип	Описание
blacklistId	Да	string	Идентификатор черносписочника
blacklistName	Да	string	Название черного списка
matchScope	Да	string	Коэффициент соответствия для правила идентификации в ЧС, по которому было найдено совпадение с ЧС
rule	Да	string	Номер правила идентификации в ЧС, по которому было найдено совпадение с ЧС
surname	Да	string	Фамилия
patronymic	Да	string	Имя
name	Да	string	Отчество
nameRawSource	Да	string	Исходные ФИО или наименование ЮЛ одной строкой.
birthdate	Да	string	Дата рождения
birthPlace	Да	string	Место рождения
inn	Да	string	ИНН
documentNumber	Да	string	Номер документа или ОГРН для юридического лица
bankruptcyDate	Да	string	Дата банкротства
bankruptcyExitDate	Да	string	Дата выхода из банкротства
bankruptcyStatus	Да	string	Стадия банкротства
actualityDate	Да	string	Дата актуальности

Для каждого из реквизитов клиента, перечисленных в данной таблице, в SOAP-ответах также возвращается поле вида **{fieldName}Author** (например, `fullNameAuthor`, `innAuthor`, `birthdateAuthor`). Это поле содержит ссылку на запись в исходном списке, из которого получено значение соответствующего реквизита. Например, **birthdateAuthor = TERRORIST\_CB:2525** говорит о том, что значение даты рождения получено из записи с идентификатором 12345 списка экстремистов ЦБ.

Контрагенты из черного списка могут иметь [Адреса](#).

Получить контрагента из черного списка можно методом `checkBlacklist`

# ЛИЦЕНЗИЯ

## Лицензия (*DOCUMENT\_LICENSE*)

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного документа
actualityDate	нет	string	Дата актуальности
qualifierType	нет	Тип лицензии (литера)	Тип лицензии
documentSeries	нет	string	Серия лицензии
documentNumber	нет	string	Номер лицензии
issueDate	нет	string	Дата выдачи
validityPeriod	нет	string	Срок действия
licensedActivity	нет	string	Лицензируемая деятельность
issueAuthority	нет	string	Орган, выдавший документ
additionalInfo	нет	string	Дополнительная информация (примечание)
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества документа (литера)	Код качества лицензии
rawSource	да	string	Исходное значение лицензии (одной строкой)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.



# ЛИЦЕНЗИЯ ЮРИДИЧЕСКОГО ЛИЦА

## Лицензия юридического лица (CERTIFICATE)

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основной лицензии
actualityDate	нет	string	Дата актуальности
series	нет	string	Серия лицензии
number	нет	string	Номер лицензии
issueDate	нет	string	Дата выдачи лицензии
issueAuthority	нет	string	Орган, выдавший лицензию
suspendDate	нет	string	Дата приостановления действия лицензии
suspendAuthority	нет	string	Орган, приостановивший действие лицензии
activityDateBegin	нет	string	Дата начала действия лицензии
activityDateEnd	нет	string	Дата окончания действия лицензии
activityValue	нет	string	Наименование деятельности, на которую рассчитана лицензия
activityPlace	нет	string	Место осуществления деятельности
author	да	string	Ссылка на источник атрибута
qualityCode	да	Код качества (литера)	Код качества лицензии
rawSource	нет	string	Исходное значение лицензии (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# ОКВЭД

## ОКВЭД

Атрибут ОКВЭД — информация для ЮЛ/ИП из общероссийского классификатора видов экономической деятельности.

Поле	Тип/Справочник	Описание
type	Тип справочника ОКВЭД (Литера)	Тип справочника ОКВЭД
name	String	Название
code	String	Код ОКВЭД
qualityCode	Код Качества ОКВЭД	Код качества ОКВЭД
rawSource	String	Исходные данные
primary	Признак основного ОКВЭД (литера)	Признак основного ОКВЭД
additionalInfo	String	Дополнительная информация
actualityDate	Date	Дата актуальности

**\*Поля возвращаются только в SOAP-ответе. По умолчанию, в онлайн-запросах изменять записи из СПАРК нельзя.**

# РЕКВИЗИТЫ КАРТОЧКИ ЕГРЮЛ/ЕГРИП

## Реквизиты ЕГРЮЛ/ЕГРИП

Поле	Тип	Описание
name	string	Полное наименование с ОПФ
nameWoOpf	string	Полное наименование без ОПФ
shortName	string	Краткое наименование с ОПФ
shortNameWoOpf	string	Краткое наименование без ОПФ
latinName	string	Наименование на латинице
ogrn	string	ОГРН
ipSurname	string	Фамилия ИП
ipName	string	Имя ИП
ipPatronymic	string	Отчество ИП
inn	string	ИНН
kpp	string	КПП
okved	string	Код ОКВЭД
opfCode	string	Код ОПФ (как в ФНС)
opf	string	Полное название ОПФ
opfShort	string	Краткое название ОПФ
opfType	string	Тип справочника ОПФ (как в ФНС)
opfDescription	string	Описание ОПФ (как в ФНС)
egrType	Тип организации (литера)	Тип организации
branchType	Тип подразделения (литера)	Тип подразделения
branchStatus	Статус наличия филиалов (литера)	Статус наличия филиалов
managementName	string	ФИО руководителя
managementPost	string	Должность руководителя
actualityDate	Дата в формате дд.мм.гггг	Дата актуальности сведений
egrHid	string	Исторический идентификатор записи в исходной ЕГР БД
activityStatusRawSource	string	Исходный статус активности организации
activityStatus	Статус активности организации (литера)	Статус активности организации

Поле	Тип	Описание
registrationDate	Дата в формате дд.мм.гггг	Дата регистрации
liquidationDate	Дата в формате дд.мм.гггг	Дата ликвидации
activityStatusDate	Дата в формате дд.мм.гггг	Дата проставления статуса
lastActivityDate	Дата в формате дд.мм.гггг	Дата последнего обращения в ФНС
capitalValue	string	Уставной капитал
capitalType	string	Тип уставного капитала
residencyCountry	Справочник кодов ОКСМ	Страна гражданства по ОКСМ
dsmpType	Тип МСП (литера)	Тип МСП
rs_opf_code	string	Код ОПФ из Росстат

# РЕКВИЗИТЫ КАРТОЧКИ ФЛ СПАРК

## Реквизиты физического лица СПАРК

Поле	Тип/Справочник	Описание
actualityDate	date	Дата актуальности в формате дд.мм.гггг
surname	string	Фамилия
name	string	Имя
patronymic	string	Отчество
fullNameQC	Статус распознавания ФИО (литера)	Статус распознавания ФИО
fullNameRawSource	string	Исходное ФИО
fullNameNSReasons	Маркеры причин NOT_SURE ФИО (литера)	Причины NOT_SURE
fullNameEditedChanges	Маркеры изменения ФИО (литера)	Изменения, внесенные при стандартизации
gender	Пол (литера)	Пол, определяемый ЕК при стандартизации ФИО
genderRawSource	string	Исходное значение пола до стандартизации
inn	string	ИНН
innQC	Код качества ИНН ЮЛ (литера)	Код качества ИНН
innRawSource	string	Исходное значение ИНН
position	string	Должность
birthdateYear	string	Год рождения
legalCapacityEndDate	date	Дата потери правоспособности ФЛ

**Поля возвращаются только в SOAP-ответе. По умолчанию, в онлайн-запросах изменять записи из СПАРК нельзя.**

# РЕКВИЗИТЫ КАРТОЧКИ ЮЛ СПАРК

## Реквизиты юридических лиц СПАРК

Поле	Тип/Справочник	Описание
actualityDate	date	Дата актуальности в формате дд.мм.гггг
rawSourceXML	string	Исходные данные в xml
sparkGuid	date	GUID в СПАРК
name	date	Полное наименование (без ОПФ) или ФИО ИП
nameRawSource	date	Исходное наименование компании одной строкой
nameQC	Код качества наименования компании (литера)	Код качества наименования
shortName	string	Краткое наименование компании без ОПФ
shortNameQC	Код качества сокращенного наименования компании (литера)	Код качества сокращенного наименования
shortNameRawSource	string	Исходное сокращенное наименование
latinShortName	string	Краткое наименование компании на английском языке
hiddenName	string	Наименование для дедупликации
ipSurname	string	Фамилия
ipName	string	Имя
ipPatronymic	string	Отчество
fullNameQC	Статус распознавания ФИО (литера)	Статус распознавания ФИО
fullNameRawSource	string	Исходное ФИО
fullNameNSReasons	Причины отправки ФИО на ручной разбор (литера)	Причины NOT_SURE
fullNameEditedChanges	Маркеры проверки и изменения ФИО (литера)	Изменения, внесенные при стандартизации
gender	Пол (литера)	Пол, определяемый ЕК при стандартизации ФИО
genderRawSource	string	Исходное значение пола до стандартизации
genderOC	string	Код качества пола

opf	string	Полная ОПФ
opfShort	string	Краткая ОПФ
opfRawSource	string	Исходное значение ОПФ
okopfTextRawSource	string	Наименование ОПФ по справочнику ОКОПФ 2013г.+. Берем то значение, которое присылает СПАРК
okopfCodeRawSource	string	Код ОКОПФ 2013г.+. Берем то значение, которое присылает СПАРК
birthdate	string	Дата рождения ИП
birthdateQC	Код качества даты рождения (литера)	Код качества даты рождения
birthdateRawSource	string	Исходное значение даты рождения
birthPlace	string	Место рождения ИП
companyType	Тип компании СПАРК (литера)	Тип компании СПАРК
inn	string	ИНН
innQC	Код качества ИНН ЮЛ (литера)	Код качества ИНН
innRawSource	string	Исходное значение ИНН
sparkLegalType	Тип записи (литера)	Тип записи: юрлицо или ИП
registrationDate	date	Дата регистрации
ogrn	string	ОГРН/ОГРНИП
ogrnQC	Код качества ОГРН (литера)	Код качества ОГРН
ogrnRawSource	string	Исходный ОГРН
kpp	string	КПП
kppQC	Код качества КПП (литера)	Код качества КПП
kppRawSource	string	Исходный КПП
consolidatedIndicator	Сводный индикатор риска (литера)	Сводный индикатор риска
okpo	string	ОКПО
okpoQC	Код качества ОКПО (литера)	Код качества ОКПО
okpoRawSource	string	Исходный ОКПО

bik	string	Код БИК банковской организации
site	string	Сайт компании
charterCapital	string	Величина уставного капитала
revenue	string	Выручка, млн. рублей
companySize	Градация размера (литера)	Градация размера
financeIncome	string	Сумма доходов от ФНС за отчетный период
financeExpenses	string	Сумма расходов от ФНС за отчетный период
residencyCountryCode	string	Код страны в СПАРК
residencyRawSource	string	Исходное значение страны регистрации/гражданства ИП
activityStatus	Статус активности организации (литера)	Статус активности организации по данным СПАРК
activityStatusCode	string	Код статуса активности из СПАРК
activityStatusTxt	string	Текст статуса активности из СПАРК
activityStatusDate	date	Дата обновления информации о статусах
activityStatusFlag	Тернарный признак (литера)	Признак действующей компании/ИП (Флаг 1/0/null)
workersRange	Диапазон численности персонала (литера)	Диапазон численности персонала
workersCount	string	Численность персонала по данным ФНС
regimeUsn	Тернарный признак (литера)	Упрощенная система налогообложения (УСН)
regimeEshn	Тернарный признак (литера)	Единый сельскохозяйственный налог (ЕСХН)
regimeEnvd	Тернарный признак (литера)	Единый налог на вмененный доход (ЕНВД)
regimeSrp	Тернарный признак (литера)	Соглашение о разделе продукции (СРП)
okoguCode	string	ОКОГУ код
okoguName	string	ОКОГУ наименование



okfsCode	string	ОКФС код
okfsName	string	ОКФС наименование
okato	string	Код ОКАТО
oktmo	string	Код ОКТМО
egrpoFlag	Признак включения в ЕГРПО (литера)	Признак включения в ЕГРПО (Росстат)
egrulLikvidati on	string	Сведения о ликвидации по данным ФНС

**Поля возвращаются только в SOAP-ответе. По умолчанию, в онлайн-запросах изменять записи из СПАРК нельзя.**

# СВИДЕТЕЛЬСТВО О РЕГИСТРАЦИИ ТС

## Свидетельство о регистрации ТС

Поле	Readonly? *	Тип	Описание
qualifierType	нет	Тип документа ТС (литера)	Тип документа транспортного средства
licensePlate	нет	string	Государственный регистрационный знак ТС
type	да	Общий тип документа (литера)	Документ транспортного средства (всегда DOCUMENT_AUTO)
documentSeries	нет	string	Серия свидетельства о гос. регистрации
documentNumber	нет	string	Номер свидетельства о гос. регистрации
actualityDate	нет	string	Дата актуальности
qualityCode	да	string	Код качества свидетельства о регистрации (всегда GOOD)
rawSource	да	string	Исходное значение свидетельства (одной строкой)
primary	нет	boolean	Признак основного свидетельства
additionalInfo	нет	string	Дополнительная информация (примечание)
author	да	string	Автор версии
issueDate	нет	string	Дата выдачи документа

# СОГЛАСИЕ

## Согласие

Поле	Readonly? *	Тип	Описание
actualityDate	нет	string	Дата актуальности
type	нет	Тип согласия	Тип согласия (литера)
consentValue	нет	Значение согласия	Значение согласия (литера)
rawSource	да	string	Исходное значение согласия
primary	да	boolean	Признак основного согласия (всегда FALSE)
qualityCode	да	string	Код качества согласия
author	да	string	Автор версии
additionalInfo	да	string	Дополнительная информация (примечание)

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# СПИСОК СПАРК

## *Список СПАРК*

Списки СПАРК — информация о вхождении ЮЛ/ИП в списки СПАРК.

Поле	Тип/Справочник	Описание
type	Тип списка СПАРК (литера)	Тип списка СПАРК
name	String	Наименование списка
value	String	Значение списка
rawSource	String	Исходные данные
negative	Признак негативности списка (литера)	Признак негативности списка
additionalInfo	String	Дополнительная информация
actualityDate	Date	Дата актуальности
primary	Признак основного (литера)	Признак основного

**Поля возвращаются только в SOAP-ответе. По умолчанию, в онлайн-запросах изменять записи из СПАРК нельзя.**

# ТЕЛЕФОН

## Телефон

Поле	Readonly? *	Тип	Описание
primary	нет	boolean	Признак основного телефона
actualityDate	нет	string	Дата актуальности
type	нет	Тип телефона (литера)	Тип телефона
operator	да	string	Оператор связи
operatorAddress	да	string	Район/город/НП, определенный по номеру телефона
operatorRegion	да	string	Регион, определенный по номеру телефона
countryCode	нет	string	Код страны
cityCode	нет	string	Код города
number	нет	string	Номер телефона
extension	нет	string	Добавочный номер
author	да	string	Ссылка на источник атрибута
qualityCode	да	string	Код качества телефона
rawSource	нет	string	Исходное значение телефона (одной строкой)
additionalInfo	нет	string	Дополнительная информация (примечание)
timezone	да	Часовой пояс (литера)	Часовой пояс
numberProfile	да	Характеристика номера телефона (литера)	Характеристика номера телефона
popularity	да	Код популярности (литера)	Популярность

\*Признак, можно ли указывать поле в SOAP-запросе, или оно фигурирует только в SOAP-ответе.

# ТИП ИНФОРМАЦИИ О КОНТРАГЕНТЕ

## *Тип информации о контрагенте*

Литера	Описание
REQUISITE	Реквизиты
ATTRIBUTE	Атрибуты
SUBATTRIBUTE	Податрибуты. Работает только вместе с ATTRIBUTE.
SOURCE	Для объединенного контрагента информация об исходных контрагентах.
RELATION	Взаимосвязи
PAST_ATTRIBUTE	Прошлые значения

# ТИП ИНФОРМАЦИИ ОБ ИСХОДНОМ КОНТРАГЕНТЕ (ИЗ ЗАПРОСА)

*Тип информации об исходном контрагенте (из запроса)*

Литера	Описание
REQUISITE	Реквизиты
ATTRIBUTE	Атрибуты
RELATION	Взаимосвязи

# ТИП ОБНОВЛЕНИЯ КОНТРАГЕНТА

## *Тип обновления контрагента*

Литера	Описание
PARTIAL	Частичное. В запросе передаются только изменившиеся данные.  <b>Обратите внимание:</b> значения для полей, которые не были переданы в запросе, останутся без изменений – то есть останется то значение, которое было установлено ранее. Если необходимо сбросить значение какого-либо поля в «пусто», то есть очистить значения поля, необходимо в запросе явно указать данное поле с пустым значением.  Для атрибутов и связанных полей должны передавать все поля атрибута и вся группа связанных полей. То есть нельзя передать только изменение номера паспорта без остальных полей. Также нельзя передать изменение только имени клиента без передачи полей фамилии и отчества.
FULL	Полное. В запросе передается полная карточка контрагента



# УРОВЕНЬ РИСКА КЛИЕНТА

## *Классификатор (WRiskParty)*

Поле	Тип/Справочник	Описание
amlRiskLevel	AmlRiskLevel	Уровень риска клиента
amlRiskDate	Date	Дата назначения уровня риска

# ОБЩЕЕ ОПИСАНИЕ ОБЪЕКТНОЙ МОДЕЛИ

Объектная модель онлайн-интерфейсов использует 3 основных типа:

1. **WParty** — контрагент;
2. **WAttribute** — составной атрибут контрагента, который может состоять из нескольких полей — например, адрес, телефон;
3. **WField** — отдельное поле контрагента или атрибута.

## *Контрагенты и атрибуты*

Каждый контрагент может содержать произвольное количество полей и атрибутов. Каждый атрибут, в свою очередь, может содержать произвольное количество полей.

Каждое поле идентифицируется именем. Перечень разрешенных имен полей и их типы определяется конфигурацией заказчика. Общие для всех заказчиков поля описаны ниже по тексту.

Для контрагентов и атрибутов обязательно указание их типа (**//party@type** и **//party/attribute@type**). Список разрешенных типов указан в разделе [Классификаторы](#), секции [Тип контрагента](#) и [Тип атрибута](#) соответственно.

## *Идентификаторы*

Каждый контрагент содержит уникальный идентификатор **NID** (**//party@hid**). Также исходные контрагенты (контрагенты, которые не были ни с кем объединены) содержат исходный идентификатор — пару **//party@sourceSystem** + **//party@rawId**.

Атрибуты также имеют уникальные идентификаторы — **//party/attribute@hid**, и помимо это дополнительно содержат исходный идентификатор — **//party/attribute@rawId**.

## *Исходные и объединенные контрагенты*

Объединенные контрагенты (контрагенты, образовавшиеся в результате слияния двух или более исходных контрагентов) хранят информацию о привязке к своим исходным контрагентам в элементах вида **//party/source**. Каждый **//party/source** соответствует одному исходному контрагенту, и содержит его идентификаторы.

У объединенных контрагентов, в отличие от исходных, отсутствуют **//party@sourceSystem** и **/party@rawId**.

# PARTYWS И PARTYRA

Интерфейс **PartyWS** предоставляет методы, позволяющие получать информацию о контрагентах и изменять ее. Взаимодействие осуществляется по протоколу SOAP.

Аналогичные методы предоставляет интерфейс **PartyRA**, при этом взаимодействие осуществляется с помощью REST-сервисов.

Интерфейсы включают следующие методы:

- **Party.save** — Создание нового контрагента или обновление существующего.
- **Party.saveAndMerge** — Сохранить или обновить одного и более контрагентов с проверкой на дубликаты и слиянием.
- **Party.getByRawID** — Метод позволяет получить контрагента по идентификатору исходной клиентской записи.
- **Party.getByHID** — Метод позволяет получить контрагента по его идентификатору в Едином клиенте.
- **Party.search** — Метод позволяет найти всех контрагентов, удовлетворяющих заданным критериям.
- **Party.fuzzySearch** — Метод позволяет найти всех контрагентов, являющихся дубликатами для заданной записи.
- **Party.findInFns** — Метод позволяет получить ИНН от ФНС по набору ПДн без сохранения карточки в CDI ЕК (кредитный конвейер, скоринг и т.д.).
- **Party.searchRelatedParties** — Метод позволяет найти связанных контрагентов.
- **Party.checkBlacklist** — Метод позволяет проверить контрагента по черным спискам и перечню недействительных паспортов.
- **Party.checkBlacklistByRawID** — Метод проверяет контрагента по черным спискам по rawId и коду исходной системы и сразу создает связь между искомым КА и записью ЧС.
- **Party.checkBlackListByHid** — Метод проверяет контрагента по черным спискам по хиду и сразу создает связь между искомым КА и записью ЧС.
- **Party.close** — Метод позволяет закрыть контрагента по его идентификатору в Едином Клиенте (HID) или по идентификатору в исходной системе (RawId).
- **Party.closeAttribute** — Закрыть атрибут клиента по его hid.
- **Party.closeAttributeByRawId** — Метод позволяет закрыть атрибут клиента по его rawId.
- **Party.getReference** — Метод позволяет получить все значения из справочника Классификаторов <http://confluence.hflabs.ru/pages/viewpage.action?pageId=19433287> по выбранному классификатору.
- **Party.saveRelations** — Метод позволяет создавать и закрывать взаимосвязи между объектами.
- **Party.getRelations** — Метод позволяет получить взаимосвязи объекта с другими объектами по идентификатору в «Едином клиенте». Возвращает связи между золотыми контрагентами.
- **Party.getSourceAttributes** — Метод позволяет получить по идентификатору атрибута в Едином клиенте информацию об атрибутах, из которых он был составлен.
- **Party.mergeSource** — Устарел, больше не поддерживается.

- PartyWS и PartyRA (WSDL, WADL и схемы)

# PARTY.SAVE

Создание нового контрагента или обновление существующего.

- Входные параметры
- Результирующие данные
- Особенности использования
  - Создание нового контрагента
  - Обновление существующего исходного контрагента
  - Обновление закрытого контрагента
  - Обновление объединенного контрагента
  - Взаимосвязи между контрагентами
  - Взаимосвязи между атрибутами
    - Особенности использования
  - Поля контрагента
  - Стандартизация
- Исключительные ситуации
  - Исключительные ситуации при создании взаимосвязи между контрагентами
  - Исключительные ситуации при создании взаимосвязи между атрибутами
- Ограничения использования
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Создание нового контрагента
  - Пример 2. Изменение существующего исходного контрагента
  - Пример 3. В запросе указан несуществующий идентификатор NID
  - Пример 4. Создание контрагента со связью
  - Пример 5. Полное обновление контрагента
  - Пример 6. Создание контрагента со связью между атрибутами
  - Пример 7. Закрытие связи между атрибутами
  - Пример 8. Закрытие атрибута в режиме частичного (PARTIAL) обновления
- Примеры вызова PartyRA (JSON)
  - Пример 1. Создание нового контрагента
  - Пример 1. Создание нового контрагента со связью

## Входные параметры

Имя параметра	Тип	Обязательный?	Описание
party	WParty	да	Информация по контрагенту

updateType	Тип обновления контрагента	нет	Тип обновления. Умолчательное значение — PARTIAL
include/partyInfo []	Тип информации о контрагенте	нет	Дополнительная информация о контрагенте

Обратите внимание, что данный метод предполагает передачу только одной карточки контрагента (*WParty*).

## Результирующие данные

Имя параметра	Тип	Описание
party	WParty	Информация по сохраненному контрагенту

## Особенности использования

### Создание нового контрагента

Запрос на создание нового клиента должен содержать информацию по нему и составной исходный идентификатор **//party@sourceSystem + //party@rawId**, который ранее не встречался в Едином клиенте.

Идентификатор в системе Единый клиент (**//party@hid**) не передается при создании клиента. Если клиент с переданным исходным идентификатором уже есть в системе, он будет **обновлен**, либо будет создан новый клиент (с новым идентификатором **HID** в системе Единый клиент), если существующий закрыт.

Если запрос содержит **взаимосвязь (//party@relation)**, то система создаст нового клиента и установит связь между ним и указанным в связи контрагентом. Для этого вторая сторона связи должна существовать в системе.

В качестве идентификаторов отдельных атрибутов при создании используются значения **//attribute@rawId**. Если **//attribute@rawId** не задан, идентификатор атрибута будет автоматически сгенерирован системой.

### Обновление существующего исходного контрагента

Существует **два режима обновления**:

1. Частичное (по умолчанию) — в запросе передаются только изменившиеся данные. Все остальные данные в Едином клиенте остаются без изменений. Для закрытия атрибута нужно передать **//attribute@deleted = true**, либо использовать методы `Party.CloseAttribute` или `Party.CloseAttributeByRawId`,
2. Полное — в запросе передаются все данные по контрагенту, Единый клиент сам вычисляет, какие данные изменились. Все атрибуты, которые есть в актуальной версии контрагента, но не переданы в запросе, будут закрыты.

Для указания обновляемого контрагента используют его идентификатор в исходной системе или идентификатор в источнике вместе с идентификатором в Едином клиенте. При передаче обоих идентификаторов выбор происходит на основе идентификатора в Едином клиенте. Например, в запросе передан:

- **//party@sourceSystem + //party@rawId** — будет обновлен исходный клиент с указанным идентификатором.
- **//party@hid** и **//party@sourceSystem + //party@rawId** — будет обновлен клиент с указанным идентификатором в системе Единый клиент (**//party@hid**) от имени записи "**//party@sourceSystem + //party@rawId**" (сохраняется как автор полученной для обновления информации). Конфликтные случаи:
  - запись с "**//party@sourceSystem + //party@rawId**" существует в Едином клиенте с другим **hid** — будет возвращена ошибка.
  - запись с "**//party@sourceSystem + //party@rawId**" не существует в Едином клиенте — будет обновлен контрагент с переданным в запросе **//party@hid**.

В качестве идентификаторов отдельных атрибутов при обновлении используются:

- значения **//attribute@hid** — для уже существующих атрибутов (значения **//attribute@rawId** для них игнорируются);
- значения **//attribute@rawId** — при отсутствии идентификатора **//attribute@hid**. Если в системе у контрагента уже существует атрибут с указанным исходным идентификатором **//attribute@rawId**, этот атрибут будет обновлен, иначе — создан новый.

## Обновление закрытого контрагента

При вызове метода **save** с идентификатором **//party@sourceSystem + //party@rawId** система проверяет наличие такого идентификатора среди закрытых контрагентов.

1. Если таких КА нет, создается новый контрагент
2. Если такие **//party@sourceSystem + //party@rawId** есть среди закрытых, проверяется обновление по правилам **обнаружения конфликтов**.
  - a. Если конфликта нет, то закрытый контрагент переоткрывается с прежним **HID** и историей, и на него применяется обновление.
  - b. Если конфликт есть, создается новый контрагент с новым **HID**.

## Обновление объединенного контрагента

Если контрагент был объединен в результате слияния, то при вызове метода **save** с идентификатором **HID** исходного контрагента система обновит как исходного, так и объединенного контрагентов, в соответствии с правилами обновления. При этом в ответе система вернет только объединенного (но не исходного) контрагента.

При вызове метода с идентификатором НID объединенного контрагента исходный идентификатор (**//party@sourceSystem + //party@rawId**) должен соответствовать какому либо из контрагентов-источников для объединенного контрагента. В этом случае система обновит как исходного, так и объединенного контрагентов.

Если переданная информация по контрагенту содержит информацию о **взаимосвязи** (**//party@relation**), то Единый клиент обновит исходного и объединенного контрагентов и установит взаимосвязь между контрагентом-источником и указанным в связи контрагентом.

## ***Взаимосвязи между контрагентами***

При создании (обновлении) контрагента, связываемого с уже существующим, должны быть указаны:

- тип взаимосвязи (**//relation@type**);
- тип контрагента, с которым устанавливается связь (**//relation/first@type** или **//relation/second@type**);
- идентификатор НID контрагента (**//relation/first@hid** или **//relation/second@hid**), с которым устанавливается связь, либо его исходный идентификатор (**//relation/first@sourceSystem + //relation/first@rawId**) или **//relation/second@sourceSystem + //relation/second@rawId**).

Параметры контрагента, с которым устанавливается взаимосвязь, указываются в атрибуте **//relation@first**, либо **//relation@second** в зависимости от стороны, занимаемой данным контрагентом в указанном **типе взаимосвязи**. Оставшуюся сторону взаимосвязи указывать не следует, ее занимает создаваемый (обновляемый) контрагент.

Если исходный идентификатор (**//party@sourceSystem + //party@rawId**) связываемого объекта противоречит идентификатору НID (**//party@hid**), то приоритет отдается НID, исходный идентификатор не учитывается.

Если взаимосвязь уже существует:

- **//relation@deleted = FALSE (по умолчанию)**— связь останется без изменений.
- **//relation@deleted = TRUE** — связь будет закрыта.

## ***Взаимосвязи между атрибутами***

При создании (обновлении) связи между атрибутами должны быть указаны:

- тип взаимосвязи (**//attributeRelation@type**);
- типы связанных атрибутов между которыми устанавливается связь (**//attributeRelation/first@type** или **//attributeRelation/second@type**);



- идентификатор HID атрибутов (**//attributeRelation/first@hid** или **//attributeRelation/second@hid**), между которыми устанавливается связь, либо их исходный идентификатор (**//attributeRelation/first@rawId** и **//attributeRelation/second@rawId**).

Связываемые атрибуты должны быть указаны в том же порядке, что и соответствующие им сущности в описании типов взаимосвязей.

Если исходный идентификатор (**@rawId**) связываемого атрибута противоречит идентификатору HID (**@hid**), то приоритет отдается HID, исходный идентификатор не учитывается.

Если взаимосвязь уже существует:

- **//attributeRelation@deleted = FALSE (по умолчанию)**— связь останется без изменений.
- **//attributeRelation@deleted = TRUE** — связь будет закрыта.

## Особенности использования

Связи между одними и теми же атрибутами в системе, указанными в разном порядке, считаются двумя разными взаимосвязями.

Если атрибут А принадлежит атрибуту Б, то на первом месте будет атрибут-родитель Б (**//attributeRelation/first@type**) а на втором дочерний атрибут А (**//attributeRelation/second@type**).

Сохранение связи выполняется в одной транзакции с созданием/обновлением контрагента, поэтому в случае ошибки в любой части запроса никакие изменения не сохраняются.

## Поля контрагента

Перечень поддерживаемых системой полей контрагента указан в разделе [Объектная модель SOAP](#).

## Стандартизация

Значения атрибутов контрагента, которые проходят стандартизацию (например, ФИО, адреса, телефоны и т.п.) могут быть переданы одним из двух способов:

- Одной строкой, в поле вида **\*rawSource**. Например:  
 ФИО — **//party/field[@name='fullNameRawSource'];**  
 Телефон — **//attribute[@type='PHONE']/field[@name='rawSource'].**
- Раздельными полями. Например:  
 ФИО — в полях  
**//party/field[@name='surname']**

```
//party/field[@name='name']  
//party/field[@name='patronymic'];
```

Телефон — в полях

```
//attribute[@type='PHONE']/field[@name='countryCode']  
//attribute[@type='PHONE']/field[@name='cityCode']  
//attribute[@type='PHONE']/field[@name='number']
```

Если для атрибута заполнены одновременно поле **rawSource** и отдельные поля, то будет обработано значение из поля **rawSource**, гранулярные значения будут проигнорированы.

Исключением являются документы: для них используются именно гранулярные значения. В [примере 1](#) ниже паспорт передается в виде отдельных полей, а телефон — в виде единого поля **rawSource**.

## Исключительные ситуации

Система вернет ошибку в случаях:

1. Указан несуществующий код исходной системы **//party@sourceSystem**.
2. Не указан исходный идентификатор **//party@sourceSystem + //party@rawId**.
3. Указан несуществующий **//party@hid**.
4. Исходный идентификатор **//party@sourceSystem + //party@rawId** противоречит идентификатору в Едином клиенте **//party@hid**.
5. Указан **//party@hid** для объединенного контрагента, который был полностью разъединен.
6. Указан **//party@hid** для контрагента, который был закрыт.
7. В режиме полного обновления (**updateType = FULL**) передан идентификатор в Едином клиенте для контрагента **//party@hid** или атрибута **//party@attribute@hid**.
8. В режиме полного обновления (**updateType = FULL**) передан для атрибута флаг **//party/attribute@deleted = true**.
9. Передаётся несколько значений для реквизита одновременно.

## Исключительные ситуации при создании взаимосвязи между контрагентами

Система вернет ошибку в случаях:

1. Указан несуществующий тип взаимосвязи **//relation@type**.
2. Для взаимосвязи указаны обе стороны (**//relation@first** и **//relation@second**).
3. Не найден контрагент, указанный в качестве второго конца связи.
4. Контрагент, указанный в качестве второго конца связи, полностью разъединен или закрыт.

5. Второй конец связи указывает на самого себя (совпадают идентификаторы концов связи в Едином клиенте).
6. Тип создаваемого (обновляемого) контрагента (**//party@type**) и тип объекта, указанного для взаимосвязи (**//relation/first@type** или **//relation/second@type**) не соответствуют типу объекта для заданного типа взаимосвязи.

## ***Исключительные ситуации при создании взаимосвязи между атрибутами***

Система вернет ошибку в случаях:

1. Указан несуществующий тип взаимосвязи **//attributeRelation@type**.
2. Для взаимосвязи не указана одна из сторон (**//attributeRelation@first** или **//attributeRelation@second**).
3. Атрибут, указанный в качестве одного из концов связи закрыт или не найден.
4. Тип атрибута, указанного для взаимосвязи (**//attributeRelation/first@type** или **//attributeRelation/second@type**) не соответствуют типу объекта для заданного типа взаимосвязи.

## **Ограничения использования**

Если при вызове метода с идентификатором NID объединенного контрагента исходный идентификатор (**//party@sourceSystem + //party@rawId**) не соответствует ни одному из существующих контрагентов, то система вернет ошибку. Таким образом, поддерживается обновление объединенных контрагентов только от имени одного из его исходных контрагентов-источников.

При создании взаимосвязи связать создаваемого (обновляемого) контрагента можно только с уже существующим в системе контрагентом.

## **Примеры вызова PartyWS (SOAP)**

### ***Пример 1. Создание нового контрагента***

Запрос:

```
Запрос  
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <save>  
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">  
        <field name="actualityDate">22.02.2012</field>
```

```

    <field name="fullNameRawSource">Курко Олег Игоревич</field>
    <attribute type="PHONE" rawId="AL.2.PH.1">
      <field name="type">MOBILE</field>
      <field name="rawSource">9165345622</field>
    </attribute>
    <attribute type="DOCUMENT_PASSPORT" rawId="AL.2.DOC.1">
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="documentSeries">4509</field>
      <field name="documentNumber">672184</field>
    </attribute>
  </party>
</save>
</soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">КУРКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <attribute type="PHONE" hid="1025" rawId="AL.2.PH.1">
          <field name="actualityDate"/>
          <field name="type">MOBILE</field>
          <field name="countryCode">7</field>
          <field name="cityCode">916</field>
          <field name="number">5345622</field>
          <field name="extension"/>
          <field name="primary"/>
          <field name="additionalInfo"/>
          <field name="author">AL:2</field>
          <field name="qualityCode">GOOD</field>
          <field name="rawSource">9165345622</field>
        </attribute>
        <attribute type="DOCUMENT_PASSPORT" hid="1025" rawId="AL.2.DOC.1">
          <field name="actualityDate"/>
          <field name="qualifierType">PASSPORT_RU</field>
          <field name="documentSeries">45 09</field>
          <field name="documentNumber">672184</field>
          <field name="issueDate"/>
          <field name="issueAuthority"/>
          <field name="departmentCode"/>
          <field name="expiryDate"/>
          <field name="author">AL:2</field>
          <field name="qualityCode">EDITED</field>
          <field name="rawSource">4509 672184</field>
        </attribute>
      </party>

```

```
</saveResponse>
</soapenv:Body>
</soapenv:Envelope>
```

## Пример 2. Изменение существующего исходного контрагента

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">
        <field name="actualityDate">25.02.2012</field>
        <field name="surname">КЛИЧКО</field>
        <field name="birthdateRawSource">02.01.1980</field>
        <attribute type="PHONE" rawId="PHONE.1024">
          <field name="type">MOBILE</field>
          <field name="countryCode">7</field>
          <field name="cityCode">916</field>
          <field name="number">2263098</field>
        </attribute>
      </party>
    </save>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1534" sourceSystem="AL" rawId="2">
        <field name="actualityDate">25.02.2012</field>
        <field name="surname">КЛИЧКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate">02.01.1980</field>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">02.01.1980</field>
        <attribute type="PHONE" hid="1025" rawId="AL.2.PH.1">
          <field name="actualityDate"/>
          <field name="type">MOBILE</field>
          <field name="countryCode">7</field>
          <field name="cityCode">916</field>
          <field name="number">9754221</field>
          <field name="extension"/>
          <field name="primary"/>
          <field name="additionalInfo"/>
        </attribute>
      </party>
    </saveResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <field name="author">AL:2</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">79165345622</field>
  </attribute>
  <attribute type="DOCUMENT_PASSPORT" hid="1025" rawId="AL.2.DOC.1">
    <field name="actualityDate"/>
    <field name="qualifierType">PASSPORT_RU</field>
    <field name="documentSeries">45 09</field>
    <field name="documentNumber">672184</field>
    <field name="issueDate"/>
    <field name="issueAuthority"/>
    <field name="departmentCode"/>
    <field name="expiryDate"/>
    <field name="author">AL:2</field>
    <field name="qualityCode">EDITED</field>
    <field name="rawSource">4509 672184</field>
  </attribute>
</party>
</saveResponse>
</soapenv:Body>
</soapenv:Envelope>

```

### Пример 3. В запросе указан несуществующий идентификатор HID

Запрос:

```

          Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" hid="12345669420" sourceSystem="AL" rawId="12345669420">
        <field name="surname">КЛИЧКО</field>
      </party>
    </save>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>No party version found for 'PHYSICAL' with HID 123456</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

## Пример 4. Создание контрагента со связью

В примере создается новый контрагент (физ.лицо), который является собственником по отношению к существующему в системе юридическому лицу. Так как в данном типе взаимосвязи юрлицо является стороной Б, то его реквизиты указываются в параметре `\\relation@second`.

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="maritalStatus">SINGLE_MALE</field>
        <relation type="11">
          <second type="LEGAL" sourceSystem="AL" rawId="156"/>
        </relation>
      </party>
    </save>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">КУРКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <field name="maritalStatus">SINGLE_MALE</field>
        <field name="maritalStatusAuthor">AL:2</field>
      </party>
    </saveResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Пример 5. Полное обновление контрагента

До обработки запроса в карточке контрагента есть телефон и документ. Поскольку запрос не содержит документа и параметр updateType = FULL, документ закрывается.

Исходное состояние:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">25.02.2012</field>
        <field name="surname">КЛИЧКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate">02.01.1980</field>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">02.01.1980</field>
        <attribute type="PHONE" hid="1025" rawId="AL.2.PH.1">
          <field name="actualityDate"/>
          <field name="type">MOBILE</field>
          <field name="countryCode">7</field>
          <field name="cityCode">916</field>
          <field name="number">9754221</field>
          <field name="extension"/>
          <field name="primary"/>
          <field name="additionalInfo"/>
          <field name="author">AL:2</field>
          <field name="qualityCode">GOOD</field>
          <field name="rawSource">79165345622</field>
        </attribute>
        <attribute type="DOCUMENT_PASSPORT" hid="1025" rawId="AL.2.DOC.1">
          <field name="actualityDate"/>
          <field name="qualifierType">PASSPORT_RU</field>
          <field name="documentSeries">45 09</field>
          <field name="documentNumber">672184</field>
          <field name="issueDate"/>
          <field name="issueAuthority"/>
          <field name="departmentCode"/>
          <field name="expiryDate"/>
          <field name="author">AL:2</field>
          <field name="qualityCode">EDITED</field>
          <field name="rawSource">4509 672184</field>
        </attribute>
      </party>
    </saveResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Запрос:

\_\_\_\_\_



### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">
        <field name="actualityDate">25.02.2012</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate">02.01.1980</field>
        <attribute type="PHONE" rawId="AL.2.PH.1">
          <field name="type">MOBILE</field>
          <field name="countryCode">7</field>
          <field name="cityCode">916</field>
          <field name="number">9754221</field>
        </attribute>
      </party>
      <updateType>FULL</updateType>
    </save>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">25.02.2012</field>
        <field name="surname">КУРКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate">02.01.1980</field>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">02.01.1980</field>
        <attribute type="PHONE" hid="1025" rawId="AL.2.PH.1">
          <field name="actualityDate"/>
          <field name="type">MOBILE</field>
          <field name="countryCode">7</field>
          <field name="cityCode">916</field>
          <field name="number">9754221</field>
          <field name="extension"/>
          <field name="primary"/>
          <field name="additionalInfo"/>
          <field name="author">AL:2</field>
          <field name="qualityCode">GOOD</field>
          <field name="rawSource">79165345622</field>
        </attribute>
      </party>
    </saveResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Пример 6. Создание контрагента со связью между атрибутами

В примере создается контрагент со связанными атрибутами телефон - аккаунт.

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <attribute type="PHONE" rawId = "PH.1">
          <field name="type">HOME</field>
          <field name="rawSource">9165345004</field>
        </attribute>
        <attribute type="ACCOUNT" rawId="ACC.1">
          <field name="type">SKYPE</field>
          <field name="account">lucky</field>
        </attribute>
        <attributeRelation type="1000">
          <first type="PHONE" rawId="PH.1">
            <second type="ACCOUNT" rawId="ACC.1"/>
          </attributeRelation>
        </party>
      </save>
    </soapenv:Body>
  </soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="2">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        ...
        <attribute type="ACCOUNT" hid="1024" rawId="ACC.1">
          <field name="type">SKYPE</field>
          <field name="account">lucky</field>
        ...
        </attribute>
        <attribute type="PHONE" hid="1024" rawId="PH.1">
          <field name="qualityCode">GOOD</field>
          <field name="type">MOBILE</field>
          <field name="rawSource">9165345004</field>
        ...
        </attribute>
        <attributeRelation type="1000" hid="1024">
          <first type="PHONE" rawId="PH.1" hid="1024"/>
          <second type="ACCOUNT" rawId="ACC.1" hid="1024"/>
        </attributeRelation>
      </party>
    </saveResponse>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

## Пример 7. Закрытие связи между атрибутами

В примере на существующего КА со связанными атрибутами телефон - аккаунт приходит обновление в котором связь закрывается.

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <attribute type="PHONE" rawId = "PH.1">
          <field name="type">HOME</field>
          <field name="rawSource">9165345004</field>
        </attribute>
        <attribute type="ACCOUNT" rawId="ACC.1">
          <field name="type">SKYPE</field>
          <field name="account">lucky</field>
        </attribute>
        <attributeRelation type="1000" deleted = "true">
          <first type="PHONE" rawId="PH.1"/>
          <second type="ACCOUNT" rawId="ACC.1"/>
        </attributeRelation>
      </party>
    </save>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="2">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        ...
        <attribute type="ACCOUNT" hid="1024" rawId="ACC.1">
          <field name="type">SKYPE</field>
          <field name="account">lucky</field>
        </attribute>
        <attribute type="PHONE" hid="1024" rawId="PH.1">
          <field name="qualityCode">GOOD</field>
          <field name="type">MOBILE</field>
          <field name="rawSource">9165345004</field>
        </attribute>
        ...
      </party>
    </saveResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</party>
</saveResponse>
</soapenv:Body>
</soapenv:Envelope>
```

## Пример 8. Заккрытие атрибута в режиме частичного (PARTIAL) обновления

На существующего КА с атрибутами телефон и аккаунт приходит обновление с закрытием аккаунта.

Запрос:

```
Запрос
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <save>
      <party type="PHYSICAL" sourceSystem="AL" rawId="2">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <attribute type="PHONE" rawId = "PH.1">
          <field name="type">HOME</field>
          <field name="rawSource">9165345004</field>
        </attribute>
        <attribute type="ACCOUNT" rawId="ACC.1" deleted = "true">
          <field name="type">SKYPE</field>
          <field name="account">lucky</field>
        </attribute>
      </party>
    </save>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveResponse>
      <party type="PHYSICAL" hid="24" sourceSystem="AL" rawId="2">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        ...
        <attribute type="PHONE" hid="24" rawId="PH.1">
          <field name="qualityCode">GOOD</field>
          <field name="type">MOBILE</field>
          <field name="rawSource">9165345004</field>
          ...
        </attribute>
      </party>
    </saveResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Создание нового контрагента

Запрос:

```
{
  "party": {
    "field": [{
      "value": "24.06.2013",
      "name": "birthdateRawSource"
    }, {"value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"}],
    "sourceSystem": "AL",
    "rawId": "1",
    "attribute": [ {
      "field": [ {
        "value": "Россия, Санкт-Петербург",
        "name": "rawSource"
      }, {
        "value": "false",
        "name": "primary"
      }, {
        "value": "23.02.2012",
        "name": "actualityDate"
      } ],
      "type": "ADDRESS"
    } ],
    "type": "PHYSICAL"
  }
}
```

Ответ:

```
{
  "party": {
    "field": [
      {
        "value": "ИВАНОВ",
        "name": "surname"
      },
      {
        "value": "ИВАН",
        "name": "name"
      },
      {
        "value": "СЕРГЕЕВИЧ",
        "name": "patronymic"
      },
      {
        "value": "MALE",
        "name": "gender"
      },
      {
        "value": "EDITED",
        "name": "fullNameQC"
      },
      {

```

```

    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Иванов Иван Сергеевич",
    "name": "fullNameRawSource"
  },
  {
    "value": "24.06.2013",
    "name": "birthdate"
  },
  {
    "value": "AL:1",
    "name": "birthdateAuthor"
  },
  {
    "value": "EDITED",
    "name": "birthdateQC"
  },
  {
    "value": "24.06.2013",
    "name": "birthdateRawSource"
  },
  {
    "value": "",
    "name": "birthPlace"
  },
  {
    "value": "AL:1",
    "name": "birthPlaceAuthor"
  },
  {
    "value": "UNKNOWN",
    "name": "maritalStatus"
  },
  {
    "value": "AL:1",
    "name": "maritalStatusAuthor"
  },
  {
    "value": "AL:1",
    "name": "actualityDateAuthor"
  },
  {
    "value": "",
    "name": "actualityDate"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [ {
  "field": [
    {
      "value": "UNDEF_03",
      "name": "qualityCode"
    },
    {
      "value": "VALIDATED",
      "name": "validationCode"
    },
    {

```

```

    "value": "OTHER",
    "name": "type"
  },
  {
    "value": "05.08.2013",
    "name": "kladrActualityDate"
  },
  {
    "value": "190000",
    "name": "postalCode"
  },
  {
    "value": "190000",
    "name": "kladrPostalCode"
  },
  {
    "value": "РОССИЯ",
    "name": "country"
  },
  {
    "value": "СЕВЕРО-ЗАПАДНЫЙ",
    "name": "district"
  },
  {
    "value": "Г",
    "name": "regionType"
  },
  {
    "value": "САНКТ-ПЕТЕРБУРГ",
    "name": "region"
  },
  {
    "value": "",
    "name": "rayonType"
  },
  {
    "value": "",
    "name": "rayon"
  },
  {
    "value": "Г",
    "name": "cityType"
  },
  {
    "value": "САНКТ-ПЕТЕРБУРГ",
    "name": "city"
  },
  {
    "value": "7800000000000",
    "name": "kladrCode"
  },
  {
    "value": "40000000000",
    "name": "okatoCode"
  },
  {
    "value": "",
    "name": "validationExtension"
  },
  {
    "value": "•Россия, Санкт-Петербург",
    "name": "rawSource"
  }

```

```

    },
    {
      "value": "false",
      "name": "primary"
    },
    {
      "value": "",
      "name": "additionalInfo"
    },
    {
      "value": "23.02.2012",
      "name": "actualityDate"
    },
    {
      "value": "AL:1",
      "name": "author"
    }
  ],
  "type": "ADDRESS",
  "hid": 1024,
  "rawId": "ADDRESS.1024"
}],
"source": [],
"relation": [],
"type": "PHYSICAL"
}}

```

## Пример 1. Создание нового контрагента со связью

Запрос:

```

{
  "party": {
    "field": [{
      "value": "24.06.2013",
      "name": "birthdateRawSource"
    }, {"value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"}],
    "sourceSystem": "AL",
    "rawId": "1",
    "type": "PHYSICAL",
    "attribute": [ {
      "field": [ {
        "value": "Россия, Санкт-Петербург",
        "name": "rawSource"
      }, {
        "value": "false",
        "name": "primary"
      }, {
        "value": "23.02.2012",
        "name": "actualityDate"
      } ]
    } ],
    "type": "ADDRESS"
  } ],
  "relation": [
    {
      "type": 100,
      "second": {
        "type": "AGREEMENT",

```



```

        "sourceSystem": "BT",
        "rawId": "1"
      }
    ]
  }
}

```

ОТВЕТ:

```

{
  "party": {
    "field": [ {
      "value": "",
      "name": "surname"
    }, {
      "value": "",
      "name": "name"
    }, {
      "value": "",
      "name": "patronymic"
    }, {
      "value": "UNKNOWN",
      "name": "gender"
    }, {
      "value": "UNKNOWN",
      "name": "genderRawSource"
    }, {
      "value": "NOT_SURE",
      "name": "fullNameQc"
    }, {
      "value": "AL:1",
      "name": "fullNameAuthor"
    }, {
      "value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"
    }, {
      "value": "EMPTY",
      "name": "surnameQc"
    }, {
      "value": "EMPTY",
      "name": "firstnameQc"
    }, {
      "value": "EMPTY",
      "name": "patronymicQc"
    }, {
      "value": "UNDEF",
      "name": "genderQc"
    }, {
      "value": "",
      "name": "fullNameEditedChanges"
    }, {
      "value": "GARBAGE GENDER_UNKNOWN FNAME_EMPTY LNAME_EMPTY MNAME_EMPTY",
      "name": "fullNameNSReasons"
    }, {
      "value": "24.06.2013",
      "name": "birthdate"
    }, {
      "value": "AL:1",
      "name": "birthdateAuthor"
    }, {

```

```

"value" : "EDITED",
"name" : "birthdateQC"
}, {
"value" : "NONE",
"name" : "birthdatePopularity"
}, {
"value" : "24.06.2013",
"name" : "birthdateRawSource"
}, {
"value" : "",
"name" : "deathDate"
}, {
"value" : "AL:1",
"name" : "deathDateAuthor"
}, {
"value" : "",
"name" : "birthPlace"
}, {
"value" : "AL:1",
"name" : "birthPlaceAuthor"
}, {
"value" : "",
"name" : "birthPlaceCountry"
}, {
"value" : "",
"name" : "birthPlaceRegion"
}, {
"value" : "",
"name" : "birthPlaceRayon"
}, {
"value" : "",
"name" : "birthPlaceCity"
}, {
"value" : "",
"name" : "birthPlaceSettlement"
}, {
"value" : "",
"name" : "birthPlaceActualCountry"
}, {
"value" : "",
"name" : "birthPlaceActualRegion"
}, {
"value" : "",
"name" : "birthPlaceActualRayon"
}, {
"value" : "",
"name" : "birthPlaceActualCity"
}, {
"value" : "",
"name" : "birthPlaceActualSettlement"
}, {
"value" : "EMPTY",
"name" : "birthPlaceQualityCode"
}, {
"value" : "VALIDATED",
"name" : "birthPlaceValidationCode"
}, {
"value" : "",
"name" : "birthPlaceValidationExtension"
}, {
"value" : "UNKNOWN",
"name" : "maritalStatus"

```

```

}, {
  "value": "AL:1",
  "name": "maritalStatusAuthor"
}, {
  "value": "",
  "name": "inn"
}, {
  "value": "AL:1",
  "name": "innAuthor"
}, {
  "value": "EMPTY",
  "name": "innQC"
}, {
  "value": "",
  "name": "innRawSource"
}, {
  "value": "",
  "name": "snils"
}, {
  "value": "AL:1",
  "name": "snilsAuthor"
}, {
  "value": "EMPTY",
  "name": "snilsQC"
}, {
  "value": "",
  "name": "snilsRawSource"
}, {
  "value": "UNKNOWN",
  "name": "vip"
}, {
  "value": "AL:1",
  "name": "vipAuthor"
}, {
  "value": "",
  "name": "citizenship"
}, {
  "value": "AL:1",
  "name": "citizenshipAuthor"
}, {
  "value": "",
  "name": "dependantsCount"
}, {
  "value": "AL:1",
  "name": "dependantsCountAuthor"
}, {
  "value": "UNKNOWN",
  "name": "sourceChannel"
}, {
  "value": "AL:1",
  "name": "sourceChannelAuthor"
}, {
  "value": "",
  "name": "branch"
}, {
  "value": "AL:1",
  "name": "branchAuthor"
}, {
  "value": "",
  "name": "organization"
}, {
  "value": "AL:1",

```

```

"name" : "organizationAuthor"
}, {
"value" : "",
"name" : "job"
}, {
"value" : "AL:1",
"name" : "jobAuthor"
}, {
"value" : "UNKNOWN",
"name" : "occupation"
}, {
"value" : "AL:1",
"name" : "occupationAuthor"
}, {
"value" : "UNKNOWN",
"name" : "segment"
}, {
"value" : "AL:1",
"name" : "segmentAuthor"
}, {
"value" : "29027",
"name" : "unifiedId"
}, {
"value" : "23.06.2021",
"name" : "createdDate"
}, {
"value" : "AL:1",
"name" : "actualityDateAuthor"
}, {
"value" : "",
"name" : "actualityDate"
} ],
"hid" : 29027,
"sourceSystem" : "AL",
"rawId" : "1",
"attribute" : [ {
"field" : [ {
"value" : "UNDEF_01",
"name" : "qualityCode"
}, {
"value" : "VALIDATED",
"name" : "validationCode"
}, {
"value" : "OTHER",
"name" : "type"
}, {
"value" : "09.04.2021",
"name" : "kladrActualityDate"
}, {
"value" : "",
"name" : "postalCode"
}, {
"value" : "",
"name" : "kladrPostalCode"
}, {
"value" : "Москва",
"name" : "country"
}, {
"value" : "",
"name" : "district"
}, {
"value" : "",

```

```

"name" : "regionType"
}, {
"value" : "",
"name" : "region"
}, {
"value" : "",
"name" : "rayonType"
}, {
"value" : "",
"name" : "rayon"
}, {
"value" : "",
"name" : "cityType"
}, {
"value" : "",
"name" : "city"
}, {
"value" : "",
"name" : "cityAreaType"
}, {
"value" : "",
"name" : "cityArea"
}, {
"value" : "",
"name" : "settlementType"
}, {
"value" : "",
"name" : "settlement"
}, {
"value" : "",
"name" : "streetType"
}, {
"value" : "",
"name" : "street"
}, {
"value" : "",
"name" : "houseNumber"
}, {
"value" : "",
"name" : "militaryNumber"
}, {
"value" : "",
"name" : "postalBox"
}, {
"value" : "",
"name" : "korpus"
}, {
"value" : "",
"name" : "stroenie"
}, {
"value" : "",
"name" : "vladenie"
}, {
"value" : "",
"name" : "hostel"
}, {
"value" : "",
"name" : "floor"
}, {
"value" : "",
"name" : "entrance"
}, {

```

```

"value" : "",
"name" : "section"
}, {
"value" : "",
"name" : "flat"
}, {
"value" : "",
"name" : "flat2"
}, {
"value" : "",
"name" : "office"
}, {
"value" : "",
"name" : "office2"
}, {
"value" : "",
"name" : "room"
}, {
"value" : "",
"name" : "room2"
}, {
"value" : "",
"name" : "kladrCode"
}, {
"value" : "",
"name" : "okatoCode"
}, {
"value" : "643",
"name" : "oksmCode"
}, {
"value" : "",
"name" : "oktmoCode"
}, {
"value" : "MISSED",
"name" : "timezone"
}, {
"value" : "",
"name" : "fiasCode"
}, {
"value" : "FIAS_COUNTRY",
"name" : "fiasLevel"
}, {
"value" : "",
"name" : "validationExtension"
}, {
"value" : "NONE",
"name" : "popularity"
}, {
"value" : "",
"name" : "geoLatitude"
}, {
"value" : "",
"name" : "geoLongitude"
}, {
"value" : "MISSED",
"name" : "geoQualityCode"
}, {
"value" : "Россия, Москва",
"name" : "rawSource"
}, {
"value" : "false",
"name" : "primary"

```

```
}, {  
  "value": "",  
  "name": "additionalInfo"  
}, {  
  "value": "23.02.2012 00:00:00",  
  "name": "actualityDate"  
}, {  
  "value": "AL:1",  
  "name": "author"  
} ],  
"type": "ADDRESS",  
"hid": 16025,  
"rawId": "ADDRESS.16025",  
"deleted": null  
} ],  
"source": [ {  
  "hid": 29027,  
  "sourceSystem": "OIS",  
  "rawId": "1"  
} ],  
"type": "PHYSICAL" }}
```

# PARTY.SAVEANDMERGE

Сохранить или обновить одного и более контрагентов с проверкой на дубликаты и слиянием.

- Описание
  - Входные параметры
  - Результирующие данные
  - Как работает
  - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 0. Пример запроса и ответа с возможными в них блоками
  - Пример 1. Сохранение контрагентов-дубликатов
  - Пример 2. Сохранение связанных контрагентов разных типов
  - Пример 3. Сохранение клиента с договором
- Примеры вызова PartyRA (JSON)
  - Пример 1. Сохранение контрагентов-дубликатов

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
party	WParty[]	да	Контрагенты
updateType	Тип обновления контрагента	нет	Тип обновления. По умолчанию — PARTIAL
include/partyInfo[]	Тип информации о контрагенте	нет	Дополнительная информация о контрагенте

### Результирующие данные

Имя параметра	Тип	Описание
party	WParty[]	Сохраненные контрагенты

### Как работает

Алгоритм:

1. Сохраним с помощью [Save](#) контрагентов party каждого типа и связи между ними.
2. Найдем дубликаты между party и остальными контрагентами в «Едином клиенте».
3. Объединим все найденные гарантированные дубликаты.



4. Для объединенных party вернем вместо исходных контрагентов результат объединения (пример 1 ниже).

Для остальных — результат Save на шаге 1 (исходные данные и HID).

Метод выполняется в одной транзакции — не сохраняем изменения при любой ошибке.

### Исключительные ситуации

Вернем ошибку, если произошла ошибка:

- при сохранении хотя бы одного контрагента или связи в методе **Save**;
- в процессе поиска дубликатов;
- в процессе слияния дубликатов.

Система не сохранит данные *ни по одному* из переданных контрагентов.

## Примеры вызова PartyWS (SOAP)

### Пример 0. Пример запроса и ответа с возможными в них блоками

Запрос

```


Запрос



```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMerge>
      <!-- Может быть передана как одна, так и несколько сущностей party -->
      <party type="PHYSICAL" sourceSystem="AL" rawId="2"> <!-- Обязательно должен быть задан type и sourceSystem + rawId, опционально можно указать hid -->
        <!-- Доступные поля описаны в модели данных контрагента. Могут быть переданы как все, для которых стоит отметка "можно прислать в soap", так и только часть из них -->
        <field name="actualityDate">22.02.2012</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        ...
        <field name="maritalStatus">SINGLE_MALE</field>

        <!-- Атрибуты клиента. Передаются опционально, может быть передано несколько. Набор доступных атрибутов определяется типом контрагента -->
        <attribute type="PHONE" rawId="AL.2.PH.1"> <!-- Обязательно должен быть задан type, опционально rawId -->
          <!-- Доступные поля описаны в модели данных атрибута. Могут быть переданы как все, для которых стоит отметка "можно прислать в soap", так и только часть из них -->
          <field name="type">MOBILE</field>
          ...
          <field name="rawSource">9165345622</field>
        </attribute>
        ...
      </party>
    </saveAndMerge>
  </soapenv:Body>
</soapenv:Envelope>
```


```

```

<attribute type="DOCUMENT_PASSPORT" rawId="AL.2.DOC.1">
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="documentSeries">4509</field>
  <field name="documentNumber">672184</field>
</attribute>

  <!-- Информация о связях контрагента (тип связи, информация о
связанной сущности). Передается опционально, может быть передано несколько
блоков со связями. Блок содержит фиксированный набор полей как в примере ниже -->
  <relation type="1"> <!-- Обязательно должен быть задан type -->
    <first type="PHYSICAL" sourceSystem="AL" rawId="1"/> <!-- type,
sourceSystem, rawId - обязательны -->
    </relation>
    ...
    <relation type="1">
    <second type="PHYSICAL" sourceSystem="AL" rawId="3"/>
    </relation>
</party>
...
<party type="PHYSICAL" sourceSystem="BT" rawId="3">
  <field name="actualityDate">22.02.2012</field>
  <field name="fullNameRawSource">Куркой Олэг Игоревич</field>
  ...
  <field name="maritalStatus">SINGLE_MALE</field>
  <attribute type="PHONE" rawId="BT.2.PH.2"> <!-- Обязательно должен
быть задан type, опционально rawId -->
    <!-- Доступные поля описаны в модели данных атрибута. Могут
быть переданы как все, для которых стоит отметка "можно прислать в soap", так и
только часть из них -->
    <field name="type">MOBILE</field>
    ...
    <field name="rawSource">9165355622</field>
  </attribute>
  ...
  <attribute type="DOCUMENT_PASSPORT" rawId="AL.2.DOC.1">
    <field name="qualifierType">PASSPORT_RU</field>
    <field name="documentSeries">4509</field>
    <field name="documentNumber">672284</field>
  </attribute>
</party>

  <!-- Тип обновления. Указывается опционально -->
  <updateType>FULL</updateType>

</saveAndMerge>
</soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMergeResponse>
      <!-- Может быть как один контрагент, так и несколько. Зависит от количества
сущностей в запросе -->
      <party type="PHYSICAL" hid="1026"> <!-- для 3К возвращается только type и hid, для
одиночной также sourceSystem и rawId -->

```

```

<!-- Возвращаемые поля описаны в модели данных контрагента.
Возвращаются все поля с отметкой "возвращается в soap" -->
<field name="actualityDate">22.02.2012</field>
<field name="surname">КУРКО</field>
<field name="name">ОЛЕГ</field>
<field name="patronymic">ИГОРЕВИЧ</field>
<field name="fullNameAuthor">AL:2</field>
<field name="fullNameQC">EDITED</field>
<field name="fullNameRawSource">Курко Олег Игоревич</field>
...
<field name="birthdate">12.12.1989</field>
<field name="birthdateAuthor">BT:3</field>
<field name="birthdateQC">EDITED</field>
<field name="birthdateRawSource">12.12.1989</field>

<!-- Атрибуты клиента. Возвращаются все атрибуты контрагента (если
есть) кроме умолчательных Согласий (их наличие в ответе регулируется настройкой в
АРМ Администратора) -->
<attribute type="PHONE" hid="1029" rawId="AL.2.PH.1">
<!-- Возвращаемые поля описаны в модели данных атрибута.
Возвращаются все поля с отметкой "возвращается в soap" -->
<field name="actualityDate"/>
<field name="type">MOBILE</field>
<field name="countryCode">7</field>
<field name="cityCode">916</field>
<field name="number">5345622</field>
...
<field name="author">AL:2</field>
<field name="qualityCode">GOOD</field>
<field name="rawSource">9165345622</field>
</attribute>
...
<attribute type="PHONE" hid="1030" rawId="BT.3.PH.1">
<field name="actualityDate"/>
<field name="type">HOME</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="number">2283412</field>
...
<field name="author">BT:3</field>
<field name="qualityCode">GOOD</field>
<field name="rawSource">4952283412</field>
</attribute>
<attribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.1">
<field name="actualityDate"/>
<field name="qualifierType">PASSPORT_RU</field>
<field name="documentSeries">45 09</field>
<field name="documentNumber">672184</field>
<field name="issueDate"/>
<field name="issueAuthority"/>
<field name="departmentCode"/>
...
<field name="expiryDate"/>
<field name="author">AL:2</field>
<field name="qualityCode">EDITED</field>
<field name="rawSource">4509 672184</field>
</attribute>

<!-- Прошлые значения атрибутов клиента. Наличие прошлых атрибутов в

```

```

ответе регулируется настройкой в АРМ Администратора -->
      <!-- Возвращаются все прошлые значения атрибуты контрагента (если
есть) -->
      <pastAttribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.1"
>
      <field name="actualityDate"/>
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="documentSeries">45 08</field>
      <field name="documentNumber">546767</field>
      <field name="issueDate"/>
      <field name="issueAuthority"/>
      <field name="departmentCode"/>
      ...
      <field name="expiryDate"/>
      <field name="author">AL:2</field>
      <field name="qualityCode">EDITED</field>
      <field name="rawSource">45 08546767</field>
</pastAttribute>
      ...
      <pastAttribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.1"
>
      <field name="actualityDate"/>
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="documentSeries">45 00</field>
      <field name="documentNumber">767453</field>
      <field name="issueDate"/>
      <field name="issueAuthority"/>
      <field name="departmentCode"/>
      ...
      <field name="expiryDate"/>
      <field name="author">AL:2</field>
      <field name="qualityCode">EDITED</field>
      <field name="rawSource">4500 767453</field>
</pastAttribute>

      <!-- Исходные карточки. Возвращаются опционально только, если
результат сохранения золотая карточка. Может быть несколько -->
      <source hid="1024" sourceSystem="AL" rawId="2"/>
      ...
      <source hid="1025" sourceSystem="BT" rawId="3"/>
</party>
      ...

      <party type="PHYSICAL" hid="1027" sourceSystem="AL" rawId="1">
      ...
</party>

</saveAndMergeResponse>
</soapenv:Body>
</soapenv:Envelope>

```

### Пример 1. Сохранение контрагентов-дубликатов

На входе два исходных контрагента, на выходе — один, образовавшийся в результате слияния.

Запрос:

```

Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">

```

```

<soapenv:Header/>
<soapenv:Body>
  <saveAndMerge>
    <party type="PHYSICAL" sourceSystem="AL" rawId="2">
      <field name="actualityDate">22.02.2012</field>
      <field name="fullNameRawSource">Курко Олег Игоревич</field>
      <field name="maritalStatus">SINGLE_MALE</field>
      <attribute type="PHONE" rawId="AL.2.PH.1">
        <field name="type">MOBILE</field>
        <field name="rawSource">9165345622</field>
      </attribute>
      <attribute type="DOCUMENT_PASSPORT" rawId="AL.2.DOC.1">
        <field name="qualifierType">PASSPORT_RU</field>
        <field name="documentSeries">4509</field>
        <field name="documentNumber">672184</field>
      </attribute>
    </party>
    <party type="PHYSICAL" sourceSystem="BT" rawId="3">
      <field name="actualityDate">20.02.2012</field>
      <field name="fullNameRawSource">Курко Олег Игоревич</field>
      <field name="birthdateRawSource">12.12.1989</field>
      <field name="maritalStatus">SINGLE_MALE</field>
      <attribute type="PHONE" rawId="BT.3.PH.1">
        <field name="type">HOME</field>
        <field name="rawSource">4952283412</field>
      </attribute>
      <attribute type="PHONE" rawId="AL.2.PH.1">
        <field name="type">MOBILE</field>
        <field name="rawSource">9165345622</field>
      </attribute>
      <attribute type="DOCUMENT_PASSPORT" rawId="BT.2.DOC.1">
        <field name="qualifierType">PASSPORT_RU</field>
        <field name="documentSeries">4509</field>
        <field name="documentNumber">672184</field>
      </attribute>
    </party>
  </saveAndMerge>
</soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMergeResponse>
      <party type="PHYSICAL" hid="1026">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">КУРКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
        <field name="birthdate">12.12.1989</field>
        <field name="birthdateAuthor">BT:3</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">12.12.1989</field>
        <field name="maritalStatus">SINGLE_MALE</field>
        <field name="maritalStatusAuthor">AL:2</field>
      </party>
    </saveAndMergeResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<attribute type="PHONE" hid="1029" rawId="AL.2.PH.1">
  <field name="actualityDate"/>
  <field name="type">MOBILE</field>
  <field name="countryCode">7</field>
  <field name="cityCode">916</field>
  <field name="number">5345622</field>
  <field name="extension"/>
  <field name="primary"/>
  <field name="additionalInfo"/>
  <field name="author">AL:2</field>
  <field name="qualityCode">GOOD</field>
  <field name="rawSource">9165345622</field>
</attribute>
<attribute type="PHONE" hid="1030" rawId="BT.3.PH.1">
  <field name="actualityDate"/>
  <field name="type">HOME</field>
  <field name="countryCode">7</field>
  <field name="cityCode">495</field>
  <field name="number">2283412</field>
  <field name="extension"/>
  <field name="primary"/>
  <field name="additionalInfo"/>
  <field name="author">BT:3</field>
  <field name="qualityCode">GOOD</field>
  <field name="rawSource">4952283412</field>
</attribute>
<attribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.1">
  <field name="actualityDate"/>
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="documentSeries">45 09</field>
  <field name="documentNumber">672184</field>
  <field name="issueDate"/>
  <field name="issueAuthority"/>
  <field name="departmentCode"/>
  <field name="expiryDate"/>
  <field name="author">AL:2</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource">4509 672184</field>
</attribute>
<source hid="1024" sourceSystem="AL" rawId="2"/>
<source hid="1025" sourceSystem="BT" rawId="3"/>
</party>
</saveAndMergeResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Пример 2. Сохранение связанных контрагентов разных типов

Запрос:

```

Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMerge>
      <party type="PHYSICAL" sourceSystem="AL" rawId="1">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
      </party>
      <party type="LEGAL" sourceSystem="BT" rawId="1">

```

```

    <field name="name">ООО МераМаркет</field>
      <relation type="9">
        <first type="PHYSICAL" sourceSystem="AL" rawId="1"/>
      </relation>
    </party>
  </saveAndMerge>
</soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMergeResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
      </party>
      <party type="LEGAL" hid="1025" sourceSystem="BT" rawId="1">
        <field name="name">ООО МераМаркет</field>
      </party>
    </saveAndMergeResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

### Пример 3. Сохранение клиента с договором

Запрос:

**Запрос**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMerge>
      <party type="PHYSICAL" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="fullNameRawSource">Курко Олег Игоревич</field>
      </party>
      <party type="AGREEMENT" sourceSystem="BT" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="number">334/09-2014</field>
        <relation type="101">
          <first type="PHYSICAL" sourceSystem="AL" rawId="1"/>
        </relation>
      </party>
    </saveAndMerge>
  </soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveAndMergeResponse>

```

```

<party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
  <field name="fullNameRawSource">Курко Олег Игоревич</field>
</party>
<party type="AGREEMENT" hid="1025" sourceSystem="AGR" rawId="1">
  <field name="number">334/09-2014</field>
</party>
</saveAndMergeResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Примеры вызова PartyRA (JSON)

### Пример 1. Сохранение контрагентов-дубликатов

На входе два исходных контрагента. На выходе один — результат слияния.

Запрос:

```

{"party": [{
  "field": [{
    "value": "24.06.2013",
    "name": "birthdateRawSource"
  }], {"value": "Иванов Иван Сергеевич",
    "name": "fullNameRawSource"}],
  "sourceSystem": "AL",
  "rawId": "1",
  "attribute": [ {
    "field": [ {
      "value": "Россия, Санкт-Петербург, улица Ленина, дом 15, кв 45",
      "name": "rawSource"
    }], {
      "value": "false",
      "name": "primary"
    }], {
      "value": "23.02.2012",
      "name": "actualityDate"
    }],
    "type": "ADDRESS"
  }],
  "type": "PHYSICAL"
},
{
  "field": [{
    "value": "24.06.2013",
    "name": "birthdateRawSource"
  }], {"value": "Иванов Иван Сергеевич",
    "name": "fullNameRawSource"}],
  "sourceSystem": "BT",
  "rawId": "1",
  "attribute": [ {
    "field": [ {
      "value": "Россия, Санкт-Петербург, улица Ленина, дом 15, кв 45",
      "name": "rawSource"
    }], {
      "value": "false",
      "name": "primary"
    }], {

```



```

    "value" : "23.02.2012",
    "name" : "actualityDate"
  },
  "type" : "ADDRESS"
}],
"type" : "PHYSICAL"
}]
}

```

ОТВЕТ:

```

{
  "party": [
    {
      "field": [
        {
          "value": "TO_ACTUALIZE",
          "name": "listCheckStatus"
        },
        {
          "value": "TO_ACTUALIZE",
          "name": "egrCheckStatus"
        },
        {
          "value": "",
          "name": "deathDate"
        },
        {
          "value": "BT:1",
          "name": "deathDateAuthor"
        },
        {
          "value": "Иванов",
          "name": "surname"
        },
        {
          "value": "Иван",
          "name": "name"
        },
        {
          "value": "Сергеевич",
          "name": "patronymic"
        },
        {
          "value": "MALE",
          "name": "gender"
        },
        {
          "value": "UNKNOWN",
          "name": "genderRawSource"
        },
        {
          "value": "EDITED",
          "name": "fullNameQC"
        },
        {
          "value": "BT:1",
          "name": "fullNameAuthor"
        },
        {
          "value": "Иванов Иван Сергеевич",

```

```

    "name": "fullNameRawSource"
  },
  {
    "value": "",
    "name": "surnameQc"
  },
  {
    "value": "",
    "name": "firstNameQc"
  },
  {
    "value": "",
    "name": "patronymicQc"
  },
  {
    "value": "UNDEF",
    "name": "genderQc"
  },
  {
    "value": "",
    "name": "fullNameEditedChanges"
  },
  {
    "value": "",
    "name": "fullNameNSReasons"
  },
  {
    "value": "24.06.2013",
    "name": "birthdate"
  },
  {
    "value": "BT:1",
    "name": "birthdateAuthor"
  },
  {
    "value": "EDITED",
    "name": "birthdateQC"
  },
  {
    "value": "NONE",
    "name": "birthdatePopularity"
  },
  {
    "value": "24.06.2013",
    "name": "birthdateRawSource"
  },
  {
    "value": "",
    "name": "birthPlace"
  },
  {
    "value": "BT:1",
    "name": "birthPlaceAuthor"
  },
  {
    "value": "",
    "name": "birthPlaceCountry"
  },
  {
    "value": "",
    "name": "birthPlaceRegion"
  },
}

```

```

{
  "value": "",
  "name": "birthPlaceRayon"
},
{
  "value": "",
  "name": "birthPlaceCity"
},
{
  "value": "",
  "name": "birthPlaceSettlement"
},
{
  "value": "",
  "name": "birthPlaceActualCountry"
},
{
  "value": "",
  "name": "birthPlaceActualRegion"
},
{
  "value": "",
  "name": "birthPlaceActualRayon"
},
{
  "value": "",
  "name": "birthPlaceActualCity"
},
{
  "value": "",
  "name": "birthPlaceActualSettlement"
},
{
  "value": "EMPTY",
  "name": "birthPlaceQualityCode"
},
{
  "value": "VALIDATED",
  "name": "birthPlaceValidationCode"
},
{
  "value": "",
  "name": "birthPlaceValidationExtension"
},
{
  "value": "UNKNOWN",
  "name": "maritalStatus"
},
{
  "value": "BT:1",
  "name": "maritalStatusAuthor"
},
{
  "value": "",
  "name": "inn"
},
{
  "value": "BT:1",
  "name": "innAuthor"
},
{
  "value": "EMPTY",

```

```

    "name": "innQC"
  },
  {
    "value": "",
    "name": "innRawSource"
  },
  {
    "value": "",
    "name": "snils"
  },
  {
    "value": "BT:1",
    "name": "snilsAuthor"
  },
  {
    "value": "EMPTY",
    "name": "snilsQC"
  },
  {
    "value": "",
    "name": "snilsRawSource"
  },
  {
    "value": "UNKNOWN",
    "name": "vip"
  },
  {
    "value": "BT:1",
    "name": "vipAuthor"
  },
  {
    "value": "",
    "name": "citizenship"
  },
  {
    "value": "",
    "name": "citizenshipCountry"
  },
  {
    "value": "VALIDATED",
    "name": "citizenshipValidationCode"
  },
  {
    "value": "",
    "name": "citizenshipValidationExtension"
  },
  {
    "value": "false",
    "name": "citizenshipInBlacklist"
  },
  {
    "value": "false",
    "name": "citizenshipInGreylist"
  },
  {
    "value": "BT:1",
    "name": "citizenshipAuthor"
  },
  {
    "value": "",
    "name": "dependantsCount"
  },
}

```

```

{
  "value": "BT:1",
  "name": "dependantsCountAuthor"
},
{
  "value": "UNKNOWN",
  "name": "sourceChannel"
},
{
  "value": "BT:1",
  "name": "sourceChannelAuthor"
},
{
  "value": "",
  "name": "branch"
},
{
  "value": "BT:1",
  "name": "branchAuthor"
},
{
  "value": "",
  "name": "organization"
},
{
  "value": "BT:1",
  "name": "organizationAuthor"
},
{
  "value": "",
  "name": "job"
},
{
  "value": "BT:1",
  "name": "jobAuthor"
},
{
  "value": "UNKNOWN",
  "name": "occupation"
},
{
  "value": "BT:1",
  "name": "occupationAuthor"
},
{
  "value": "UNKNOWN",
  "name": "agent"
},
{
  "value": "BT:1",
  "name": "agentAuthor"
},
{
  "value": "",
  "name": "tin"
},
{
  "value": "BT:1",
  "name": "tinAuthor"
},
{
  "value": "8219054",

```

```

    "name": "unifiedId"
  },
  {
    "value": "12.10.2023",
    "name": "createdDate"
  },
  {
    "value": "BT:1",
    "name": "actualityDateAuthor"
  },
  {
    "value": "",
    "name": "actualityDate"
  },
  {
    "value": "false",
    "name": "insurantFlag"
  },
  {
    "value": "1697120412969",
    "name": "lastChangeTimestamp"
  }
],
"hid": 1026,
"sourceSystem": null,
"rawId": null,
"attribute": [
  {
    "field": [
      {
        "value": "false",
        "name": "inBlacklist"
      },
      {
        "value": "false",
        "name": "inGreylist"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "VALIDATED",
        "name": "validationCode"
      },
      {
        "value": "OTHER",
        "name": "type"
      },
      {
        "value": "16.06.2023",
        "name": "kladrActualityDate"
      },
      {
        "value": "197136",
        "name": "postalCode"
      },
      {
        "value": "",
        "name": "kladrPostalCode"
      },
      {

```

```

    "value": "Россия",
    "name": "country"
  },
  {
    "value": "Северо-Западный",
    "name": "district"
  },
  {
    "value": "г",
    "name": "regionType"
  },
  {
    "value": "Санкт-Петербург",
    "name": "region"
  },
  {
    "value": "",
    "name": "rayonType"
  },
  {
    "value": "",
    "name": "rayon"
  },
  {
    "value": "г",
    "name": "cityType"
  },
  {
    "value": "Санкт-Петербург",
    "name": "city"
  },
  {
    "value": "",
    "name": "cityAreaType"
  },
  {
    "value": "",
    "name": "cityArea"
  },
  {
    "value": "",
    "name": "settlementType"
  },
  {
    "value": "",
    "name": "settlement"
  },
  {
    "value": "ул",
    "name": "streetType"
  },
  {
    "value": "Ленина",
    "name": "street"
  },
  {
    "value": "15",
    "name": "houseNumber"
  },
  {
    "value": "",
    "name": "militaryNumber"
  }

```

```
},
{
  "value": "",
  "name": "postalBox"
},
{
  "value": "",
  "name": "korpus"
},
{
  "value": "",
  "name": "stroenie"
},
{
  "value": "",
  "name": "vladenie"
},
{
  "value": "",
  "name": "hostel"
},
{
  "value": "",
  "name": "floor"
},
{
  "value": "",
  "name": "entrance"
},
{
  "value": "",
  "name": "section"
},
{
  "value": "45",
  "name": "flat"
},
{
  "value": "",
  "name": "flat2"
},
{
  "value": "",
  "name": "office"
},
{
  "value": "",
  "name": "office2"
},
{
  "value": "",
  "name": "room"
},
{
  "value": "",
  "name": "room2"
},
{
  "value": "7800000000069000",
  "name": "kladrCode"
},
{
```



```

    "value": "4000000000",
    "name": "okatoCode"
  },
  {
    "value": "643",
    "name": "oksmCode"
  },
  {
    "value": "40392000",
    "name": "oktmoCode"
  },
  {
    "value": "UTC+3",
    "name": "timezone"
  },
  {
    "value": "1faa3b1e-8558-42b0-9956-154daafe999f",
    "name": "fiasCode"
  },
  {
    "value": "FIAS_STREET",
    "name": "fiasLevel"
  },
  {
    "value": "",
    "name": "validationExtension"
  },
  {
    "value": "NONE",
    "name": "popularity"
  },
  {
    "value": "",
    "name": "geoLatitude"
  },
  {
    "value": "",
    "name": "geoLongitude"
  },
  {
    "value": "NOT_CHECKED",
    "name": "geoQualityCode"
  },
  {
    "value": "Россия, Санкт-Петербург, улица Ленина, дом 15, кв 45",
    "name": "rawSource"
  },
  {
    "value": "false",
    "name": "primary"
  },
  {
    "value": "",
    "name": "additionalInfo"
  },
  {
    "value": "23.02.2012 00:00:00",
    "name": "actualityDate"
  },
  {
    "value": "BT:1",
    "name": "author"
  }

```

```
    }
  ],
  "type": "ADDRESS",
  "hid": 1026,
  "rawId": "ADDRESS.1025",
  "deleted": null
}
],
"source": [
  {
    "hid": 1024,
    "sourceSystem": "AL",
    "rawId": "1"
  },
  {
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
  }
],
"type": "PHYSICAL"
}
]
```

# PARTY.GETBYRAWID

Метод позволяет получить контрагента по идентификатору исходной клиентской записи.

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
  - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Клиент обнаружен
  - Пример 2. Клиент обнаружен (объединённый)
  - Пример 3. Клиента с указанным идентификатором не существует
  - Пример 4. Клиент был закрыт
  - Пример 5. Клиент со взаимосвязями обнаружен, в ответе возвращён статус подтверждаемой связи
  - Пример 6. Новых данных по клиенту нет (в запросе передан параметр lastChangeTimestamp)
  - Пример 7. Исходная карточка в составе золотой (в запросе передан параметр lastChangeTimestamp)
- Примеры вызова PartyRA (JSON)
  - Пример 1. Клиент обнаружен
  - Пример 2. Клиент обнаружен (объединённый)
  - Пример 3. Обнаружен объединённый клиент, но возвращён исходный

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
sourceSystem	string	да	Код исходной системы.
rawId	string	да	Идентификатор клиента в исходной системе.
partyType	WPartyType	да	Тип контрагента
includeRelations	boolean	нет	Возвращать взаимосвязи контрагента. По умолчанию принимает значение <b>FALSE</b>
onlySourceParty	boolean	нет	Возвращать исходного контрагента, даже если он был объединён в результате слияния. По умолчанию принимает значение <b>FALSE</b> .

include /partyInfo[]	Тип информации о контрагенте	нет	Дополнительная информация о контрагенте
lastChangeTimestamp	number	нет	Время последнего изменения КА

## Результирующие данные

Имя параметра	Тип	Описание
party	WParty	Информация по контрагенту.

## Особенности использования

Если контрагент был объединён в результате слияния и параметр *onlySourceParty* имеет значение **FALSE**, то система вернёт только объединённого (но не исходного) контрагента.

Если параметр *onlySourceParty* имеет значение **TRUE**, то система вернёт исходного контрагента, даже если он был объединён в результате слияния.

Для объединённых контрагентов система должна возвращать информацию о привязке к своим исходным контрагентам (**//party/source**).

Если параметр *includeRelations* имеет значение **TRUE**, то система вернёт все взаимосвязи для данного контрагента. В ответе для взаимосвязей **//party/relation** будет указана только противоположная запрашиваемому контрагенту сторона взаимосвязи. При этом для связанного объекта в ответе будет указан только его тип и уникальный идентификатор в CDI (HID). А если связь между объектами подтверждаемая, то в ответе вернётся также статус подтверждения связи (см Пример 5 ниже).

Для объединённого контрагента система вернёт все взаимосвязи его исходных контрагентов.

Если указанный контрагент связан с контрагентом, который был объединён, то система вернёт связь с объединённым контрагентом.

Если параметр *lastChangeTimestamp* указан в запросе и имеет значение, совпадающее с *lastChangeTimestamp* в таблице **PARTY\_CHANGE\_META**, то возвращается пустой информационный ответ «Изменений нет» (см Пример 6 ниже).

Если параметр *lastChangeTimestamp* не указан в запросе, либо если его значение не совпадает со значением *lastChangeTimestamp* в таблице **PARTY\_CHANGE\_META**, то возвращается запрашиваемая карточка контрагента и свежая временная метка последних изменений карточки.

Если запрашиваются данные по rawID исходной карточки в составе золотой, то проверка *lastChangeTimestamp* происходит по золотой карточке, а не по запрашиваемой исходной (см Пример 7 ниже)

## Исключительные ситуации

Если контрагента с указанным кодом исходной системы и исходным идентификатором в системе не существует, то система вернёт ошибку.

Если указанный контрагент был закрыт, то система вернёт ошибку.

## Примеры вызова PartyWS (SOAP)

### Пример 1. Клиент обнаружен

Запрос:

```
Запрос  
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <getByRawID>  
      <sourceSystem>AL</sourceSystem>  
      <rawId>1</rawId>  
      <partyType>PHYSICAL</partyType>  
    </getByRawID>  
  </soapenv:Body>  
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <getByRawIDResponse>  
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">  
        <field name="actualityDate">22.02.2012</field>  
        <field name="surname">МАМОНОВ</field>  
        <field name="name">СЕРГЕЙ</field>  
        <field name="patronymic">ВИКТОРОВИЧ</field>  
        <field name="fullNameAuthor">AL:1</field>  
        <field name="fullNameQC">EDITED</field>  
        <field name="fullNameRawSource">Мамонов Сергей Викторович</field>  
        <field name="birthdate">03.12.1987</field>  
        <field name="birthdateAuthor">AL:1</field>  
        <field name="birthdateQC">EDITED</field>  
        <field name="birthdateRawSource">03.12.1987</field>  
        <attribute type="PHONE" hid="1024" rawId="AL.1.PH.1">  
          <field name="actualityDate">23.08.2011</field>  
          <field name="type">HOME</field>  
          <field name="countryCode">7</field>  
          <field name="cityCode">495</field>  
          <field name="number">3542344</field>  
          <field name="extension"/>  
          <field name="primary">>true</field>  
          <field name="additionalInfo"/>  
          <field name="author">AL:1</field>  
          <field name="qualityCode">GOOD</field>  
          <field name="rawSource">8-495-354-23-44</field>  
        </attribute>  
      </party>  
    </getByRawIDResponse>  
  </soapenv:Body>  
</soapenv:Envelope>
```

## Пример 2. Клиент обнаружен (объединённый)

Запрос:

```


Запрос



```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByRawID>
      <sourceSystem>AL</sourceSystem>
      <rawId>1</rawId>
      <partyType>PHYSICAL</partyType>
    </getByRawID>
  </soapenv:Body>
</soapenv:Envelope>
```


```

Ответ:

```


```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByRawIDResponse>
      <party type="PHYSICAL" hid="1026">
        <field name="surname">МОРОЗОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИТАЛЬЕВИЧ</field>
        <field name="gender">MALE</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameAuthor">BT:1</field>
        <field name="fullNameRawSource">Морозов Сергей Витальевич</field>
        <field name="birthdate">18.03.1975</field>
        <field name="birthdateAuthor">BT:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">18.03.1975</field>
        <field name="inn"/>
        <field name="innAuthor"/>
        <field name="innQC">MISSED</field>
        <field name="organization"/>
        <field name="organizationAuthor"/>
        <field name="additionalInfo"/>
        <field name="additionalInfoAuthor"/>
        <field name="terrorist">false</field>
        <field name="maritalStatus">UNKNOWN</field>
        <field name="maritalStatusAuthor"/>
        <field name="actualityDateAuthor"/>
        <field name="actualityDate"/>
        <source hid="1024" sourceSystem="AL" rawId="1"/>
        <source hid="1025" sourceSystem="BT" rawId="1"/>
      </party>
    </getByRawIDResponse>
  </soapenv:Body>
</soapenv:Envelope>
```


```

### Пример 3. Клиента с указанным идентификатором не существует

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByRawID>
      <sourceSystem>AL</sourceSystem>
      <rawId>1234</rawId>
      <partyType>PHYSICAL</partyType>
    </getByRawID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>No party of type 'PHYSICAL' found by system 'AL' and rawId '1234'<
/faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

### Пример 4. Клиент был закрыт

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByRawID>
      <sourceSystem>AL</sourceSystem>
      <rawId>1234</rawId>
      <partyType>PHYSICAL</partyType>
    </getByRawID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>Party of type 'PHYSICAL', system 'AL' and rawId '1234' has been closed<
/faultstring>
  </soap:Body>
</soap:Envelope>
```

```
</soap:Fault>
</soap:Body>
</soap:Envelope>
```

### Пример 5. Клиент со взаимосвязями обнаружен, в ответе возвращён статус подтверждаемой связи

В приведенном примере найден контрагент, который имеет две взаимосвязи:

1. найденный контрагент был ранее обнаружен в черном списке, но затем эта связь была отклонена дата-стюардом (HID = 2024) – в ответе возвращён статус связи, т.к. связь 40 подтверждаемая/отклоняемая;
2. найденный контрагент является собственником по отношению к юрлицу (HID = 3024) – статус связи не возвращается в ответе, т.к. связь неподтверждаемая.

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByRawID>
      <sourceSystem>AL</sourceSystem>
      <rawId>1</rawId>
      <partyType>PHYSICAL</partyType>
      <includeRelations>true</includeRelations>
    </getByRawID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHIDResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
        <attribute type="PHONE" hid="1024" rawId="AL.1.PH.1">
          <field name="actualityDate">23.08.2011</field>
          <field name="type">HOME</field>
          <field name="countryCode">7</field>
          <field name="cityCode">495</field>
          <field name="number">3542344</field>
        </attribute>
      </party>
    </getByHIDResponse>
  </soapenv:Body>
</soapenv:Envelope>
```



```

    <field name="extension"/>
    <field name="primary">true</field>
    <field name="additionalInfo"/>
    <field name="author">AL:1</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-354-23-44</field>
  </attribute>
  <relation type="500" hashCode="518fbfc8e87918dff5feb24bd78982d6">
    <first type="PHYSICAL" hid="2024">
      <field name="status">CONFIRMED</field>
    </relation>
    <relation type="11" hashCode="dcbf76d60b76e3b767c060c4eff9d868">
      <second type="LEGAL" hid="3024">
    </relation>
  </party>
</getByHIDResponse>
</soapenv:Body>
</soapenv:Envelope>

```

**Пример 6. Новых данных по клиенту нет (в запросе передан параметр *lastChangeTimestamp*)**

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;getByRawID&gt;       &lt;sourceSystem&gt;AL&lt;/sourceSystem&gt;       &lt;rawId&gt;1&lt;/rawId&gt;       &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;       &lt;lastChangeTimestamp&gt;1649142743&lt;/lastChangeTimestamp&gt;     &lt;/getByRawID&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ:

<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;getByRawIDResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;       &lt;notModified&gt;true&lt;/notModified&gt;     &lt;/getByRawIDResponse&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>
---

**Пример 7. Исходная карточка в составе золотой (в запросе передан параметр *lastChangeTimestamp*)**

В приведенном примере золотая карточка имеет метку времени последнего изменения больше, чем запрашиваемая исходная (например, в золотую карточку позже был влит ещё один контрагент).

В запросе rawID исходной карточки в составе золотой и lastChangeTimestamp исходной карточки.

В ответе золотая карточка и её lastChangeTimestamp.

Запрос:

```
Запрос  
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <getByRawID>  
      <sourceSystem>AL</sourceSystem>  
      <rawId>1</rawId>  
      <partyType>PHYSICAL</partyType>  
        <lastChangeTimestamp>1650366682</lastChangeTimestamp>  
      </getByRawID>  
    </soapenv:Body>  
  </soapenv:Envelope>
```

ОТВЕТ:

```
<getByRawIDResponse xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <party type="MOCK" hid="3">  
    <field name="blacklistCheckStatus">NOT_FOUND</field>  
    <field name="unifiedId"/>  
    <field name="createdDate"/>  
    <field name="surname">МОРОЗОВ</field>  
    <field name="name">СЕРГЕЙ</field>  
    <field name="patronymic">ВИТАЛЬЕВИЧ</field>  
    <field name="gender">MALE</field>  
    <field name="fullNameQC">EDITED</field>  
    <field name="fullNameAuthor">AL:1, AL:2</field>  
    <field name="fullNameRawSource">Морозов Сергей Витальевич</field>  
    <field name="surnameQc"/>  
    <field name="firstNameQc"/>  
    <field name="patronymicQc"/>  
    <field name="genderQc"/>  
    <field name="fullNameEditedChanges"/>  
    <field name="fullNameNSReasons"/>  
    <field name="terrorist">>false</field>  
    <field name="birthdate">18.03.1975</field>  
    <field name="birthdateAuthor">AL:1, AL:2</field>  
    <field name="birthdateQC">EDITED</field>  
    <field name="birthdatePopularity">NONE</field>  
    <field name="birthdateRawSource">18.03.1975</field>  
    <field name="inn"/>  
    <field name="innAuthor"/>  
    <field name="innQC">MISSED</field>  
    <field name="innRawSource"/>  
    <field name="snils"/>  
    <field name="snilsAuthor"/>  
    <field name="organization"/>  
    <field name="organizationAuthor"/>  
    <field name="additionalInfo"/>  
    <field name="additionalInfoAuthor"/>  
    <field name="maritalStatus">UNKNOWN</field>  
    <field name="maritalStatusAuthor"/>
```

```

<field name="monthlyAmount"/>
<field name="monthlyAmountAuthor"/>
<field name="dependantsCount"/>
<field name="dependantsCountAuthor"/>
<field name="cin"/>
<field name="cinAuthor"/>
<field name="marketingCategory">UNKNOWN</field>
<field name="marketingCategoryAuthor"/>
<field name="vipFlag">UNKNOWN</field>
<field name="vipFlagAuthor"/>
<field name="nextAdvertisingCallTime"/>
<field name="nextAdvertisingCallTimeAuthor"/>
<field name="firstAccountDate"/>
<field name="firstAccountDateAuthor"/>
<field name="corporateFlag">UNKNOWN</field>
<field name="corporateFlagAuthor"/>
<field name="closedAgreementFlag">UNKNOWN</field>
<field name="closedAgreementFlagAuthor"/>
<field name="hasInternetBankFlag">>false</field>
<field name="hasInternetBankFlagAuthor"/>
<field name="aliveFlag">>false</field>
<field name="aliveFlagAuthor"/>
<field name="aliveFlagActualityDate"/>
<field name="actualityDateAuthor"/>
<field name="actualityDate"/>
<field name="virtualFieldWithSearchScope">TRUE</field>
<field name="lastChangeTimestamp">1650963086</field>
<source hid="1" sourceSystem="AL" rawId="1"/>
<source hid="2" sourceSystem="AL" rawId="2"/>
</party>
</getByRawIDResponse>

```

## Примеры вызова PartyRA (JSON)

### Пример 1. Клиент обнаружен

Запрос:

```

{ "sourceSystem" : "AL",
  "rawId" : "1",
  "partyType" : "PHYSICAL" }

```

Ответ:

```

{"party": {
  "field": [
    {
      "value": "ИВАНОВ",
      "name": "surname"
    },
    {
      "value": "ИВАН",
      "name": "name"
    },
    {
      "value": "СЕРГЕЕВИЧ",
      "name": "patronymic"
    }
  ]
}

```

```

    },
    {
      "value": "MALE",
      "name": "gender"
    },
    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "AL:1",
      "name": "fullNameAuthor"
    },
    {
      "value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"
    },
    {
      "value": "24.06.2013",
      "name": "birthdate"
    },
    {
      "value": "AL:1",
      "name": "birthdateAuthor"
    },
    {
      "value": "EDITED",
      "name": "birthdateQC"
    },
    {
      "value": "24.06.2013",
      "name": "birthdateRawSource"
    },
    {
      "value": "",
      "name": "birthPlace"
    },
    {
      "value": "AL:1",
      "name": "birthPlaceAuthor"
    },
    {
      "value": "UNKNOWN",
      "name": "maritalStatus"
    },
    {
      "value": "AL:1",
      "name": "maritalStatusAuthor"
    },
    {
      "value": "AL:1",
      "name": "actualityDateAuthor"
    },
    {
      "value": "",
      "name": "actualityDate"
    }
  ],
  "hid": 1024,
  "sourceSystem": "AL",
  "rawId": "1",
  "attribute": [],

```

```
"source": [],  
"relation": [],  
"type": "PHYSICAL"  
}}
```

## Пример 2. Клиент обнаружен (объединённый)

Запрос:

```
{ "sourceSystem" : "AL",  
  "rawId" : "2",  
  "partyType" : "PHYSICAL" }
```

Ответ:

```
{  
  "party": {  
    "field": [  
      {  
        "value": "МОРОЗОВ",  
        "name": "surname"  
      },  
      {  
        "value": "СЕРГЕЙ",  
        "name": "name"  
      },  
      {  
        "value": "ВИТАЛЬЕВИЧ",  
        "name": "patronymic"  
      },  
      {  
        "value": "MALE",  
        "name": "gender"  
      },  
      {  
        "value": "EDITED",  
        "name": "fullNameQC"  
      },  
      {  
        "value": "BT:1",  
        "name": "fullNameAuthor"  
      },  
      {  
        "value": "Морозов Сергей Витальевич",  
        "name": "fullNameRawSource"  
      },  
      {  
        "value": "18.03.1975",  
        "name": "birthdate"  
      },  
      {  
        "value": "BT:1",  
        "name": "birthdateAuthor"  
      },  
      {  
        "value": "EDITED",  
        "name": "birthdateQC"  
      },  
      {  
        "value": "18.03.1975",
```

```

    "name": "birthdateRawSource"
  },
  {
    "value": "",
    "name": "birthPlace"
  },
  {
    "value": "BT:1",
    "name": "birthPlaceAuthor"
  },
  {
    "value": "UNKNOWN",
    "name": "maritalStatus"
  },
  {
    "value": "BT:1",
    "name": "maritalStatusAuthor"
  },
  {
    "value": "BT:1",
    "name": "actualityDateAuthor"
  },
  {
    "value": "",
    "name": "actualityDate"
  }
],
"hid": 1026,
"sourceSystem": null,
"rawId": null,
"attribute": [],
"source": [
  {
    "hid": 1024,
    "sourceSystem": "AL",
    "rawId": "2"
  },
  {
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
  }
],
"relation": [],
"type": "PHYSICAL"
}
}

```

### Пример 3. Обнаружен объединённый клиент, но возвращён исходный

Запрос:

```

{ "sourceSystem" : "AL",
  "rawId" : "2",
  "partyType" : "PHYSICAL",
  "onlySourceParty" : true }

```

Ответ:

```

{

```

```
"party": {
  "field": [
    {
      "value": "МОРОЗОВ",
      "name": "surname"
    },
    {
      "value": "СЕРГЕЙ",
      "name": "name"
    },
    {
      "value": "ВИТАЛЬЕВИЧ",
      "name": "patronymic"
    },
    {
      "value": "MALE",
      "name": "gender"
    },
    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "AL:2",
      "name": "fullNameAuthor"
    },
    {
      "value": "Морозов Сергей Витальевич",
      "name": "fullNameRawSource"
    },
    {
      "value": "19.03.1975",
      "name": "birthdate"
    },
    {
      "value": "AL:2",
      "name": "birthdateAuthor"
    },
    {
      "value": "EDITED",
      "name": "birthdateQC"
    },
    {
      "value": "19.03.1975",
      "name": "birthdateRawSource"
    },
    {
      "value": "",
      "name": "birthPlace"
    },
    {
      "value": "AL:2",
      "name": "birthPlaceAuthor"
    },
    {
      "value": "UNKNOWN",
      "name": "maritalStatus"
    },
    {
      "value": "AL:2",
      "name": "maritalStatusAuthor"
    }
  ]
}
```

```
{
  "value": "AL:2",
  "name": "actualityDateAuthor"
},
{
  "value": "",
  "name": "actualityDate"
}
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "2",
"attribute": [],
"source": [],
"relation": [],
"type": "PHYSICAL"
}
}
```



# PARTY.GETBYHID

Метод позволяет получить контрагента по его идентификатору в Едином клиенте.

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
  - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Клиент обнаружен
  - Пример 2. Клиент обнаружен (объединенный)
  - Пример 3. Клиента с указанным идентификатором не существует
  - Пример 4. Клиент был полностью разъединен
  - Пример 5. Клиент был закрыт
  - Пример 6. Клиент со взаимосвязями обнаружен, в ответе возвращён статус подтверждаемой связи
  - Пример 7. Клиент обнаружен (в запросе передан тип контрагента)
  - Пример 8. В запросе передан lastChangeTimestamp, изменений по контрагенту не было
- Примеры вызова PartyRA (JSON)
  - Пример 1. Клиент обнаружен
  - Пример 2. Клиент обнаружен (объединенный)
  - Пример 3. Обнаружен объединенный клиент, но возвращен исходный
  - Пример 4. Обнаружен объединенный клиент, но исходного клиента с указанным идентификатором не существует

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
hid	long	да	Уникальный идентификатор контрагента в Едином клиенте (HID).
includeRelations	boolean	нет	Возвращать взаимосвязи контрагента. По умолчанию принимает значение <b>false</b> .
partyType	Тип контрагента	нет	Тип контрагента
onlySourceParty	boolean	нет	Возвращать исходного контрагента, даже если он был объединен в результате слияния. По умолчанию принимает значение <b>false</b> .
include	Тип	нет	Дополнительная информация о контрагенте

/partyInfo[]	информации о контрагенте		
lastChangeTimestamp	number	нет	Время последнего изменения КА

## Результирующие данные

Имя параметра	Тип	Описание
party	WParty	Информация по контрагенту.

## Особенности использования

Если не указан тип контрагента, и существует контрагент и договор (или аналогичный объект, не являющийся "честным" контрагентом) будут возвращены данные контрагента.

Если контрагент был объединен в результате слияния и параметр *onlySourceParty* имеет значение **FALSE**, то при вызове метода с идентификатором HID исходного контрагента система вернет только объединенного (но не исходного) контрагента.

Если параметр *onlySourceParty* имеет значение **TRUE**, то при вызове метода с идентификатором HID исходного контрагента система вернет именно исходного контрагента, даже если он был объединен в результате слияния.

Для объединенных клиентов система должна возвращать информацию о привязке к своим исходным контрагентам (**//party/source**).

Если параметр *includeRelations* имеет значение **true**, то система вернет все взаимосвязи для данного контрагента. В ответе для взаимосвязей **//party/relation** будет указана только противоположная запрашиваемому контрагенту сторона взаимосвязи. При этом для связанного объекта в ответе будет указан только его тип и уникальный идентификатор в CDI (HID). А если связь между объектами подтверждаемая, то в ответе вернется также статус подтверждения связи (см Пример 6).

Для объединенного контрагента система вернет все взаимосвязи его исходных контрагентов.

Если указанный контрагент связан с контрагентом, который был объединен, то система вернет связь с объединенным контрагентом.

Если параметр *lastChangeTimestamp* указан в запросе и имеет значение, совпадающее с *lastChangeTimestamp* в таблице **PARTY\_CHANGE\_META**, то возвращается пустой информационный ответ «Изменений нет» (см пример 8).

Если параметр *lastChangeTimestamp* не указан в запросе, либо если его значение не совпадает со значением *lastChangeTimestamp* в таблице **PARTY\_CHANGE\_META**, то возвращается запрашиваемая карточка контрагента и временная метка последних изменений карточки.

## Исключительные ситуации

Если контрагента с указанным идентификатором в системе не существует, то система вернет ошибку.

Если указанный контрагент был полностью разъединен, то система вернет ошибку.

Если указанный контрагент был закрыт, то система вернет ошибку.

Если параметр *onlySourceParty* имеет значение **TRUE** и указанный контрагент является объединенным, то система вернет ошибку.

## Примеры вызова PartyWS (SOAP)

### Пример 1. Клиент обнаружен

Запрос:

```


Запрос



```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1534</hid>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>
```


```

Ответ:

```


```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHIDResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
        <attribute type="PHONE" hid="1024" rawId="AL.1.PH.1">
          <field name="actualityDate">23.08.2011</field>
          <field name="type">HOME</field>
          <field name="countryCode">7</field>
          <field name="cityCode">495</field>
          <field name="number">3542344</field>
          <field name="extension"/>
          <field name="primary">true</field>
          <field name="additionalInfo"/>
          <field name="author">AL:1</field>
          <field name="qualityCode">GOOD</field>
          <field name="rawSource">8-495-354-23-44</field>
        </attribute>
      </party>
    </getByHIDResponse>
  </soapenv:Body>
</soapenv:Envelope>
```


```

```
</party>
</getByHIDResponse>
</soapenv:Body>
</soapenv:Envelope>
```

## Пример 2. Клиент обнаружен (объединенный)

Запрос:

```
Запрос
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1026</hid>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHIDResponse>
      <party type="PHYSICAL" hid="1026">
        <field name="surname">МОРОЗОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИТАЛЬЕВИЧ</field>
        <field name="gender">MALE</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameAuthor">BT:1</field>
        <field name="fullNameRawSource">Морозов Сергей Витальевич</field>
        <field name="birthdate">18.03.1975</field>
        <field name="birthdateAuthor">BT:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">18.03.1975</field>
        <field name="inn"/>
        <field name="innAuthor"/>
        <field name="innQC">MISSED</field>
        <field name="organization"/>
        <field name="organizationAuthor"/>
        <field name="additionalInfo"/>
        <field name="additionalInfoAuthor"/>
        <field name="terrorist">false</field>
        <field name="maritalStatus">UNKNOWN</field>
        <field name="maritalStatusAuthor"/>
        <field name="actualityDateAuthor"/>
        <field name="actualityDate"/>
        <source hid="1024" sourceSystem="AL" rawId="1"/>
        <source hid="1025" sourceSystem="BT" rawId="1"/>
      </party>
    </getByHIDResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 3. Клиента с указанным идентификатором не существует

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1234</hid>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>No party version found for 'PHYSICAL' HID 1234</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

### Пример 4. Клиент был полностью разъединен

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1234</hid>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>Unexpected merged status 'UNMERGED' for party with HID 1234</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## Пример 5. Клиент был закрыт

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1234</hid>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>No party found with HID 1234</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## Пример 6. Клиент со взаимосвязями обнаружен, в ответе возвращён статус подтверждаемой связи

В приведенном примере найден контрагент, который имеет две взаимосвязи:

1. найденный контрагент был ранее обнаружен в черном списке, но затем эта связь была отклонена дата-стюардом (HID = 2024) – в ответе возвращён статус связи, т.к. связь 40 подтверждаемая/отклоняемая;
2. найденный контрагент является собственником по отношению к юрлицу (HID = 3024) – статус связи не возвращается в ответе, т.к. связь неподтверждаемая.

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1024</hid>
      <includeRelations>true</includeRelations>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHIDResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
        <attribute type="PHONE" hid="1024" rawId="AL.1.PH.1">
          <field name="actualityDate">23.08.2011</field>
          <field name="type">HOME</field>
          <field name="countryCode">7</field>
          <field name="cityCode">495</field>
          <field name="number">3542344</field>
          <field name="extension"/>
          <field name="primary">true</field>
          <field name="additionalInfo"/>
          <field name="author">AL:1</field>
          <field name="qualityCode">GOOD</field>
          <field name="rawSource">8-495-354-23-44</field>
        </attribute>
        <relation type="500" hashCode="518fbfc8e87918dff5feb24bd78982d6">
          <first type="PHYSICAL" hid="2024">
            <field name="status">CONFIRMED</field>
          </relation>
          <relation type="11" hashCode="dcbf76d60b76e3b767c060c4eff9d868">
            <second type="LEGAL" hid="3024"/>
          </relation>
        </party>
      </getByHIDResponse>
    </soapenv:Body>
  </soapenv:Envelope>

```

### Пример 7. Клиент обнаружен (в запросе передан тип контрагента)

Запрос:

```

Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHID>
      <hid>1024</hid>
      <partyType>PHYSICAL</partyType>
    </getByHID>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getByHIDResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
        <attribute type="PHONE" hid="1024" rawId="AL.1.PH.1">
          <field name="actualityDate">23.08.2011</field>
          <field name="type">HOME</field>
          <field name="countryCode">7</field>
          <field name="cityCode">495</field>
          <field name="number">3542344</field>
          <field name="extension"/>
          <field name="primary">true</field>
          <field name="additionalInfo"/>
          <field name="author">AL:1</field>
          <field name="qualityCode">GOOD</field>
          <field name="rawSource">8-495-354-23-44</field>
        </attribute>
      </party>
    </getByHIDResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

*Пример 8. В запросе передан lastChangeTimestamp, изменений по контрагенту не было*

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns=" http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;getByHID&gt;       &lt;hid&gt;1234&lt;/hid&gt;       &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;       &lt;lastChangeTimestamp&gt;1649142743&lt;/lastChangeTimestamp&gt;     &lt;/soapenv:Body&gt;   &lt;/soapenv:Envelope&gt; </pre>

Ответ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getByHIDResponse>
      <notModified>true</notModified>

```



```
</getByHIDResponse>
</soap:Body>
</soap:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Клиент обнаружен

Запрос:

```
{ "hid" : 1024 }
```

Ответ:

```
{ "party": {
  "field": [
    {
      "value": "ИВАНОВ",
      "name": "surname"
    },
    {
      "value": "ИВАН",
      "name": "name"
    },
    {
      "value": "СЕРГЕЕВИЧ",
      "name": "patronymic"
    },
    {
      "value": "MALE",
      "name": "gender"
    },
    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "AL:1",
      "name": "fullNameAuthor"
    },
    {
      "value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"
    },
    {
      "value": "24.06.2013",
      "name": "birthdate"
    },
    {
      "value": "AL:1",
      "name": "birthdateAuthor"
    },
    {
      "value": "EDITED",
      "name": "birthdateQC"
    },
    {
```

```

    "value": "24.06.2013",
    "name": "birthdateRawSource"
  },
  {
    "value": "",
    "name": "birthPlace"
  },
  {
    "value": "AL:1",
    "name": "birthPlaceAuthor"
  },
  {
    "value": "UNKNOWN",
    "name": "maritalStatus"
  },
  {
    "value": "AL:1",
    "name": "maritalStatusAuthor"
  },
  {
    "value": "AL:1",
    "name": "actualityDateAuthor"
  },
  {
    "value": "",
    "name": "actualityDate"
  }
},
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [],
"source": [],
"relation": [],
"type": "PHYSICAL"
}}

```

## Пример 2. Клиент обнаружен (объединенный)

Запрос:

```
{ "hid" : 1025 }
```

Ответ:

```

{
  "party": {
    "field": [
      {
        "value": "МОРОЗОВ",
        "name": "surname"
      },
      {
        "value": "СЕРГЕЙ",
        "name": "name"
      },
      {
        "value": "ВИТАЛЬЕВИЧ",
        "name": "patronymic"
      }
    ]
  }
}

```

```

{
  "value": "MALE",
  "name": "gender"
},
{
  "value": "EDITED",
  "name": "fullNameQC"
},
{
  "value": "BT:1",
  "name": "fullNameAuthor"
},
{
  "value": "Морозов Сергей Витальевич",
  "name": "fullNameRawSource"
},
{
  "value": "18.03.1975",
  "name": "birthdate"
},
{
  "value": "BT:1",
  "name": "birthdateAuthor"
},
{
  "value": "EDITED",
  "name": "birthdateQC"
},
{
  "value": "18.03.1975",
  "name": "birthdateRawSource"
},
{
  "value": "",
  "name": "birthPlace"
},
{
  "value": "BT:1",
  "name": "birthPlaceAuthor"
},
{
  "value": "UNKNOWN",
  "name": "maritalStatus"
},
{
  "value": "BT:1",
  "name": "maritalStatusAuthor"
},
{
  "value": "BT:1",
  "name": "actualityDateAuthor"
},
{
  "value": "",
  "name": "actualityDate"
}
},
"hid": 1027,
"sourceSystem": null,
"rawId": null,
"attribute": [],
"source": [

```

```

{
  "hid": 1025,
  "sourceSystem": "AL",
  "rawId": "2"
},
{
  "hid": 1026,
  "sourceSystem": "BT",
  "rawId": "1"
}
],
"relation": [],
"type": "PHYSICAL"
}
}

```

### Пример 3. Обнаружен объединенный клиент, но возвращен исходный

Запрос:

```

{ "hid" : 1025,
  "onlySourceParty" : true }

```

Ответ:

```

{
  "party": {
    "field": [
      {
        "value": "МОРОЗОВ",
        "name": "surname"
      },
      {
        "value": "СЕРГЕЙ",
        "name": "name"
      },
      {
        "value": "ВИТАЛЬЕВИЧ",
        "name": "patronymic"
      },
      {
        "value": "MALE",
        "name": "gender"
      },
      {
        "value": "EDITED",
        "name": "fullNameQC"
      },
      {
        "value": "AL:2",
        "name": "fullNameAuthor"
      },
      {
        "value": "Морозов Сергей Витальевич",
        "name": "fullNameRawSource"
      },
      {
        "value": "19.03.1975",
        "name": "birthdate"
      }
    ]
  }
}

```

```

{
  "value": "AL:2",
  "name": "birthdateAuthor"
},
{
  "value": "EDITED",
  "name": "birthdateQC"
},
{
  "value": "19.03.1975",
  "name": "birthdateRawSource"
},
{
  "value": "",
  "name": "birthPlace"
},
{
  "value": "AL:2",
  "name": "birthPlaceAuthor"
},
{
  "value": "UNKNOWN",
  "name": "maritalStatus"
},
{
  "value": "AL:2",
  "name": "maritalStatusAuthor"
},
{
  "value": "AL:2",
  "name": "actualityDateAuthor"
},
{
  "value": "",
  "name": "actualityDate"
}
},
"hid": 1025,
"sourceSystem": "AL",
"rawId": "2",
"attribute": [],
"source": [],
"relation": [],
"type": "PHYSICAL"
}
}

```

*Пример 4. Обнаружен объединенный клиент, но исходного клиента с указанным идентификатором не существует*

Запрос:

```

{ "hid" : 1027,
  "onlySourceParty" : true }

```

Ответ:

```

{
  "errorType": "IllegalArgumentException",

```

```
"errorMessage": "Error searching by HID 1027 and partyType ": Requested party  
(hid=1027) is resulted but onlySourceParty flag is set"  
}
```

# PARTY.SEARCH

Метод позволяет найти всех контрагентов, удовлетворяющих заданным критериям.

- Описание

- Входные параметры

- Результирующие данные

- Особенности использования

- Дополнительная информация о контрагенте

- Исключительные ситуации

- Примеры вызова PartyWS (SOAP)

- Пример 1. Поиск физического лица без дополнительных параметров

- Пример 2. Поиск клиентов без указания типа контрагента (partyType)

- Пример 3. Ничего не найдено (пустой результат поиска)

- Пример 4. Ошибка, когда найдено слишком много контрагентов

- Пример 5. Ошибка, когда передан слишком длинный запрос

- Пример 6. Контрагенты найдены по адресу "Ул. Ленина, д.1, корп. 1, кв 55"

- Пример 7. Поиск по телефону — контрагенты найдены

- Пример 8. Поиск по паспорту — контрагенты найдены

- Пример 9. Возврат ссылок на исходные записи в результатах поиска.

- Пример 10. Возврат реквизитов и связей клиентов в результатах поиска

- Пример 11. Поиск с фильтрацией "Ручная обработка: телефоны"

- Пример 12. Поиск с фильтрацией "Ручная обработка: телефоны", ничего не найдено

- Пример 13. Расширенный поиск по разным сущностям

- Пример 14. Возврат прошлых значений

- Пример 15. Ошибка, когда найдено контрагентов больше maxCount

- Примеры вызова PartyRA (JSON)

- Пример 1. Контрагенты найдены

- Пример 2. Возврат ссылок на исходные записи в результатах поиска.

- Пример 3. Запрос для поиска физлица по ФИО и ИНН

- Пример 4. Запрос для поиска юрлица по наименованию, ИНН, ОГРН, КПП

- Пример 5. Ошибка авторизации

- Пример 6. Ошибка, когда найдено слишком много результатов.

- Пример 7. Ошибка, когда передан слишком длинный запрос

- Пример 8. Расширенный поиск по разным сущностям

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
query	string	да	Критерии поиска
restriction	Фильтр	нет	Дополнительная фильтрация данных
partyType	WPartyType	нет	Тип контрагента
maxCount	int	нет	Максимальное количество возвращаемых записей (не более 2000 — параметр <code>webservice.maxPartiesCount</code> )
include /partyInfo[]	Тип информации о контрагенте	нет	Дополнительная информация о контрагенте

### Результирующие данные

Имя параметра	Тип	Описание
party	WParty	Информация по контрагенту (если явно не заданы параметр <code>include/partyInfo[]</code> , то не содержит множественные атрибуты — адреса, телефоны, документы и т.д.)

### Особенности использования

Поиск осуществляется по всем атрибутам контрагента (ФИО, дата рождения, адреса, документы, и т.д.) При этом в ответе по каждому контрагенту возвращаются только базовые поля, без множественных атрибутов.

Поиск осуществляется только по нормализованным значениям.

Если параметр `//restriction` задан, то система вернет только тех контрагентов, которые удовлетворяют заданному в фильтре критерию.

Если задан параметр `maxCount` и количество найденных контрагентов меньше или равно `maxCount`, система вернет в ответе всех контрагентов.

Если количество найденных контрагентов больше `maxCount`, система вернет ошибку (см. пример 15 ниже).

Если по заданным критериям ничего не будет найдено, система вернет пустой ответ (см. пример 2 ниже).

При поиске контрагентов по атрибутам следует учитывать следующие особенности:

#### 1. Возможные форматы строки для телефона:

- "{код страны} {код} {номер телефона}", например, "7 926 1234567";
- "{код} {номер телефона}", например, "926 1234567"; (работает с версии 3.14)



- "{номер телефона}", например, "1234567";
2. **Серию и номер документа** в строке поиска следует указывать точно так же, как они сохранены в карточке контрагента. Пример строки поиска для прошедшего стандартизацию паспорта РФ: "45 06 123456".
  3. **Адрес** должен быть указан без точек и запятых, через пробел. Например, клиента с адресом "ул. Ленина, д. 1, к. 1, кв 55" можно найти двумя способами:
    - в строке поиска указать только название улицы: "Ленина";
    - по полному адресу, разделенному пробелами: "Ленина 1 1 55".

В качестве критерия для поиска возможно задание условия **расширенного поиска**:  
 "?PhysicalParty.surname=Пашук AND PhysicalParty.name=Александр AND PhysicalParty.patronymic=Геннадьевич AND Passport.documentSeries=XXVIMЮ AND Passport.documentNumber=734419 AND PhysicalParty.birthdate=21.04.196"

## Дополнительная информация о контрагенте

Если задан параметр **//include**, то система вернет только те данные контрагента, которые в нем перечислены.

Значения параметра <b>//include /partyInfo</b>	Описание	Комментарий
REQUISITE	Информация о реквизитах	
ATTRIBUTE	Информация об атрибутах	Информация о соглашениях возвращается только если для параметра <b>webservice.response.attributes.defaultValues</b> выставлено значение <b>true</b> .
SOURCE	Информация о привязке к исходным контрагентам	
RELATION	Информация о связях	Возвращает тип связи, информацию о связанной сущности. Для подтверждаемых связей возвращает также статус связи (пример 10).
PAST_ATTRIBUTE	Информация о прошлых значениях ФИО и документов	<p>Возврат прошлых значений зависит только от заданного <b>include PAST_ATTRIBUTE</b> и не зависит от параметра <b>webservice.response.attributes.pastValues</b>.</p> <p>Информация о найденных прошлых значениях возвращается в сущности <b>pastAttribute</b>.</p> <p>Если прошлые значения не найдены, блок полей в ответе возвращается пустым, ошибки нет.</p>

## Исключительные ситуации

- Если по заданным критериям поиска найдено более 2000 (**webservice.maxPartiesCount**) контрагентов, то система вернет ошибку (см. пример 4).
- Если длина критерия поиска превышает 2000 символов, то система вернет ошибку.

- Если maxCount задан больше 2000 (webservice.maxPartiesCount), количество найденных контрагентов проверяется по параметру webservice.maxPartiesCount. Дополнительно в логе будет записано предупреждение: **WARN PartyWebService - Request has maxCount=N but configured limit is X. Custom maxCount will be ignored.**

## Примеры вызова PartyWS (SOAP)

### Пример 1. Поиск физического лица без дополнительных параметров

Запрос:

```


Запрос


<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>мамонов СЕРГЕЙ</query>
      <partyType>PHYSICAL</partyType>
    </search>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">Мамонов</field>
        <field name="name">Сергей</field>
        <field name="patronymic">Викторович</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">Мамонов</field>
        <field name="name">Сергей</field>
        <field name="patronymic">Игоревич</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов СЕРГЕЙ Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <field name="maritalStatus">SINGLE_MALE</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

    <field name="maritalStatusAuthor">AL:2</field>
  </party>
</searchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

В данном примере у контрагента с hid = 1024 в поле "исходное ФИО" указано значение, отличное от стандартизованных значений:

- "Исходное ФИО" — Мамонов Сергей Викторович
- Фамилия, Имя и Отчество (нормализованные значения) — Мамонов Сергей Викторович

Так как поиск ведется по нормализованным значениям, контрагент был найден. Если бы в запросе была "Сергей", поиск вернул бы пустой результат (не нашел бы ни одного из клиентов).

## Пример 2. Поиск клиентов без указания типа контрагента (partyType)

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;search&gt;       &lt;query&gt;мамонов&lt;/query&gt;     &lt;/search&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ (часть полей опущена для краткости) — найдены как ФЛ «Мамонов Сергей Викторович», так и ЮЛ — «ОАО "Мамонов"»

<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;searchResponse&gt;       &lt;party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"&gt;         &lt;field name="actualityDate"&gt;22.02.2012&lt;/field&gt;         &lt;field name="surname"&gt;МАМОНОВ&lt;/field&gt;         &lt;field name="name"&gt;СЕРГЕЙ&lt;/field&gt;         &lt;field name="patronymic"&gt;ВИКТОРОВИЧ&lt;/field&gt;         &lt;field name="fullNameAuthor"&gt;AL:1&lt;/field&gt;         &lt;field name="fullNameQC"&gt;EDITED&lt;/field&gt;         &lt;field name="fullNameRawSource"&gt;Иванов Сергей Викторович&lt;/field&gt;         &lt;field name="birthdate"&gt;03.12.1987&lt;/field&gt;         &lt;field name="birthdateAuthor"&gt;AL:1&lt;/field&gt;         &lt;field name="birthdateQC"&gt;EDITED&lt;/field&gt;         &lt;field name="birthdateRawSource"&gt;03.12.1987&lt;/field&gt;       &lt;/party&gt;       &lt;party type="LEGAL" hid="1025" sourceSystem="AL" rawId="2"&gt;         &lt;field name="actualityDate"&gt;22.02.2012&lt;/field&gt;         &lt;field name="fullNameRawSource"&gt;ОАО "Мамонов"&lt;/field&gt;         &lt;field name="fullNameQC"&gt;EDITED&lt;/field&gt;       &lt;/party&gt;     &lt;/searchResponse&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>
---

```

    <field name="fullNameAuthor">AL:2</field>
      <field name="shortName">Мамонов</field>
      <field name="shortNameAuthor">AL:2</field>
    <field name="foreignName"/>
    <field name="foreignNameAuthor">AL:2</field>
    <field name="inn">7710353606</field>
    <field name="innAuthor">AL:2</field>
    <field name="innQC">EDITED</field>
    <field name="innRawSource">7710353606</field>
  </party>
</searchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

### Пример 3. Ничего не найдено (пустой результат поиска)

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;search&gt;       &lt;query&gt;-&lt;/query&gt;       &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;     &lt;/search&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

ОТВЕТ:

<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_12"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;searchResponse&gt;&lt;/searchResponse&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>
---

### Пример 4. Ошибка, когда найдено слишком много контрагентов

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;search&gt;       &lt;query&gt;K%&lt;/query&gt;       &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;     &lt;/search&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

ОТВЕТ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>Too many parties found (expected no more than 2000, got 3124)<
    /faultstring>
  </soap:Fault>
</soap:Body>
</soap:Envelope>

```

### Пример 5. Ошибка, когда передан слишком длинный запрос

Запрос:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>

```

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз.

```

      </query>
    </search>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:INVALID_REQUEST<
    /faultcode>
      <faultstring>Invalid 'Search' field length (expected no greater than 2000, got 2160)<

```

```

/faultstring>
  <detail>
    <errorMessage xmlns="http://hflabs.ru/cdi/soap/2_13">
      <errorType>IllegalArgumentException</errorType>
      <errorMessage>Error searching by param '[Personal data #5]': Invalid 'Search' field
length (expected no greater than 2000, got 2160)</errorMessage>
    </errorMessage>
  </detail>
</soap:Fault>
</soap:Body>
</soap:Envelope>

```

### Пример 6. Контрагенты найдены по адресу "Ул. Ленина, д.1, корп. 1, кв 55"

Запрос:

```

Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>Ленина 1 1 55</query>
      <partyType>PHYSICAL</partyType>
    </search>
  </soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

    <field name="birthdateRawSource"/>
    <field name="maritalStatus">SINGLE_MALE</field>
    <field name="maritalStatusAuthor">AL:2</field>
  </party>
</searchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Напомним, что возвращается только информация по самому контрагенту, без атрибутов.

### Пример 7. Поиск по телефону — контрагенты найдены

Запрос:

```

          Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>79261234567</query>
      <partyType>PHYSICAL</partyType>
    </search>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <field name="maritalStatus">SINGLE_MALE</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

    <field name="maritalStatusAuthor">AL:2</field>
  </party>
</searchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Пример 8. Поиск по паспорту — контрагенты найдены

Запрос:

```

                                Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>45 06 123456</query>
      <partyType>PHYSICAL</partyType>
    </search>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <field name="maritalStatus">SINGLE_MALE</field>
        <field name="maritalStatusAuthor">AL:2</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>

```



```
</searchResponse>
</soapenv:Body>
</soapenv:Envelope>
```

### Пример 9. Возврат ссылок на исходные записи в результатах поиска.

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>45 06 123456</query>
      <partyType>PHYSICAL</partyType>
      <include>
        <partyInfo>SOURCE</partyInfo>
      </include>
    </search>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1026">
        <source hid="1024" sourceSystem="AL" rawId="101"/>
        <source hid="1025" sourceSystem="BT" rawId="101"/>
      </party>
      <party type="PHYSICAL" hid="1099" sourceSystem="AL" rawId="1"/>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

В результате найдены два контрагента: один объединенный, второй исходный.

### Пример 10. Возврат реквизитов и связей клиентов в результатах поиска

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>45 06 123456</query>
      <partyType>PHYSICAL</partyType>
      <include>
        <partyInfo>REQUISITE</partyInfo>
        <partyInfo>RELATION</partyInfo>
      </include>
    </search>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
        <relation type="500" hashCode="518bfc8e87918dff5feb24bd78982d6">
          <first type="PHYSICAL" hid="2024"/>
          <field name="status">UNKNOWN</field>
        </relation>
        <relation type="11" hashCode="dcbf76d60b76e3b767c060c4eff9d868">
          <second type="LEGAL" hid="3024"/>
        </relation>
      </party>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <field name="maritalStatus">SINGLE_MALE</field>
        <field name="maritalStatusAuthor">AL:2</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

В результате найдены два контрагента: первый со связями, второй без.

### **Пример 11. Поиск с фильтрацией "Ручная обработка: телефоны"**

Запрос: ищем клиентов с номером телефона "+7 926 123 45 67", у которых есть телефоны, отправленные после стандартизации на ручной разбор (получили плохой качества).

#### **Запрос**

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
```

```

<soapenv:Header/>
<soapenv:Body>
  <search>
    <query>79261234567</query>
    <restriction>MANUAL_PHONE</restriction>
    <partyType>PHYSICAL</partyType>
  </search>
</soapenv:Body>
</soapenv:Envelope>

```

Ответ (возвращаются только контрагенты, попадающие под указанный фильтр):

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="2">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">ПЕЧОРИН</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Мамонов Олег Игоревич</field>
        <field name="birthdate"/>
        <field name="birthdateAuthor">AL:2</field>
        <field name="birthdateQC">NOT_SURE</field>
        <field name="birthdateRawSource"/>
        <field name="maritalStatus">SINGLE_MALE</field>
        <field name="maritalStatusAuthor">AL:2</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

### Пример 12. Поиск с фильтрацией "Ручная обработка: телефоны", ничего не найдено

Дано: телефон "+7 926 111 22 33" есть только у одного клиента. Телефон при стандартизации получил хороший код качества (GOOD), поэтому он не попадает под фильтр "Ручная обработка: телефоны".

Запрос:

---

### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>79261112233</query>
      <restriction>MANUAL_PHONE</restriction>
      <partyType>PHYSICAL</partyType>
    </search>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <searchResponse xmlns="http://hflabs.ru/cdi/soap/2_13"/>
  </soap:Body>
</soap:Envelope>
```

### Пример 13. Расширенный поиск по разным сущностям

Запрос:

### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>? PhysicalParty.surname=МАМОХОВ or LegalParty.inn=7705974943</query>
    </search>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ (часть полей опущена для краткости):

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse>
      <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">МАМОХОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateAuthor">AL:1</field>
        <field name="birthdateQC">EDITED</field>
        <field name="birthdateRawSource">03.12.1987</field>
          <field name="inn">7705974943</field>
          <field name="innAuthor">performer</field>
          <field name="innQC">CONFIRMED_MANUALLY</field>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <field name="innRawSource">7705974943</field>
  </party>
  <party type="LEGAL" hid="1025" sourceSystem="AL" rawId="2">
    <field name="actualityDate">22.02.2012</field>
    <field name="fullNameRawSource">ОАО "Мамонов"</field>
    <field name="fullNameQC">EDITED</field>
    <field name="fullNameAuthor">AL:2</field>
    <field name="shortName">Мамонов</field>
    <field name="shortNameAuthor">AL:2</field>
    <field name="foreignName"/>
    <field name="foreignNameAuthor">AL:2</field>
    <field name="inn">7705974943</field>
    <field name="innAuthor">AL:2</field>
    <field name="innQC">EDITED</field>
    <field name="innRawSource">7705974943</field>
  </party>
</searchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

### Пример 14. Возврат прошлых значений

Дано: контрагент с прошлыми ФИО и паспортом, поиск по прошлому имени, в блоке **include** есть параметр **PAST\_ATTRIBUTE**

Запрос:

```

          Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search xmlns="http://hflabs.ru/cdi/soap/2_13">
      <query>Сергей</query>
      <include>
        <partyInfo>REQUISITE</partyInfo>
        <partyInfo>ATTRIBUTE</partyInfo>
        <partyInfo>PAST_ATTRIBUTE</partyInfo>
      </include>
    </search>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ: информация о прошлых значениях возвращается как **pastAttribute**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <party hid="1" type="PHYSICAL">
        <field name="fullNameAuthor">AL:1</field>
        <field name="fullNameQC">EDITED</field>
        <field name="fullNameRawSource">Морозов Иван Витальевич</field>
        <field name="surname">МОРОЗОВ</field>
        <field name="name">ИВАН</field>
        <field name="patronymic">ВИТАЛЬЕВИЧ</field>
        <field name="actualityDate"/>
        <field name="actualityDateAuthor"/>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<field name="additionalInfo"/>
<field name="additionalInfoAuthor"/>
<field name="birthdate">18.03.1975</field>
<field name="birthdateAuthor">AL:1</field>
<field name="birthdateQC">EDITED</field>
<field name="birthdateRawSource">18.03.1975</field>
<field name="gender">MALE</field>
<field name="inn"/>
<field name="innAuthor"/>
<field name="innQC">MISSED</field>
<field name="maritalStatus">UNKNOWN</field>
<field name="maritalStatusAuthor"/>
<field name="monthlyAmount"/>
<field name="monthlyAmountAuthor"/>
<field name="organization"/>
<field name="organizationAuthor"/>
<field name="terrorist">>false</field>
<attribute type="DOCUMENT_PASSPORT" hid="22" rawId="1.DOC.2">
  <field name="actualityDate"/>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
  <field name="departmentCode">771-819</field>
  <field name="documentNumber">123445</field>
  <field name="documentSeries">4508</field>
  <field name="expiryDate"/>
  <field name="issueAuthority">ОВД Бибирево гор. Москвы</field>
  <field name="issueDate">01.09.2010</field>
  <field name="primary">>false</field>
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource">Паспорт РФ: 4508 123445, выдан 01.09.2010
ОВД Бибирево гор. Москвы, код 771-819</field>
  <field name="type">PASSPORT</field>
</attribute>
<pastAttribute type="DOCUMENT_PASSPORT" hid="33" rawId="AL.DOC.1">
  <field name="actualityDate">05.10.2012</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
  <field name="departmentCode">771-819</field>
  <field name="documentNumber">631945</field>
  <field name="documentSeries">45 07</field>
  <field name="expiryDate"/>
  <field name="invalidStatus">ACTUAL</field>
  <field name="issueAuthority">ОВД Бибирево гор. Москвы</field>
  <field name="issueDate">13.08.2007</field>
  <field name="popularity">NONE</field>
  <field name="primary">>false</field>
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource"/>
  <field name="type">PASSPORT</field>
</pastAttribute>
<pastAttribute type="FIO" hid="22">
  <field name="actualityDate">01.10.2011</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
  <field name="primary">>false</field>
  <field name="qualityCode">EDITED</field>
  <field name="rawSource">Морозов Сергей Витальевич</field>
  <field name="surname">МОРОЗОВ</field>
  <field name="name">СЕРГЕЙ</field>
  <field name="patronymic">ВИТАЛЬЕВИЧ</field>

```

```
        </pastAttribute>
      </party>
    </searchResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 15. Ошибка, когда найдено контрагентов больше maxCount

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <search>
      <query>K%</query>
      <partyType>PHYSICAL</partyType>
      <maxCount>5</maxCount>
    </search>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>Too many parties found (expected no more than 5, got 15)</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Контрагенты найдены

Запрос:

```
{ "query" : "Иванов" }
```

Ответ:

```
{ "party": [
  {
    "field": [
      {
        "value": "ИВАНОВ",
        "name": "surname"
      },
      {
        "value": "ИВАН",
        "name": "name"
      }
    ]
  }
]
```

```

    {
      "value": "СЕРГЕЕВИЧ",
      "name": "patronymic"
    },
    {
      "value": "MALE",
      "name": "gender"
    },
    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "AL:1",
      "name": "fullNameAuthor"
    },
    {
      "value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"
    },
    {
      "value": "24.06.2013",
      "name": "birthdate"
    },
    {
      "value": "AL:1",
      "name": "birthdateAuthor"
    },
    {
      "value": "EDITED",
      "name": "birthdateQC"
    },
    {
      "value": "24.06.2013",
      "name": "birthdateRawSource"
    },
    {
      "value": "",
      "name": "birthPlace"
    },
    {
      "value": "AL:1",
      "name": "birthPlaceAuthor"
    },
    {
      "value": "UNKNOWN",
      "name": "maritalStatus"
    },
    {
      "value": "AL:1",
      "name": "maritalStatusAuthor"
    },
    {
      "value": "AL:1",
      "name": "actualityDateAuthor"
    },
    {
      "value": "",
      "name": "actualityDate"
    }
  ],
  "hid": 1024,

```



```
"sourceSystem": "AL",
"rawId": "1",
"attribute": [],
"source": [],
"relation": [],
"type": "PHYSICAL"
}}
```

### *Пример 2. Возврат ссылок на исходные записи в результатах поиска.*

Запрос:

```
{
  "query": "45 06 123456", "partyType": "PHYSICAL", "include": {"partyInfo": ["SOURCE"]}
}
```

Ответ:

```
{"party": [
  {
    "hid": 1026,
    "sourceSystem": null,
    "rawId": null,
    "source": [
      {
        "hid": 1024,
        "sourceSystem": "AL",
        "rawId": "101"
      },
      {
        "hid": 1025,
        "sourceSystem": "BT",
        "rawId": "101"
      }
    ]
  },
  {
    "type": "PHYSICAL"
  }
]}
```

### *Пример 3. Запрос для поиска физлица по ФИО и ИНН*

Запрос:

```
{"query": "\"? PhysicalParty.fullName =\\\"Иванов Иван Иванович\\\" AND PhysicalParty.inn=110806179730\"}
```

В данном примере выполнено экранирование кавычек при помощи обратного слэша. Способ экранирования может отличаться, необходимо адаптировать для той утилиты, через которую выполняется запрос.

### *Пример 4. Запрос для поиска юрлица по наименованию, ИНН, ОГРН, КПП*

```
{"query": "\"? LegalParty.name=\\\"БИС КООП\\\" AND LegalParty.inn=2320059455 AND LegalParty.ogrn=1112366010701 AND LegalParty.kpp=500901001\"}
```

В данном примере выполнено экранирование кавычек при помощи обратного слэша. Способ экранирования может отличаться, необходимо адаптировать для той утилиты, через которую выполняется запрос.

### Пример 5. Ошибка авторизации

```
{
  "errorType" : "AccessDeniedException",
  "errorMessage" : "Error searching by param '? PhysicalParty.fullName =A*': Access is denied"
}
```

### Пример 6. Ошибка, когда найдено слишком много результатов.

Запрос:

```
{
  "query": "? PhysicalParty.fullName =A*"
}
```

Ответ:

```
{
  "errorType" : "IllegalArgumentException",
  "errorMessage" : "Error searching by param '? PhysicalParty.fullName =A*': Too many parties found (expected no more than 2000, got 25654374)"
}
```

### Пример 7. Ошибка, когда передан слишком длинный запрос

Запрос:

```
{
  "query": "Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз. Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз. Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз. Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз. Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз. Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз. Это поисковый запрос из 2160 символов. Этот запрос демонстрирует, какая вернется ошибка, если задать условие поискового запроса больше ограничения на его длину в 2000 символов. В самой фразе 240 символов, поэтому она будет повторена 9 раз."
}
```



```

{"party": [
  {
    "field": [
      {
        "value": "UNKNOWN",
        "name": "egrType"
      },
      {
        "value": "МАМОНОВ",
        "name": "surname"
      },
      {
        "value": "СЕРГЕЙ",
        "name": "name"
      },
      {
        "value": "ВИТАЛЬЕВИЧ",
        "name": "patronymic"
      },
      {
        "value": "18.03.1975",
        "name": "birthdate"
      },
      {
        "value": "",
        "name": "inn"
      },
      {
        "value": "",
        "name": "actualityDate"
      },
      {
        "value": "1558962657144",
        "name": "lastChangeTimestamp"
      }
    ],
    "hid": 17349,
    "sourceSystem": "AL",
    "rawId": "2415we444we232331",
    "type": "PHYSICAL"
  },
  {
    "field": [
      {
        "value": "NOT_FOUND",
        "name": "egrCheckStatus"
      },
      {
        "value": "ТОФ",
        "name": "name"
      },
      {
        "value": "ООО ТОФУ",
        "name": "nameRawSource"
      },
      {
        "value": "ООО",

```

```

    "name": "legalForm"
  },
  {
    "value": "Общество с ограниченной ответственностью",
    "name": "opfFull"
  },
  {
    "value": "7705974943",
    "name": "inn"
  },
  {
    "value": "22.04.2015",
    "name": "actualityDate"
  },
  {
    "value": "1531483288528",
    "name": "lastChangeTimestamp"
  }
],
"hid": 2050,
"sourceSystem": null,
"rawId": null,
"type": "LEGAL"
},
{
  "field": [
    {
      "value": "NOT_FOUND",
      "name": "egrCheckStatus"
    },
    {
      "value": "БЕЛЫЙ ЖУРАВЛЬ",
      "name": "name"
    },
    {
      "value": "ООО \"Белый журавль\"",
      "name": "nameRawSource"
    },
    {
      "value": "EDITED",
      "name": "nameQC"
    },
    {
      "value": "Белый журавль",
      "name": "shortName"
    },
    {
      "value": "ООО",
      "name": "legalForm"
    },
    {
      "value": "Общество с ограниченной ответственностью",
      "name": "opfFull"
    },
    {
      "value": "7705974943",
      "name": "inn"
    }
  ],

```

```

    {
      "value": "04.04.2016",
      "name": "actualityDate"
    },
    {
      "value": "1474527164019",
      "name": "lastChangeTimestamp"
    }
  ],
  "hid": 9215,
  "sourceSystem": null,
  "rawId": null,
  "type": "LEGAL"
},
{
  "field": [
    {
      "value": "NOT_FOUND",
      "name": "egrCheckStatus"
    },
    {
      "value": "Черный журавль",
      "name": "name"
    },
    {
      "value": "ООО \"Черный журавль\"",
      "name": "nameRawSource"
    },
    {
      "value": "EDITED",
      "name": "nameQC"
    },
    {
      "value": "Черный журавля",
      "name": "shortName"
    },
    {
      "value": "ООО",
      "name": "legalForm"
    },
    {
      "value": "Общество с ограниченной ответственностью",
      "name": "opfFull"
    },
    {
      "value": "7705974943",
      "name": "inn"
    },
    {
      "value": "04.04.2016",
      "name": "actualityDate"
    },
    {
      "value": "1515696668273",
      "name": "lastChangeTimestamp"
    }
  ],
  "hid": 196410,
  "sourceSystem": "AL",
  "rawId": "5",
  "type": "LEGAL"
}

```

»  
D

# PARTY.FUZZYSEARCH

Метод позволяет найти всех контрагентов, являющихся дубликатами для заданной записи.

- Описание

- Входные параметры

- Результирующие данные

- Особенности использования

- Дополнительная информация о контрагенте

- Исключительные ситуации

- Ограничения использования

- Примеры вызова PartyWS (SOAP)

- Пример 0. Пример запроса и ответа с возможными в них блоками

- Пример 1. Контрагенты найдены

- Пример 2. По заданным критериям ничего не найдено

- Пример 3. Найти дубликаты с ограничением максимального количество возвращаемых записей

- Пример 4. Найдено слишком много контрагентов

- Пример 5. Контрагенты найдены, возвращена полная информация по ним

- Пример 6. Использование связанных сущностей при поиске

- Пример 7. Поиск по ЕГРЮЛ

- Пример 8. Использование минимального коэффициента соответствия в поисковом запросе

- Пример 9. Использование набора правил в поисковом запросе

- Пример 10. Использование набора правил в поисковом запросе (несуществующее правило)

- Пример 11. Использование набора правил в поисковом запросе (для указанного в запросе правила не найдены дубликаты)

- Пример 12. Использование набора правил, минимального коэффициента соответствия и количества найденных дубликатов

- Пример 13. Использование блока include одновременно с флагом includePartyFields

- Пример 14. Использование блока include для проверки по ЧС

- Пример 15. В блоке matchedCounts отображены все дубликаты, в ответе отображены все дубликаты

- Пример 16. В блоке matchedCounts отображены все дубликаты, в ответе отображен один дубликат

- Пример 17. В блоке matchedCounts отображены все дубликаты, в ответе дубликатов нет

- Пример 18. Использование блока includeRequestData

- Пример 19. В запросе указан несуществующий тип связи

- Пример 20. В запросе указана некорректная сущность для одного конца связи



Пример 21. Для поиска по связанным сущностям используется слабая связь

Пример 1. Контрагенты найдены

Пример 2. Пример REST-запроса со связанными сущностями

Пример 3. Пример REST-запроса поиска по заданным правилам

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
party	WParty	да	Информация по контрагенту
relatedEssences	WParty[]	нет	Информация по связанным сущностям
maxCount	int	нет	Максимальное количество возвращаемых записей (не более 2000 — параметр <code>webservice.maxPartiesCount</code> )
minMatchScope	int	нет	Минимальный коэффициент соответствия, ниже которого не возвращать совпадения
matchRules/rule[]	int	нет	Набор правил, по которым интересен ответ
includePartyFields	boolean	нет	Возвращать реквизиты найденных контрагентов. По умолчанию <code>false</code> .  ⚠️ Параметр игнорируется при использовании блока <code>include/partyInfo[]</code>
include/partyInfo[]	Тип информации о контрагенте	нет	Дополнительная информация о контрагенте
includeRequestData/partyInfo[]	Тип информации об исходном контрагенте (из запроса)	нет	Стандартизированные данные, переданные в запросе

### Результирующие данные

Имя параметра	Тип	Описание
party	WMatchedParty[]	Информация по контрагенту (если явно не заданы параметр <code>include/partyInfo[]</code> , то не содержит множественные атрибуты — адреса, телефоны, документы и т.д.)
matchedCount s\guaranteed	int	Количество найденных гарантированных дубликатов  ⚠️ не включен по умолчанию, подключается дополнительно
matchedCount s\potential	int	Количество найденных негарантированных дубликатов  ⚠️ не включен по умолчанию, подключается дополнительно

## Особенности использования

⚠ Метод работает только при наличии меппинга очистки для соответствующей сущности

Метод работает по следующему алгоритму:

1. Произвести стандартизацию переданной записи без сохранения контрагента в системе.
2. Найти дубликаты между ним и всеми остальными контрагентами, хранящимися в системе.
3. В ответе вернуть всех найденных на шаге 2 дубликатов с указанием:
  - a. правила идентификации дубликатов, по которому был найден каждый из дубликатов;
  - b. коэффициента соответствия для правила, по которому контрагенты были идентифицированы как дубликаты;
  - c. количества гарантированных и негарантированных дубликатов.

Параметры `maxCount`, `minMatchScope`, `matchRules` могут влиять на количество отображаемых дубликатов в ответе, но не количество `matchedCounts\guaranteed` и `matchedCounts\potential`.

Оно показывает наличие дубликатов согласно параметрам запроса. (см пример 15, 16, 17). Подключается по запросу.

Если контрагент был объединен, то система вернет его `hid`, а также исходный идентификатор и `hid` для каждого исходного контрагента.

Возвращаемые методом контрагенты отсортированы в порядке убывания коэффициента соответствия.

Если на входе метода явно указано максимальное количество возвращаемых записей и количество найденных дубликатов превышает его, то система вернет то же количество, что указано в `maxCount` в порядке убывания коэффициента соответствия.

Если на входе метода явно указан минимальный коэффициент соответствия, ниже которого не возвращать совпадения, то система вернет только те дубликаты, коэффициент соответствия которых выше или равен указанному в `minMatchScope`. (см. Пример 8)

Если на входе метода явно указан набор правил, по которым интересен ответ, то система вернет дубли по указанным правилам. (см. Пример 9)

Если какое-то из правил, указанное в `rule`, не существует или по нему не найдены дубликаты, то вернется пустой ответ. (см. Пример 10-11)

Если на входе метода явно указан минимальный коэффициент соответствия, а также набор правил, и количество найденных дубликатов превышает `maxCount`, то система вернет список количество дублей, равное `maxCount`, с максимальным `minMatchScope`. (см. Пример 12)

Если параметр `includePartyFields` = **true**, то в ответе будут возвращены все реквизиты найденных контрагентов (см. Пример 5). Если задан параметр `//include`, параметр `includePartyFields` игнорируется.

## Дополнительная информация о контрагенте

Если задан параметр **//include**, то система вернет только те данные контрагента, которые в нем перечислены.

Значения параметра <b>//include /partyInfo</b>	Описание	Комментарий
REQUISITE	Информация о реквизитах	
ATTRIBUTE	Информация об атрибутах	Информация о соглашениях возвращается только если для параметра <b>web service.response.attributes.defaultValues</b> выставлено значение <b>true</b> .
SOURCE	Информация о привязке к исходным контрагентам	
RELATION	Информация о связях	В том числе связи с ЧС. Если связь подтверждаемая, в ответе также вернется её статус (примеры 0, 14)
PAST_ATTRIBUTE	Информация о прошлых значениях ФИО и документов	Возврат прошлых значений зависит только от заданного <b>include PAST_ATTRIBUTE</b> и не зависит от параметра <b>webservice.response.attributes.pastValues</b>  Если прошлые значения не найдены, блок полей в ответе возвращается пустым, ошибки нет  Если найдено более 100 прошлых значений, система вернёт ошибку

## Исключительные ситуации

Система вернет ошибку, если:

1. На входе метода не указано максимальное количество возвращаемых записей или оно превышает 2000 и по заданному контрагенту найдено более 2000 контрагентов-дубликатов (см. пример 4).
2. В запросе указан несуществующий тип связи (пример 19).
3. В запросе указана некорректная сущность для одного конца связи (пример 20).
4. Для поиска по связанным сущностям используется слабая связь (пример 21).

## Ограничения использования

На шаге 2 алгоритма поиск производится по состоянию клиентской базы на момент предыдущего поиска дубликатов.

## Примеры вызова PartyWS (SOAP)

### Пример 0. Пример запроса и ответа с возможными в них блоками

Запрос

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <!-- Может быть передана только одна сущность party -->
      <party type="PHYSICAL"> <!-- Достаточно указать только type -->

        <!-- Доступные поля описаны в модели данных контрагента. Могут
        быть переданы как все, для которых стоит отметка "можно прислать в soap", так и
        только часть из них. -->
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        ...
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateRawSource">03.12.1987</field>

        <!-- Атрибуты клиента. Передаются опционально, может быть
        передано несколько. Набор доступных атрибутов определяется типом контрагента -->
        <attribute type="PHONE" rawId="AL.2.PH.1"> <!-- Обязательно должен быть
        задан type, опционально rawId -->
          <!-- Доступные поля описаны в модели данных атрибута. Могут
          быть переданы как все, для которых стоит отметка "можно прислать в soap", так и
          только часть из них -->
          <field name="type">MOBILE</field>
          ...
          <field name="rawSource">9165345622</field>
        </attribute>
        ...
        <attribute type="DOCUMENT_PASSPORT" rawId="AL.2.DOC.1">
          <field name="qualifierType">PASSPORT_RU</field>
          <field name="documentSeries">4509</field>
          <field name="documentNumber">672184</field>
        </attribute>
      </party>

      <!-- Информация о связанных сущностях. Передается опционально -->
      <relatedEssences>
        <!-- Может быть передано несколько связанных сущностей -->
        <!-- Структура аналогична основной сущность party -->
        <party type="AGREEMENT">
          <field name="number">3935/045/0001685/01</field>
          ...
        </party>
      </relatedEssences>
    </maxCount>10</maxCount> <!--Опционально-->
  </body>
</Envelope>
```

```

    <includePartyFields>true</includePartyFields> <!-- Опционально. Устаревший
    параметр, не нужно использовать. Оставлен для обратной совместимости. Для новых
    интеграций следует использовать секцию <include>. -->
    <minMatchScore>85</minMatchScore> <!-- Коэффициенты соответствия
    текущих правил указаны в документации с перечнем правил идентификации
    дубликатов -->

    <!-- Опционально -->
    <matchRules>
        <!-- Может быть задано одно или несколько значений. Указываются
    идентификаторы правил (доступные значения указаны в документации с перечнем
    правил) -->
        <rule>3</rule>
        <rule>5</rule>
        ...
        <rule>10</rule>
    </matchRules>

    <!-- Опционально -->
    <include>
        <!-- Может быть задано одно или несколько значений -->
        <partyInfo>REQUISITE</partyInfo>
        <partyInfo>ATTRIBUTE</partyInfo>
        <partyInfo>SOURCE</partyInfo>
        <partyInfo>RELATION</partyInfo>
        <partyInfo>PAST_ATTRIBUTE</partyInfo>
    </include>

    </fuzzySearch>

</soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <!-- Может не вернуться в ответе, быть один контрагент или несколько. Зависит
    от количества найденных сущностей -->
      <matchedParty>
        <matchRule>3</matchRule>
        <matchScore>100</matchScore>

        <!-- Ниже показан максимальный набор допустимых полей в party.
    Может меняться в зависимости от переданных в запросе параметров -->
        <party type="PHYSICAL" hid="1026"> <!-- для 3К возвращается только
    type и hid, для одиночной также sourceSystem и rawId -->

        <!-- Возвращаемые поля описаны в модели данных контрагента.
    Возвращаются все поля с отметкой "возвращается в soap" -->
        <!-- Поля возвращаются только если в запросе в partyInfo есть
    значение REQUISITE -->
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">КУРКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        <field name="fullNameAuthor">AL:2</field>

```

```

<field name="fullNameQC">EDITED</field>
<field name="fullNameRawSource">Курко Олег Игоревич</field>
...
<field name="birthdate">12.12.1989</field>
<field name="birthdateAuthor">BT:3</field>
<field name="birthdateQC">EDITED</field>
<field name="birthdateRawSource">12.12.1989</field>

<!-- Атрибуты клиента. Возвращаются все атрибуты контрагента,
если есть -->
<!-- Поля возвращаются только если в запросе в partyInfo есть
значение ATTRIBUTE -->
<attribute type="PHONE" hid="1029" rawId="AL.2.PH.1">
<!-- Возвращаемые поля описаны в модели данных атрибута.
Возвращаются все поля с отметкой "возвращается в soap" -->
<field name="actualityDate"/>
<field name="type">MOBILE</field>
<field name="countryCode">7</field>
<field name="cityCode">916</field>
<field name="number">5345622</field>
...
<field name="author">AL:2</field>
<field name="qualityCode">GOOD</field>
<field name="rawSource">9165345622</field>
</attribute>
...
<attribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.1">
<field name="actualityDate"/>
<field name="qualifierType">PASSPORT_RU</field>
<field name="documentSeries">45 09</field>
<field name="documentNumber">672184</field>
<field name="issueDate"/>
<field name="issueAuthority"/>
<field name="departmentCode"/>
...
<field name="expiryDate"/>
<field name="author">AL:2</field>
<field name="qualityCode">EDITED</field>
<field name="rawSource">4509 672184</field>
</attribute>
<!-- Прошлые значения атрибутов клиента. Наличие прошлых
атрибутов в ответе регулируется настройкой в АРМ Администратора -->
<!-- Поля возвращаются только если в запросе в partyInfo есть
значение PAST_ATTRIBUTE -->
<!-- Возвращаются все прошлые значения атрибуты контрагента (если
есть) -->
<pastAttribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.
1">
<field name="actualityDate"/>
<field name="qualifierType">PASSPORT_RU</field>
<field name="documentSeries">45 08</field>
<field name="documentNumber">546767</field>
<field name="issueDate"/>
<field name="issueAuthority"/>
<field name="departmentCode"/>
...
<field name="expiryDate"/>
<field name="author">AL:2</field>
<field name="qualityCode">EDITED</field>
<field name="rawSource">45 08546767</field>
</pastAttribute>
...

```

```

1">
    <pastAttribute type="DOCUMENT_PASSPORT" hid="1031" rawId="AL.2.DOC.
        <field name="actualityDate"/>
        <field name="qualifierType">PASSPORT_RU</field>
        <field name="documentSeries">45 00</field>
        <field name="documentNumber">767453</field>
        <field name="issueDate"/>
        <field name="issueAuthority"/>
        <field name="departmentCode"/>
        ...
        <field name="expiryDate"/>
        <field name="author">AL:2</field>
        <field name="qualityCode">EDITED</field>
        <field name="rawSource">4500 767453</field>
    </pastAttribute>

    <!-- Исходные карточки. Возвращаются опционально только,
если результат сохранения золотая карточка. Может быть несколько -->
    <!-- Поля возвращаются только если в запросе в partyInfo есть
значение SOURCE -->
        <source hid="1024" sourceSystem="AL" rawId="2"/>
        ...
        <source hid="1025" sourceSystem="BT" rawId="3"/>

    <!-- Информация о связях контрагента (тип связи, информация о
связанной сущности), если есть. Может быть несколько. Блок содержит
фиксированный набор полей как в примере ниже. -->
    <!-- Поля возвращаются только если в запросе в partyInfo есть
значение RELATION -->
        <relation type="40" hashCode="
37179b491acded3ec9ec60736d7230f4">
            <second type="BLACKLIST" hid="1024"/>
            <field name="status">UNKNOWN</field> <!-- Если связь
подтверждаемая, то в ответе вернётся также статус подтверждения связи -->
        </relation>
        <relation type="1" hashCode="37179b491342ffec9ec60736d7230f4"
>
            <first type="PHYSICAL" hid="1025"/>
        </relation>

    </party>
</matchedParty>
...
<matchedParty>
    <matchRule>10</matchRule>
<matchScope>50</matchScope>
    <party type="PHYSICAL" hid="1028">
        ...
    </party>
</matchedParty>

</saveAndMergeResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Пример 1. Контрагенты найдены

Запрос:

### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ (Найдено три контрагента. Контрагент с HID=2026 — объединенная запись из двух исходных в ЕК с HID=2024 и HID=2025. Помимо идентификаторов в ЕК (HID) для каждого контрагента возвращаются его исходная система sourceSystem и идентификатор в исходной системе — rawId):

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearchResponse>
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
          </party>
        </matchedParty>
      <matchedParty>
        <matchRule>11</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="2026">
          <source hid="2024" sourceSystem="AL" rawId="11"/>
          <source hid="2025" sourceSystem="BT" rawId="11"/>
        </party>
        </matchedParty>
      <matchedParty>
        <matchRule>23</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="1026" sourceSystem="AL" rawId="5">
          </party>
        </matchedParty>
      </fuzzySearchResponse>
    </soapenv:Body>
  </soapenv:Envelope>
```

### Пример 2. По заданным критериям ничего не найдено

Запрос:

### Запрос



```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="surname">МАМОНОВСКИЙ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearchResponse/>
  </soapenv:Body>
</soapenv:Envelope>

```

### Пример 3. Найти дубликаты с ограничением максимального количество возвращаемых записей

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns=" http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;fuzzySearch&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="surname"&gt;МАМОНОВ&lt;/field&gt;         &lt;field name="name"&gt;СЕРГЕЙ&lt;/field&gt;         &lt;field name="patronymic"&gt;ВИКТОРОВИЧ&lt;/field&gt;         &lt;field name="fullNameRawSource"&gt;Иванов Сергей Викторович&lt;/field&gt;         &lt;field name="birthdate"&gt;03.12.1987&lt;/field&gt;         &lt;field name="birthdateRawSource"&gt;03.12.1987&lt;/field&gt;       &lt;/party&gt;       &lt;maxCount&gt;2&lt;/maxCount&gt;     &lt;/fuzzySearch&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

ОТВЕТ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearchResponse>

```

```

<matchedParty>
  <matchRule>1</matchRule>
  <matchScope>100</matchScope>
  <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
</party>
</matchedParty>
<matchedParty>
  <matchRule>11</matchRule>
  <matchScope>100</matchScope>
  <party type="PHYSICAL" hid="2026">
    <source hid="2024" sourceSystem="AL" rawId="11"/>
    <source hid="2025" sourceSystem="BT" rawId="11"/>
  </party>
</matchedParty>
</fuzzySearchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

#### Пример 4. Найдено слишком много контрагентов

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;fuzzySearch&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="surname"&gt;ИВАНОВ&lt;/field&gt;         &lt;field name="name"&gt;ИВАН&lt;/field&gt;         &lt;field name="patronymic"&gt;ИВАНОВИЧ&lt;/field&gt;         &lt;field name="fullNameRawSource"&gt;ИВАНОВ ИВАН ИВАНОВИЧ&lt;/field&gt;         &lt;field name="birthdate"&gt;03.12.1987&lt;/field&gt;         &lt;field name="birthdateRawSource"&gt;03.12.1987&lt;/field&gt;       &lt;/party&gt;     &lt;/fuzzySearch&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ:

<pre> &lt;soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;soap:Body&gt;     &lt;soap:Fault&gt;       &lt;faultcode&gt;soap:Server&lt;/faultcode&gt;       &lt;faultstring&gt;Too many parties found (expected no more than 2000, got 3124)&lt; /faultstring&gt;     &lt;/soap:Fault&gt;   &lt;/soap:Body&gt; &lt;/soap:Envelope&gt; </pre>
---

#### Пример 5. Контрагенты найдены, возвращена полная информация по ним

Запрос:

Запрос
--------

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
      <includePartyFields>true</includePartyFields>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ (среди найденных контрагентов есть объединенные):

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearchResponse>
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
          <field name="actualityDate">22.02.2012</field>
          <field name="surname">МАМОНОВ</field>
          <field name="name">СЕРГЕЙ</field>
          <field name="patronymic">ВИКТОРОВИЧ</field>
          <field name="fullNameAuthor">AL:1</field>
          <field name="fullNameQC">EDITED</field>
          <field name="fullNameRawSource">Мамонов Сергей Викторович</field>
          <field name="birthdate">03.12.1987</field>
          <field name="birthdateAuthor">AL:1</field>
          <field name="birthdateQC">EDITED</field>
          <field name="birthdateRawSource">03.12.1987</field>
        </party>
      </matchedParty>
      <matchedParty>
        <matchRule>11</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="2026">
          <field name="actualityDate">22.02.2012</field>
          <field name="surname">МАМОНОВ</field>
          <field name="name">СЕРГЕЙ</field>
          <field name="patronymic">ВИКТОРОВИЧ</field>
          <field name="fullNameAuthor">AL:2</field>
          <field name="fullNameQC">EDITED</field>
          <field name="fullNameRawSource">Курко Олег Игоревич</field>
          <field name="birthdate">12.12.1989</field>
          <field name="birthdateAuthor">BT:3</field>
          <field name="birthdateQC">EDITED</field>
          <field name="birthdateRawSource">12.12.1989</field>
          <field name="maritalStatus">SINGLE_MALE</field>
          <field name="maritalStatusAuthor">AL:2</field>
          <source hid="2024" sourceSystem="AL" rawId="2"/>
        </party>
        <party type="PHYSICAL" hid="2025" sourceSystem="BT" rawId="3"/>

```

```

    </party>
  </matchedParty>
</matchedParty>
  <matchRule>23</matchRule>
  <matchScope>60</matchScope>
    <party type="PHYSICAL" hid="1026" sourceSystem="AL" rawId="5">
      <field name="actualityDate">22.02.2012</field>
      <field name="surname">МАМОНТОВ</field>
      <field name="name">СЕРГЕЙ</field>
      <field name="patronymic">ВИКТОРОВИЧ</field>
      <field name="fullNameAuthor">AL:5</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameRawSource">Мамонтов Сергей Викторович</field>
      <field name="birthdate">03.12.1987</field>
      <field name="birthdateAuthor">AL:5</field>
      <field name="birthdateQC">EDITED</field>
      <field name="birthdateRawSource">03.12.1987</field>
    </party>
  </matchedParty>
</fuzzySearchResponse>
</soapenv:Body>
</soapenv:Envelope>

```

### Пример 6. Использование связанных сущностей при поиске

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;fuzzySearch&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="fullNameRawSource"&gt;Алешин Владимир Владимирович&lt;/field&gt;         &lt;field name="birthdate"&gt;13.05.1977&lt;/field&gt;       &lt;/party&gt;       &lt;relatedEssences&gt;         &lt;party type="AGREEMENT"&gt;           &lt;field name="number"&gt;3935/045/0001685/01&lt;/field&gt;           &lt;relation type="102"/&gt;         &lt;/party&gt;       &lt;/relatedEssences&gt;     &lt;/fuzzySearch&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ:

<pre> &lt;fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;matchedParty&gt;     &lt;matchRule&gt;30&lt;/matchRule&gt;     &lt;matchScope&gt;95&lt;/matchScope&gt;     &lt;party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"/&gt;   &lt;/matchedParty&gt; &lt;/fuzzySearchResponse&gt; </pre>
---

## Пример 7. Поиск по ЕГРЮЛ

Запрос (модуль ЕГРЮЛ подключается отдельно):

```


Запрос


<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:_13="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
      <party type="EGR">
        <field name="inn">6145009520</field>
        <field name="shortName">ООО СТРОЙДОН</field>
        <attribute type="ADDRESS">
          <field name="rawSource">Донецк, мкр 3, 19</field>
        </attribute>
      </party>
      <includePartyFields>true</includePartyFields>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>5</matchRule>
        <matchScope>97</matchScope>
        <party type="EGR" hid="576364" sourceSystem="FTS_LG" rawId="1076145000315">
          <field name="inn">6145009520</field>
          <field name="innAuthor">FTS_LG:1076145000315</field>
          <field name="ogrn">1076145000315</field>
          <field name="ogrnAuthor">FTS_LG:1076145000315</field>
          <field name="kpp">614501001</field>
          <field name="kppAuthor">FTS_LG:1076145000315</field>
          <field name="egrType">LG</field>
          <field name="egrTypeAuthor">FTS_LG:1076145000315</field>
          <field name="registrationDate">31.08.2007</field>
          <field name="registrationDateAuthor">FTS_LG:1076145000315</field>
          <field name="liquidationDate">22.11.2010</field>
          <field name="liquidationDateAuthor">FTS_LG:1076145000315</field>
          <field name="activityStatus">LIQUIDATED</field>
          <field name="activityStatusAuthor">FTS_LG:1076145000315</field>
          <field name="name">ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
"СТРОЙДОН"</field>
          <field name="nameWoOpf">СТРОЙДОН</field>
          <field name="shortName">ООО "СТРОЙДОН"</field>
          <field name="shortNameWoOpf">СТРОЙДОН</field>
          <field name="nameAuthor">FTS_LG:1076145000315</field>
          <field name="latinName"/>
          <field name="latinNameAuthor">FTS_LG:1076145000315</field>
          <field name="opf">Общество с ограниченной ответственностью</field>
          <field name="opfShort">ООО</field>
          <field name="opfCode">12300</field>
          <field name="opfAuthor">FTS_LG:1076145000315</field>
          <field name="okved">74.20</field>
          <field name="okvedAuthor">FTS_LG:1076145000315</field>
        </party>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>
```

```

    <field name="branchType">MAIN</field>
    <field name="branchTypeAuthor">FTS_LG:1076145000315</field>
    <field name="managementName">Котляров Евгений Михайлович</field>
    <field name="managementPost">Руководитель юридического лица</field>
    <field name="managementNameAuthor">FTS_LG:1076145000315</field>
    <field name="actualityDateAuthor">FTS_LG:1076145000315</field>
    <field name="actualityDate">05.01.2016</field>
  </party>
</matchedParty>
</fuzzySearchResponse>
</soap:Body>
</soap:Envelope>

```

### Пример 8. Использование минимального коэффициента соответствия в поисковом запросе

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;fuzzySearch&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="name"&gt;Анган&lt;/field&gt;         &lt;field name="surname"&gt;Онкуежий&lt;/field&gt;         &lt;field name="birthdate"&gt;01.11.2001&lt;/field&gt;       &lt;/party&gt;       &lt;minMatchScope&gt;60&lt;/minMatchScope&gt;     &lt;/fuzzySearch&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2851">
          <source hid="1442" sourceSystem="AL" rawId="20180323074401"/>
          <source hid="2850" sourceSystem="AL" rawId="201808174235"/>
          <source hid="2855" sourceSystem="AL" rawId="201808174236"/>
          <source hid="2857" sourceSystem="AL" rawId="201808174237"/>
        </party>
      </matchedParty>
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2884" sourceSystem="GM" rawId="564"/>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>

```

### Пример 9. Использование набора правил в поисковом запросе

### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="name">Иван</field>
        <field name="patronymic">Игоревич</field>
        <field name="surname">Куков</field>
        <field name="inn">111111110000</field>
        <field name="birthdate">20.11.1987</field>
      </party>
      <matchRules>
        <rule>32</rule>
        <rule>46</rule>
      </matchRules>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>32</matchRule>
        <matchScope>88</matchScope>
        <party type="PHYSICAL" hid="2875">
          <source hid="2873" sourceSystem="AL" rawId="1053"/>
          <source hid="2874" sourceSystem="BT" rawId="1053"/>
        </party>
      </matchedParty>
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2881" sourceSystem="GM" rawId="101223"/>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>
```

### Пример 10. Использование набора правил в поисковом запросе (несуществующее правило)

### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="name">Иван</field>
        <field name="patronymic">Игоревич</field>
        <field name="surname">Куков</field>
      </party>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <field name="inn">111111110000</field>
    <field name="birthdate">20.11.1987</field>
  </party>
  <matchRules>
    <rule>0</rule>
  </matchRules>
</fuzzySearch>
</soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13"/>
  </soap:Body>
</soap:Envelope>

```

*Пример 11. Использование набора правил в поисковом запросе (для указанного в запросе правила не найдены дубликаты)*

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;fuzzySearch&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="name"&gt;Антон&lt;/field&gt;         &lt;field name="patronymic"&gt;Олегович&lt;/field&gt;         &lt;field name="surname"&gt;Проверялкин&lt;/field&gt;         &lt;field name="birthdate"&gt;01.11.2001&lt;/field&gt;       &lt;/party&gt;       &lt;matchRules&gt;         &lt;rule&gt;32&lt;/rule&gt;       &lt;/matchRules&gt;     &lt;/fuzzySearch&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

ОТВЕТ

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13"/>
  </soap:Body>
</soap:Envelope>

```

*Пример 12. Использование набора правил, минимального коэффициента соответствия и количества найденных дубликатов*

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:_13="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt; </pre>



```

<soapenv:Body>
  <fuzzySearch>
    <party type="PHYSICAL">
      <field name="name">Полезен</field>
      <field name="surname">Овес</field>
      <field name="birthdate">01.03.1964</field>
    </party>
    <maxCount>1</maxCount>
    <minMatchScope>60</minMatchScope>
    <matchRules>
      <rule>46</rule>
    </matchRules>
  </fuzzySearch>
</soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>46</matchRule>
        <matchScope>60</matchScope>
        <party type="PHYSICAL" hid="2752">
          <source hid="1628" sourceSystem="AL" rawId="20170925084201"/>
          <source hid="1644" sourceSystem="AL" rawId="20170925084217"/>
        </party>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>

```

### Пример 13. Использование блока *include* одновременно с флагом *includePartyFields*

Запрос

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;fuzzySearch&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="name"&gt;Иван&lt;/field&gt;         &lt;field name="patronymic"&gt;Игоревич&lt;/field&gt;         &lt;field name="surname"&gt;Куков&lt;/field&gt;         &lt;field name="inn"&gt;111111110000&lt;/field&gt;         &lt;field name="birthdate"&gt;20.11.1987&lt;/field&gt;       &lt;/party&gt;       &lt;includePartyFields&gt;true&lt;/includePartyFields&gt;       &lt;include&gt;         &lt;partyInfo&gt;RELATION&lt;/partyInfo&gt;       &lt;/include&gt;     &lt;/fuzzySearch&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ (вернем связи, без информации о реквизитах КА)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="14">
          <relation type="8" hashCode="37179b491acded3ec9ec60736d7230f4">
            <second type="PHYSICAL" hid="1024"/>
          </relation>
        </party>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>
```

### Пример 14. Использование блока include для проверки по ЧС

Запрос

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <fuzzySearch>
      <party type="PHYSICAL">
        <field name="name">Равиль</field>
        <field name="patronymic">Хусьяинович</field>
        <field name="surname">Абдуллоев</field>
        <field name="birthdate">03.06.1959</field>
      </party>
      <include>
        <partyInfo>RELATION</partyInfo>
      </include>
    </fuzzySearch>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ (вернем связи с найденным черносписочником, связь подтверждаемая, поэтому вернулся её статус)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="PHYSICAL" hid="1025" sourceSystem="AL" rawId="14">
          <relation type="40" hashCode="37179b491acded3ec9ec60736d7230f4">
            <second type="BLACKLIST" hid="1024"/>
            <field name="status">UNKNOWN</field>
          </relation>
        </party>
      </matchedParty>
    </fuzzySearchResponse>
  </soap:Body>
</soap:Envelope>
```

```
</soap:Body>
</soap:Envelope>
```

*Пример 15. В блоке `matchedCounts` отображены все дубликаты, в ответе отображены все дубликаты*

Запрос

```
Запрос
<?xml version="1.0" encoding="UTF-8"?>
<fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party type="PHYSICAL">
    <field name="fullNameRawSource">Морозов Илья Ильич</field>
    <field name="birthdate">21.09.1980</field>
    <field name="birthdateRawSource">21.09.1980</field>
    <attribute type="ADDRESS">
      <field name="type">HOME</field>
      <field name="rawSource">191014,Санкт-Петербург, ул Жуковского, д. 29</field>
    </attribute>
  </party>
</fuzzySearch>
```

Ответ с количеством дубликатов

```
<fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
  <matchedCounts>
    <guaranteed>1</guaranteed>
    <potential>2</potential>
  </matchedCounts>
  <matchedParty>
    <matchRule>10</matchRule>
    <matchScope>90</matchScope>
    <party type="PHYSICAL" hid="5" sourceSystem="AL" rawId="5">
      <source hid="5" sourceSystem="AL" rawId="5"/>
    </party>
  </matchedParty>
  <matchedParty>
    <matchRule>8</matchRule>
    <matchScope>56</matchScope>
    <party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
      <source hid="1" sourceSystem="AL" rawId="1"/>
    </party>
  </matchedParty>
  <matchedParty>
    <matchRule>8</matchRule>
    <matchScope>56</matchScope>
    <party type="PHYSICAL" hid="7" sourceSystem="AL" rawId="7">
      <source hid="7" sourceSystem="AL" rawId="7"/>
    </party>
  </matchedParty>
</fuzzySearchResponse>
```

*Пример 16. В блоке `matchedCounts` отображены все дубликаты, в ответе отображен один дубликат*

Запрос

#### Запрос

```
<?xml version="1.0" encoding="UTF-8"?>
<fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party type="PHYSICAL">
    <field name="fullNameRawSource">Морозов Илья Ильич</field>
    <field name="birthdate">21.09.1980</field>
    <field name="birthdateRawSource">21.09.1980</field>
    <attribute type="ADDRESS">
      <field name="type">HOME</field>
      <field name="rawSource">191014,Санкт-Петербург, ул Жуковского, д. 29</field>
    </attribute>
  </party>
  <minMatchScope>90</minMatchScope>
</fuzzySearch>
```

Ответ (количество отображаемых дублей отрегулировано параметром matchScope)

```
<fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
  <matchedCounts>
    <guaranteed>1</guaranteed>
    <potential>2</potential>
  </matchedCounts>
  <matchedParty>
    <matchRule>10</matchRule>
    <matchScope>90</matchScope>
    <party type="PHYSICAL" hid="5" sourceSystem="AL" rawId="5">
      <source hid="5" sourceSystem="AL" rawId="5"/>
    </party>
  </matchedParty>
</fuzzySearchResponse>
```

*Пример 17. В блоке matchedCounts отображены все дубликаты, в ответе дубликатов нет*

Запрос

#### Запрос

```
<?xml version="1.0" encoding="UTF-8"?>
<fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party type="PHYSICAL">
    <field name="fullNameRawSource">Морозов Илья Ильич</field>
    <field name="birthdate">21.09.1980</field>
    <field name="birthdateRawSource">21.09.1980</field>
    <attribute type="ADDRESS">
      <field name="type">HOME</field>
      <field name="rawSource">191014,Санкт-Петербург, ул Жуковского, д. 29</field>
    </attribute>
  </party>
  <matchRules>
    <rule>11</rule>
  </matchRules>
</fuzzySearch>
```

Ответ (количество отображаемых дублей отрегулировано параметром matchRules)

```
<fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
  <matchedCounts>
```

```

    <guaranteed>1</guaranteed>
    <potential>2</potential>
  </matchedCounts>
</fuzzySearchResponse>

```

## Пример 18. Использование блока `includeRequestData`

### Запрос

```

<?xml version="1.0" encoding="UTF-8"?> <?xml version="1.0" encoding="UTF-8"?>
<fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party type="PHYSICAL">
    <field name="fullNameRawSource">Алешин Владимир Владимирович</field>
    <field name="birthdate">13.05.1977</field>
    <attribute type="DOCUMENT_PASSPORT">
      <field name="type">PASSPORT</field>
      <field name="actualityDate">01.02.2014</field>
      <field name="additionalInfo">инфо 1</field>
      <field name="departmentCode">660-119</field>
      <field name="documentNSReasons">SERIES_CODE_CLASH</field>
      <field name="documentNumber">654343</field>
      <field name="documentSeries">45 07</field>
      <field name="expiryDate">03.09.2016</field>
      <field name="issueAuthority">отделение Московское</field>
      <field name="issueDate">03.08.2007</field>
      <field name="primary">>true</field>
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="rawSource">45 07•654343•03.08.2007•отделение Московское•660-
119•03.09.2016</field>
    </attribute>
    <relation type="100">
      <second type="AGREEMENT"/>
    </relation>
  </party>
  <relatedEssences>
    <party type="AGREEMENT">
      <field name="number">111111-01</field>
      <relation type="100"/>
    </party>
  </relatedEssences>
  <includeRequestData>
    <partyInfo>REQUISITE</partyInfo>
    <partyInfo>ATTRIBUTE</partyInfo>
    <partyInfo>RELATION</partyInfo>
  </includeRequestData>
</fuzzySearch>

```

Ответ: дубликаты не найдены, но вернулись стандартизированные данные из запроса.

```

<fuzzySearchResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party hid="-1" type="PHYSICAL">
    <field name="actualityDate"/>
    <field name="actualityDateAuthor">null:null</field>
    <field name="birthdateAuthor">null:null</field>
    <field name="birthdateQC">EDITED</field>
    <field name="birthdateRawSource">13.05.1977</field>
    <field name="birthPlace"/>
    <field name="birthPlaceAuthor">null:null</field>
    <field name="blacklistCheckStatus">NOT_FOUND</field>
    <field name="branch"/>
  </party>

```

```

<field name="branchAuthor">null:null</field>
<field name="crossId"/>
<field name="crossIdAuthor">null:null</field>
<field name="dependantsCount"/>
<field name="dependantsCountAuthor">null:null</field>
<field name="education">UNKNOWN</field>
<field name="educationAuthor">null:null</field>
<field name="fullNameAuthor">null:null</field>
<field name="fullNameQC">EDITED</field>
<field name="fullNameRawSource">Алешин Владимир Владимирович</field>
<field name="inn"/>
<field name="innAuthor">null:null</field>
<field name="innQC">EMPTY</field>
<field name="innRawSource"/>
<field name="job"/>
<field name="jobAuthor">null:null</field>
<field name="jobStartDate"/>
<field name="jobStartDateAuthor">null:null</field>
<field name="jobType"/>
<field name="jobTypeAuthor">null:null</field>
<field name="maritalStatus">UNKNOWN</field>
<field name="maritalStatusAuthor">null:null</field>
<field name="marketingCategory">UNKNOWN</field>
<field name="marketingCategoryAuthor">null:null</field>
<field name="monthlyIncome"/>
<field name="monthlyIncomeAuthor">null:null</field>
<field name="organization"/>
<field name="organizationAuthor">null:null</field>
<field name="organizationType"/>
<field name="organizationTypeAuthor">null:null</field>
<field name="previousFullNameAuthor">null:null</field>
<field name="previousFullNameQC">EDITED</field>
<field name="previousFullNameRawSource"/>
<field name="previousName"/>
<field name="previousPatronymic"/>
<field name="previousSurname"/>
<field name="residence">UNKNOWN</field>
<field name="residenceAuthor">null:null</field>
<field name="snils"/>
<field name="snilsAuthor">null:null</field>
<field name="snilsQC">EMPTY</field>
<field name="snilsRawSource"/>
<field name="sourceChannel">UNKNOWN</field>
<field name="sourceChannelAuthor">null:null</field>
<attribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1">
  <field name="actualityDate">01.02.2014</field>
  <field name="additionalInfo">инфо 1</field>
  <field name="author">null:null</field>
  <field name="departmentCode">660-119</field>
  <field name="documentNSReasons">SERIES_CODE_CLASH</field>
  <field name="documentNumber">654343</field>
  <field name="documentSeries">45 07</field>
  <field name="expiryDate">03.09.2016</field>
  <field name="invalidStatus">INVALID</field>
  <field name="issueAuthority">отделение Московское</field>
  <field name="issueDate">03.08.2007</field>
  <field name="popularity">NONE</field>
  <field name="primary">>true</field>
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="qualityCode">CODE_SERIES_CLASH</field>
  <field name="rawSource">45 07•654343•03.08.2007•отделение Московское•660-
119•03.09.2016</field>

```

```

    <field name="type">PASSPORT</field>
  </attribute>
  <relation type="100">
    <second type="AGREEMENT"/>
  </relation>
</party>
<relatedEssences>
  <party type="AGREEMENT">
    <field name="actualityDate"/>
    <field name="actualityDateAuthor">null:null</field>
    <field name="dateAuthor">null:null</field>
    <field name="description"/>
    <field name="descriptionAuthor">null:null</field>
    <field name="number">111111-01</field>
    <field name="numberAuthor">null:null</field>
  </party>
</relatedEssences>
</fuzzySearchResponse>

```

*Пример 19. В запросе указан несуществующий тип связи*

Запрос:

Запрос
<pre> &lt;?xml version="1.0" encoding="UTF-8"?&gt; &lt;fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;party type="PHYSICAL"&gt;     &lt;field name="fullNameRawSource"&gt;Морозов Илья Ильич&lt;/field&gt;     &lt;field name="birthdate"&gt;21.09.1980&lt;/field&gt;     &lt;field name="birthdateRawSource"&gt;21.09.1980&lt;/field&gt;   &lt;/party&gt;   &lt;relatedEssences&gt;     &lt;party type="LEGAL"&gt;       &lt;field name="name"&gt;ТРУЛЯЛЯ&lt;/field&gt;       &lt;relation type="100500"/&gt;     &lt;/party&gt;   &lt;/relatedEssences&gt; &lt;/fuzzySearch&gt; </pre>

Ответ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:INVALID_REQUEST<
/faultcode>
      <faultstring>'RelationType' with ID '100500' not found</faultstring>
      <detail>
        <errorMessage xmlns="http://hflabs.ru/cdi/soap/2_13">
          <errorType>IllegalArgumentException</errorType>
          <errorMessage>Searching error: 'RelationType' with ID '100500' not found<
/errorMessage>
        </errorMessage>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

*Пример 20. В запросе указана некорректная сущность для одного конца связи*

Запрос:

```


Запрос



```
<?xml version="1.0" encoding="UTF-8"?>
<fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party type="PHYSICAL">
    <field name="fullNameRawSource">Морозов Илья Ильич</field>
    <field name="birthdate">21.09.1980</field>
    <field name="birthdateRawSource">21.09.1980</field>
  </party>
  <relatedEssences>
    <party type="PHYSICAL">
      <field name="name">ТРУЛЯЛЯ</field>
      <relation type="11"/>
    </party>
  </relatedEssences>
</fuzzySearch>
```


```

Ответ:

```


```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:INVALID_REQUEST<
/faultcode>
      <faultstring>Incorrect second link type PHYSICAL for relation 11</faultstring>
      <detail>
        <errorMessage xmlns="http://hflabs.ru/cdi/soap/2_13">
          <errorType>IllegalArgumentException</errorType>
          <errorMessage>Searching error: Incorrect second link type PHYSICAL for relation
11</errorMessage>
        </errorMessage>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```


```

*Пример 21. Для поиска по связанным сущностям используется слабая связь*

Запрос:

```


Запрос



```
<?xml version="1.0" encoding="UTF-8"?>
<fuzzySearch xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party type="PHYSICAL">
    <field name="fullNameRawSource">Морозов Илья Ильич</field>
    <field name="birthdate">21.09.1980</field>
    <field name="birthdateRawSource">21.09.1980</field>
  </party>
  <relatedEssences>
    <party type="LEGAL">
      <field name="name">ТРУЛЯЛЯ</field>
      <relation type="11"/>
    </party>
  </relatedEssences>
</fuzzySearch>
```


```



```
</party>
</relatedEssences>
</fuzzySearch>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:INVALID_REQUEST<
/faultcode>
      <faultstring>Incorrect second link type PHYSICAL for relation 11</faultstring>
      <detail>
        <errorMessage xmlns="http://hflabs.ru/cdi/soap/2_13">
          <errorType>IllegalArgumentException</errorType>
          <errorMessage>Searching error: Relation 11 must be strong</errorMessage>
        </errorMessage>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Контрагенты найдены

Запрос:

```
{ "includePartyFields": "false",
  "party": {
    "field": [{
      "value": "24.06.2013",
      "name": "birthdateRawSource"
    }, {"value": "Иванов Иван Сергеевич",
      "name": "fullNameRawSource"}],
    "sourceSystem": "AL",
    "rawId": "1",
    "attribute": [ {
      "field": [ {
        "value": "Россия, Санкт-Петербург",
        "name": "rawSource"
      }, {
        "value": "false",
        "name": "primary"
      }, {
        "value": "23.02.2012",
        "name": "actualityDate"
      } ],
      "type": "ADDRESS"
    } ],
    "type": "PHYSICAL"
  }
}
```

ОТВЕТ:

```

{"matchedParty": [{
  "matchRule": "21",
  "matchScope": 60,
  "party": {
    "field": [],
    "hid": 1,
    "sourceSystem": "BT",
    "rawId": "1",
    "attribute": [],
    "source": [],
    "relation": [],
    "type": "PHYSICAL"
  }
}]

```

## Пример 2. Пример REST-запроса со связанными сущностями

Запрос:

```

{
  "party": {
    "field": [
      {
        "name": "fullNameRawSource",
        "value": "Рожкин Владимир Николаевич"
      }
    ],
    "type": "PHYSICAL"
  },
  "relatedEssences": [
    {
      "party": [
        {
          "field": [
            {
              "name": "contractNumber",
              "value": "0338893"
            }
          ],
          {
            "name": "contractSeries",
            "value": "5915"
          }
        ],
        "relation": [
          {
            "type": "101"
          }
        ],
        "type": "CONTRACT"
      }
    ]
  }
}

```

Ответ:

```

{"matchedParty": [
  {
    "matchRule": "21",
    "matchScope": 60,
    "party": {
      "field": [],
      "hid": 1,

```

```

    "sourceSystem": "BT",
    "rawId": "1",
    "attribute": [],
    "source": [],
    "relation": [],
    "type": "PHYSICAL" } ]}]

```

### Пример 3. Пример REST-запроса поиска по заданным правилам

Запрос:

```

{ "minMatchScope": "500",
  "matchRules":{
    "rule":["4", "21", "41", "42", "162", "72", "73", "54"]
  },
  "include":{
    "partyInfo" : ["ATTRIBUTE","REQUISITE","RELATION", "SOURCE", "PAST_ATTRIBUTE"]
  },
  "party" :
  {
    "field" :
    [
      {
        "value" : "24.06.2013",
        "name" : "birthdateRawSource"
      },
      {
        "value" : "123456789011",
        "name" : "inn"
      },
      {
        "value" : "Петров Максим Игоревич",
        "name" : "fullNameRawSource"
      }
    ],
    "attribute" : [
      {
        "field" : [
          {
            "value" : "198323, Россия, г. Санкт-Петербург, ул. Советская (тер Горелово), д. 123, корп.1. стр. 2",
            "name" : "rawSource"
          },
          {
            "value" : "REGISTRATION",
            "name" : "type"
          }
        ],
        "type" : "ADDRESS"
      },
      {
        "field" : [
          {
            "value" : "798111111",
            "name" : "rawSource"
          },
          {
            "value" : "MOBILE",
            "name" : "type"
          }
        ],

```

```

    "type" : "PHONE"
  }
  {
    "field" : [
      {
        "value" : "PASSPORT_RU",
        "name" : "qualifierType"
      },
      {
        "value" : "13.02.2021",
        "name" : "issueDate"
      },
      {
        "value" : "4508",
        "name" : "documentSeries"
      },
      {
        "value" : "546767",
        "name" : "documentNumber"
      }
    ],
    "type" : "DOCUMENT_PASSPORT"
  }
  ],
  "type" : "PHYSICAL"
}
}

```

ОТВЕТ:

```

{ "matchedCounts": null,
  "matchedParty": [
    {
      "matchRule": "41",
      "matchScope": 900,
      "party": {
        "field": [],
        "hid": 1,
        "sourceSystem": "BT",
        "rawId": "1",
        "attribute": [],
        "source": [],
        "relation": [],
        "type": "PHYSICAL"
      }
    },
    {
      "matchRule": "42",
      "matchScope": 880,
      "party": {
        "field": [],
        "hid": 1,
        "sourceSystem": "BT",
        "rawId": "1",
        "attribute": [],
        "source": [],
        "relation": [],
        "type": "PHYSICAL"
      }
    }
  ]
}

```



# PARTY.FINDINFNS

Метод позволяет получить ИНН от ФНС по набору ПДн без сохранения карточки в CDI ЕК (кредитный конвейер, скоринг и т.д.).

Работает только если подключен модуль актуализации ИНН физлиц.

- **Описание**

  - Входные параметры

  - Результирующие данные

  - Логика обработки запроса

  - Используемые методы API ФНС

  - Особенности использования

  - Исключительные ситуации

- **Пример вызова PartyWS (SOAP)**

  - Пример 1. ИНН найден в ФНС

  - Пример 2. ИНН не найден в ФНС

  - Пример 2. Модуль актуализации ИНН физлиц не подключен, ошибка

- **Пример вызова PartyRA (JSON)**

  - Пример 1. ИНН найден в ФНС

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
surname	String(60)	Да	Фамилия
name	String(60)	Да	Имя
patronymic	String(60)	Нет	Отчество
birthdate	String(10)	Да	Дата рождения. Формат уууу-mm-dd
documentCode	String(50)	Да	Код ДУЛ (по СПДУЛ) или тип документа ФЛ в ЕК
documentSeries	String(30)	Нет	Серия документа
documentNumber	String(30)	Да	Номер документа

### Результирующие данные

Имя параметра	Тип	Обязательный?	Описание тега
found	boolean	Да	Найден в ФНС

Имя параметра	Тип	Обязательный?	Описание тега
inn	String(12)	Заполняется при found = true	ИНН
fnsErrorCode	String(255)	Нет	Код ошибки ФНС
fnsErrorMessage	String(255)	Нет	Описание ошибки ФНС

## Логика обработки запроса

1. Провалидировать запрос на обязательность заполнения и ФЛК:
  - a. Если в запросе не хватает обязательных полей — свалиться с ошибкой, не дергая ФНС.
  - b. Если запрос не проходит ФЛК — свалиться с ошибкой, не дергая ФНС.
2. Стандартизировать данные, переданные в запросе, рассчитать от них хэш.  
Если после стандартизации какие-то данные были опустошены (неудовлетворительное качество данных), то использовать для них исходные данные (которые переданы на вход до стандартизации), рассчитать хэш от них.
3. Проверить наличие хэша в таблице запросов/ответов.
  - a. Если найден - сформировать ответ по данным таблицы
  - b. Если не найден - сформировать SINGLE-запрос в ФНС, обработать ответ, сохранить в кэш.  
В запросе передать стандартизированные данные. Если после стандартизации какие-то данные были опустошены (неудовлетворительное качество данных), то использовать для них исходные данные (которые переданы на вход до стандартизации).
4. Вернуть ответ (ОК или ошибка)

## Используемые методы API ФНС

При работе метода **findInFns** вызываются следующие POST-методы API ФНС:

- Получение токена доступа (accessToken)\*: `/auth/v1/token`
- Проверка одиночного ИНН: `/ion/v1/inn`  
В заголовке REST запроса указывается токен доступа (accessToken)

\*Запрос токена доступа происходит по необходимости: он кешируется и если время его жизни не истекло, то метод `/auth/v1/token` не вызывается. Даты начала и окончания действия токена доступа (accessTokenStartDate, accessTokenEndDate) приходят в ответе ФНС вместе токеном.

## Особенности использования

1. Работа метода - синхронная.
2. Производительность метода API напрямую зависит и ограничена пропускной способностью API ФНС. Максимальное число обращений к методам API ФНС - 50 000 раз в сутки. Количество запросов соответствует количеству обращений к каждому методу вне зависимости от содержания бизнес сообщения.

## Исключительные ситуации

Если атрибут с указанным НID и типом отсутствует или уже закрыт, система вернет ошибку.

## Пример вызова PartyWS (SOAP)

### Пример 1. ИНН найден в ФНС

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <findInFns>
      <surname>Русин</surname>
      <name>Игорь</name>
      <patronymic>Антонович</patronymic>
      <birthdate>2000-08-13</birthdate>
      <documentCode>21</documentCode>
      <documentSeries>4507</documentSeries>
      <documentNumber>631945</documentNumber>
    </findInFns>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <findInFnsResponse>
      <found>true</found>
      <inn>111111110000</inn>
    </findInFnsResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 2. ИНН не найден в ФНС

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <findInFns>
      <surname>Русин</surname>
      <name>Игорь</name>
      <patronymic>Антонович</patronymic>
      <birthdate>2000-08-13</birthdate>
    </findInFns>
  </soapenv:Body>
</soapenv:Envelope>
```



```
<documentCode>21</documentCode>
<documentSeries>4507</documentSeries>
<documentNumber>631945</documentNumber>
</findInFns>
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
<soapenv:Header/>
<soapenv:Body>
<findInFnsResponse>
<found>>false</found>
<inn/>
<fnsErrorCode>inn.not.found</fnsErrorCode>
<fnsErrorMessage>Невозможно предоставить ИНН по указанным в запросе
сведениям о НП</fnsErrorMessage>
</findInFnsResponse>
</soapenv:Body>
</soapenv:Envelope>
```

## Пример 2. Модуль актуализации ИНН физлиц не подключен, ошибка

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
<soapenv:Header/>
<soapenv:Body>
<findInFns>
<surname>Русин</surname>
<name>Игорь</name>
<patronymic>Антонович</patronymic>
<birthdate>2000-08-13</birthdate>
<documentCode>21</documentCode>
<documentSeries>4507</documentSeries>
<documentNumber>631945</documentNumber>
</findInFns>
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<soap:Fault>
<faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:INTERNAL_ERROR<
/faultcode>
<faultstring>findInFns is not supported</faultstring>
<detail>
<errorMessage xmlns="http://hflabs.ru/cdi/soap/2_13">
<errorType>UnsupportedOperationException</errorType>
<errorMessage>Error finding in FNS: findInFns is not supported</errorMessage>
</errorMessage>
</detail>
</soap:Fault>
```

```
</soap:Body>  
</soap:Envelope>
```

## Пример вызова PartyRA (JSON)

### Пример 1. ИНН найден в ФНС

Запрос:

Запрос
<pre>{   "surname"      : "Русин",   "name"        : "Игорь",   "patronymic"  : "Антонович",   "birthdate"   : "2000-08-13",   "documentCode" : "21",   "documentSeries" : "45 07",   "documentNumber" : "631945" }</pre>

Ответ:

```
{  
  "found" : "true",  
  "inn"   : "111111110000"  
}
```

# PARTY.SEARCHRELATEDPARTIES

Метод позволяет найти связанных контрагентов.

- Описание
  - Входные параметры
  - Результирующие данные
  - Сценарий
  - Исключительные ситуации
  - Параметр returnSourceParties
  - Ограничения
  - Особенности использования
  - Дополнительная информация о контрагенте
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Все параметры заданы, связанные контрагенты найдены
  - Пример 2. Не задан критерий для поиска второго контрагента
  - Пример 3. Запрос с include
  - Пример 4. Запрос с параметром returnSourceParties
  - Пример 5. Контрагенты не связаны заданными типами связей
- Примеры вызова PartyRA (JSON)
  - Пример 1. Не заданы типы связей
  - Пример 2. Контрагенты не связаны заданными типами связей

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание	Комментарий
firstPartySearch/query	string	да	Критерии поиска первого контрагента	
firstPartySearch/partyType	WPartyType	да	Тип первого контрагента	
secondPartySearch/query	string	да	Критерии поиска второго контрагента	Блок secondPartySearch можно не указывать, если задан хотя бы один relationType
secondPartySearch/partyType	WPartyType	да	Тип второго контрагента	
relationTypes/relationType[]	Тип связи	нет	Тип взаимосвязи между контрагентами	Блок relationTypes можно не указывать, если задан блок secondPartySearch

include /partyInfo[]	Тип информации о контрагенте	нет	Дополнительная информация о контрагенте
returnSourceParties	boolean	нет	Возвращать исходные карточки найденных контрагентов.  По умолчанию false.  Если true, то обязательно указать типы связи relationTypes.

## Результирующие данные

Имя параметра	Тип	Описание
searchStatus	string	Статус поиска: <ul style="list-style-type: none"> <li>NOT_FOUND – по заданным критериям ничего не найдено</li> <li>TOO_MANY_FOUND_BY_FIRST – найдено больше 100 записей по первому контрагенту</li> <li>TOO_MANY_FOUND_BY_SECOND – найдено больше 100 записей по второму контрагенту</li> <li>NOT_ALL_ENTRIES_RETURNED – в ответе возвращены не все комбинации (по параметрам поиска нашлось слишком много связей)</li> </ul> <p>В остальных случаях статус в ответе будет отсутствовать.</p>
firstParty	WParty	Информация по контрагенту (если явно не заданы параметры <b>include/partyInfo[]</b> , то не содержит множественные атрибуты — адреса, телефоны, документы и т.д.)
secondParty	WParty	Информация по контрагенту (если явно не заданы параметры <b>include/partyInfo[]</b> , то не содержит множественные атрибуты — адреса, телефоны, документы и т.д.)
relation[]	WRelation	Взаимосвязи между контрагентами

## Сценарий

1. Выполняется поиск первого контрагента;
2. Выполняется поиск второго контрагента;
3. Между всеми записями, полученными на шагах 1 и 2, выполняется поиск связи с заданным типом. Пары записей, между которыми есть указанная связь, возвращаются в ответе.

## Исключительные ситуации

- Если на шагах 1 и 2 найдено 0 записей, то возвращается ответ NOT\_FOUND.
- Если на шагах 1 и 2 найдено больше 100 записей, то возвращается ответ TOO\_MANY\_FOUND\_BY\_FIRST (для первого КА) или TOO\_MANY\_FOUND\_BY\_SECOND (для второго КА).
- Если в запросе не задан тип связи, то в ответе возвращаются все найденные на шагах 1 и 2 записи, которые связаны между собой любым типом связи.

- Если в запросе не задан тип связи и записи, найденные на шагах 1 и 2, не связаны между собой, то возвращается ответ `NOT_FOUND`.
- Если в запросе не указан блок `secondPartySearch`, в ответе возвращаются записи, найденные на шаге 1, которые связаны с любыми контрагентами с заданными в запросе типами связей.
- Если в запросе не указан блок `secondPartySearch` и записи, найденные на шаге 1, не связаны ни с одним контрагентом с заданными в запросе типами связей, то возвращается ответ `NOT_FOUND`.
- Если по параметрам поиска нашлось слишком много связей (параметр `webservice.searchRelatedParties.response.maxEntries`), то помимо пар связанных записей в ответе также вернется статус `NOT_ALL_ENTRIES_RETURNED`.

### *Параметр `returnSourceParties`*

Если параметр `returnSourceParties = true`, в ответе возвращаются только связанные исходные контрагенты: первый контрагент — влитый или одиночный, второй контрагент — влитый или одиночный.

Если один из концов связи объединенная золотая карточка, т.е. связи нет на влитых исходных — такая связанная пара не вернется в ответе.

### *Ограничения*

- В ответе возвращается не более 20 пар связанных записей (параметр `webservice.searchRelatedParties.response.maxEntries`).
- Блок **`secondPartySearch`** и блок **`relationTypes`** не могут быть пустыми одновременно в рамках одного запроса.
- Если длина критерия поиска превышает 2000 символов, то система вернет ошибку.

### *Особенности использования*

Поиск осуществляется по всем атрибутам контрагента (ФИО, дата рождения, адреса, документы, и т.д.). В ответе по каждому контрагенту возвращаются только базовые поля, без множественных атрибутов.

Если контрагент был объединен и слит в золотую карточку, то в ответе вернется золотая карточка.

Поиск осуществляется только по нормализованным значениям.

При поиске контрагентов по атрибутам следует учитывать следующие особенности:

#### **1. Возможные форматы строки для телефона:**

- "{код страны} {код} {номер телефона}", например, "7 926 1234567";
- "{код} {номер телефона}", например, "926 1234567"; (работает с версии 3.14)
- "{номер телефона}", например, "1234567";

- 2. Серию и номер документа** в строке поиска следует указывать точно так же, как они сохранены в карточке контрагента. Пример строки поиска для прошедшего стандартизацию паспорта РФ: "45 06 123456".
- 3. Адрес** должен быть указан без точек и запятых, через пробел. Например, клиента с адресом "ул. Ленина, д. 1, к. 1, кв 55" можно найти двумя способами:
  - в строке поиска указать только название улицы: "Ленина";
  - по полному адресу, разделенному пробелами: "Ленина 1 1 55".

В качестве критерия для поиска возможно задание условия [расширенного поиска](#):

"?PhysicalParty.surname=Пашук AND PhysicalParty.name=Александр AND PhysicalParty.patronymic=Геннадьевич AND Passport.documentSeries=XXVIMЮ AND Passport.documentNumber=734419 AND PhysicalParty.birthdate=21.04.196"

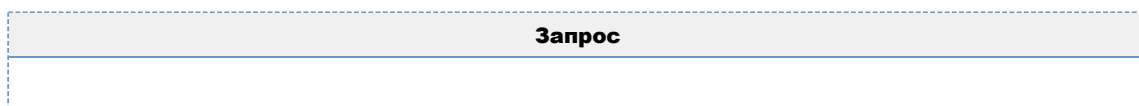
## Дополнительная информация о контрагенте

Если задан параметр **//include**, то система вернет только те данные контрагента, которые в нем перечислены.

Значения параметра //include /partyInfo	Описание	Комментарий
REQUISITE	Информация о реквизитах	
ATTRIBUTE	Информация об атрибутах	Информация о согласиях возвращается только если для параметра <b>web.service.response.attributes.defaultValues</b> выставлено значение <b>true</b> .
SOURCE	Информация о привязке к исходным контрагентам	
RELATION	Информация о связях	Возвращает тип связи, информацию о связанной сущности. Для подтверждаемых связей возвращает также статус связи (пример 2).
PAST_ATTRIBUTE	Информация о прошлых значениях ФИО и документов	<p>Возврат прошлых значений зависит только от заданного <b>include PAST_ATTRIBUTE</b> и не зависит от параметра <b>webservice.response.attributes.pastValues</b>.</p> <p>Информация о найденных прошлых значениях возвращается в сущности <b>pastAttribute</b>.</p> <p>Если прошлые значения не найдены, блок полей в ответе возвращается пустым, ошибки нет.</p>

## Примеры вызова PartyWS (SOAP)

### Пример 1. Все параметры заданы, связанные контрагенты найдены



```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchRelatedParties>
      <firstPartySearch>
        <query>Алексей</query>
        <partyType>PHYSICAL</partyType>
      </firstPartySearch>
      <secondPartySearch>
        <query>Союз</query>
        <partyType>LEGAL</partyType>
      </secondPartySearch>
      <relationTypes>
        <relationType>11</relationType>
      </relationTypes>
    </searchRelatedParties>
  </soapenv:Body>
</soapenv:Envelope>

```

Часть полей в ответе опущена для краткости: ФЛ «Морозов Алексей» и ЮЛ «БАНК СОЮЗ» связаны между собой 11 типом связи.

#### Ответ

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <searchRelatedPartiesResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <relatedPartiesEntry>
        <firstParty type="PHYSICAL" hid="1024" sourceSystem="DS7" rawId="1">
          <field name="sparkCheckStatus">NOT_FOUND</field>
          <field name="surname">Морозов</field>
          <field name="name">Алексей</field>
          <field name="patronymic">Витальевич</field>
          <field name="gender">MALE</field>
          <field name="genderRawSource"/>
          <field name="fullNameQC">EDITED</field>
          <field name="fullNameAuthor">DS7:1</field>
          <field name="fullNameRawSource">Морозов • Алексей • Витальевич</field>
        </firstParty>
        <secondParty type="LEGAL" hid="1030" sourceSystem="DS7" rawId="11">
          <field name="name">БАНК СОЮЗ</field>
          <field name="nameRawSource">Банк СОЮЗ (АО)</field>
          <field name="nameQC">EDITED</field>
          <field name="shortName">**СОЮЗ</field>
          <field name="opf"/>
        </secondParty>
        <relation type="11">
          <first type="PHYSICAL" hid="1024"/>
          <second type="LEGAL" hid="1030"/>
        </relation>
      </relatedPartiesEntry>
    </searchRelatedPartiesResponse>
  </soap:Body>
</soap:Envelope>

```

### Пример 2. Не задан критерий для поиска второго контрагента

#### Запрос

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <searchRelatedParties>
      <firstPartySearch>
        <query>Бубликов</query>
        <partyType>PHYSICAL</partyType>
      </firstPartySearch>
      <relationTypes>
        <relationType>15</relationType>
        <relationType>9</relationType>
        <relationType>500</relationType>
      </relationTypes>
    </searchRelatedParties>
  </soapenv:Body>
</soapenv:Envelope>

```

Часть полей в ответе опущена для краткости, для подтверждаемой связи 500 также возвращен статус связи:

Ответ
<pre> &lt;soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;soap:Body&gt;     &lt;searchRelatedPartiesResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;       &lt;relatedPartiesEntry&gt;         &lt;firstParty hid="1" type="PHYSICAL"&gt;           &lt;field name="name"&gt;Степан&lt;/field&gt;         &lt;/firstParty&gt;         &lt;secondParty hid="2" type="PHYSICAL"&gt;           &lt;field name="name"&gt;Петр&lt;/field&gt;         &lt;/secondParty&gt;         &lt;relation type="500"&gt;           &lt;first type="PHYSICAL" hid="1"/&gt;           &lt;second type="PHYSICAL" hid="2"/&gt;           &lt;field name="status"&gt;UNKNOWN&lt;/field&gt;         &lt;/relation&gt;       &lt;/relatedPartiesEntry&gt;       &lt;relatedPartiesEntry&gt;         &lt;firstParty hid="1" type="PHYSICAL"&gt;           &lt;field name="name"&gt;Степан&lt;/field&gt;         &lt;/firstParty&gt;         &lt;secondParty hid="1" type="LEGAL"&gt;           &lt;field name="name"&gt;ОАО Панорама&lt;/field&gt;         &lt;/secondParty&gt;         &lt;relation type="9"&gt;           &lt;first type="PHYSICAL" hid="1"/&gt;           &lt;second type="LEGAL" hid="1"/&gt;         &lt;/relation&gt;       &lt;/relatedPartiesEntry&gt;       &lt;relatedPartiesEntry&gt;         &lt;firstParty hid="1" type="PHYSICAL"&gt;           &lt;field name="name"&gt;Степан&lt;/field&gt;         &lt;/firstParty&gt;         &lt;secondParty hid="2" type="LEGAL"&gt;           &lt;field name="name"&gt;ООО Горизонт&lt;/field&gt;         &lt;/secondParty&gt;         &lt;relation type="9"&gt;           &lt;first type="PHYSICAL" hid="1"/&gt;           &lt;second type="LEGAL" hid="2"/&gt;         &lt;/relation&gt;       &lt;/relatedPartiesEntry&gt;     &lt;/searchRelatedPartiesResponse&gt;   &lt;/soap:Body&gt; &lt;/soap:Envelope&gt; </pre>



```

    </relation>
  </relatedPartiesEntry>
</searchRelatedPartiesResponse>
</soap:Body>
</soap:Envelope>

```

### Пример 3. Запрос с include

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;searchRelatedParties&gt;       &lt;firstPartySearch&gt;         &lt;query&gt;Алексей&lt;/query&gt;         &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;       &lt;/firstPartySearch&gt;       &lt;secondPartySearch&gt;         &lt;query&gt;Союз&lt;/query&gt;         &lt;partyType&gt;LEGAL&lt;/partyType&gt;       &lt;/secondPartySearch&gt;       &lt;relationTypes&gt;         &lt;relationType&gt;11&lt;/relationType&gt;       &lt;/relationTypes&gt;       &lt;include&gt;         &lt;partyInfo&gt;REQUISITE&lt;/partyInfo&gt;       &lt;/include&gt;     &lt;/searchRelatedParties&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Часть полей в ответе опущена для краткости:

Ответ
<pre> &lt;soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;soap:Body&gt;     &lt;searchRelatedPartiesResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;       &lt;relatedPartiesEntry&gt;         &lt;firstParty type="PHYSICAL" hid="1024" sourceSystem="DS7" rawId="1"&gt;           &lt;field name="sparkCheckStatus"&gt;NOT_FOUND&lt;/field&gt;           &lt;field name="surname"&gt;Морозов&lt;/field&gt;           &lt;field name="name"&gt;Алексей&lt;/field&gt;           &lt;field name="patronymic"&gt;Витальевич&lt;/field&gt;           &lt;field name="gender"&gt;MALE&lt;/field&gt;           &lt;field name="genderRawSource"/&gt;           &lt;field name="fullNameQC"&gt;EDITED&lt;/field&gt;           &lt;field name="fullNameAuthor"&gt;DS7:1&lt;/field&gt;           &lt;field name="fullNameRawSource"&gt;Морозов • Алексей • Витальевич&lt;/field&gt;         &lt;/firstParty&gt;         &lt;secondParty type="LEGAL" hid="1030" sourceSystem="DS7" rawId="11"&gt;           &lt;field name="name"&gt;БАНК СОЮЗ&lt;/field&gt;           &lt;field name="nameRawSource"&gt;Банк СОЮЗ (АО)&lt;/field&gt;           &lt;field name="nameQC"&gt;EDITED&lt;/field&gt;           &lt;field name="shortName"&gt;**СОЮЗ&lt;/field&gt;           &lt;field name="opf"/&gt;         &lt;/secondParty&gt;         &lt;relation type="11"&gt;           &lt;first type="PHYSICAL" hid="1024"/&gt; </pre>

```

    <second type="LEGAL" hid="1030"/>
  </relation>
</relatedPartiesEntry>
</searchRelatedPartiesResponse>
</soap:Body>
</soap:Envelope>

```

#### Пример 4. Запрос с параметром *returnSourceParties*

В базе ФЛ и ЮЛ:

- объединенный ФЛ (hid=1026), состоящий из AL:1 (hid=1024), и AL:2 (hid=1025)
- одиночный ЮЛ DS:7 (hid=1027) (наименование = Союз)
- объединенный ЮЛ (hid=1030, наименование = Союз), состоящий из DS:8 (hid=1028), и DS:9 (hid=1029)
- AL:1 и DS:7 связаны 11 связью
- объединенный ФЛ 1026 связан с объединенным ЮЛ 1030 11 связью

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;searchRelatedParties&gt;       &lt;firstPartySearch&gt;         &lt;query&gt;Алексей&lt;/query&gt;         &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;       &lt;/firstPartySearch&gt;       &lt;secondPartySearch&gt;         &lt;query&gt;Союз&lt;/query&gt;         &lt;partyType&gt;LEGAL&lt;/partyType&gt;       &lt;/secondPartySearch&gt;       &lt;relationTypes&gt;         &lt;relationType&gt;11&lt;/relationType&gt;       &lt;/relationTypes&gt;       &lt;returnSourceParties&gt;true&lt;/returnSourceParties&gt;     &lt;/searchRelatedParties&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Часть полей в ответе опущена для краткости:

- возвращается связанная пара исходных карточек AL:1 и DS:7
- связанная пара ФЛ 1026 и ЮЛ 1030 не возвращается, связь есть только на золотых объединенных карточках, на исходных ее нет

Ответ
<pre> &lt;soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;soap:Body&gt;     &lt;searchRelatedPartiesResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;       &lt;relatedPartiesEntry&gt;         &lt;firstParty type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"&gt;           &lt;field name="surname"&gt;Морозов&lt;/field&gt; </pre>

```

    <field name="name">Алексей</field>
    <field name="patronymic">Витальевич</field>
    <field name="fullNameRawSource">Морозов • Алексей • Витальевич</field>
  </firstParty>
  <secondParty type="LEGAL" hid="1027" sourceSystem="DS" rawId="7">
    <field name="name">БАНК СОЮЗ</field>
    <field name="nameRawSource">Банк СОЮЗ (АО)</field>
    <field name="nameQC">EDITED</field>
    <field name="shortName">**СОЮЗ</field>
    <field name="opf"/>
  </secondParty>
  <relation type="11">
    <first type="PHYSICAL" hid="1024"/>
    <second type="LEGAL" hid="1027"/>
  </relation>
</relatedPartiesEntry>
</searchRelatedPartiesResponse>
</soap:Body>
</soap:Envelope>

```

### Пример 5. Контрагенты не связаны заданными типами связей

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;searchRelatedParties&gt;       &lt;firstPartySearch&gt;         &lt;query&gt;Алексей&lt;/query&gt;         &lt;partyType&gt;PHYSICAL&lt;/partyType&gt;       &lt;/firstPartySearch&gt;       &lt;secondPartySearch&gt;         &lt;query&gt;Союз&lt;/query&gt;         &lt;partyType&gt;LEGAL&lt;/partyType&gt;       &lt;/secondPartySearch&gt;       &lt;relationTypes&gt;         &lt;relationType&gt;11&lt;/relationType&gt;       &lt;/relationTypes&gt;       &lt;include&gt;         &lt;partyInfo&gt;REQUISITE&lt;/partyInfo&gt;       &lt;/include&gt;     &lt;/searchRelatedParties&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>
Ответ
<pre> &lt;searchRelatedPartiesResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;searchStatus&gt;NOT_FOUND&lt;/searchStatus&gt; &lt;/searchRelatedPartiesResponse&gt; </pre>

## Примеры вызова PartyRA (JSON)

### Пример 1. Не заданы типы связей

Запрос
<pre>{   "firstPartySearch": {     "query": "Петров",     "partyType": "PHYSICAL"   },   "secondPartySearch": {     "query": "Мария Петровна",     "partyType": "PHYSICAL"   } }</pre>

Часть полей в ответе опущена для краткости:

Ответ
<pre>{   "relatedPartiesEntry": [     {       "firstParty": {         "field": [           {             "value": "Петров",             "name": "surname"           },           {             "value": "Юрий",             "name": "name"           },           {             "value": "Сергеевич",             "name": "patronymic"           },           {             "value": "MALE",             "name": "gender"           }         ]       },       "hid": 2778,       "sourceSystem": null,       "rawId": null,       "type": "PHYSICAL"     },     {       "secondParty": {         "field": [           {             "value": "Ипатова",             "name": "surname"           },           {             "value": "Мария",             "name": "name"           },           { </pre>

```

      "value": "Петровна",
      "name": "patronymic"
    },
    {
      "value": "FEMALE",
      "name": "gender"
    }
  ],
  "hid": 1320,
  "sourceSystem": "AL",
  "rawId": "21001012",
  "type": "PHYSICAL"
},
"relation": [ {
  "first": {
    "type": "PHYSICAL",
    "hid": 2778,
    "rawId": null,
    "sourceSystem": null
  },
  "second": {
    "type": "PHYSICAL",
    "hid": 1320,
    "rawId": null,
    "sourceSystem": null
  },
  "type": 6,
  "deleted": null,
  "hashCode": "0ef6cef7b2685149bcd0a813f472d4c8"
}
}]
}}
```

### Пример 2. Контрагенты не связаны заданными типами связей

Запрос
<pre> {   "firstPartySearch": {     "query": "Петров",     "partyType": "PHYSICAL"   },   "secondPartySearch": {     "query": "Мария Петровна",     "partyType": "PHYSICAL"   },   "relationTypes": {     "relationType": ["5", "7", "9"]   } } </pre>
Ответ
<pre> {} </pre>

# PARTY.CHECKBLACKLIST

Метод позволяет проверить контрагента по черным спискам и перечню недействительных паспортов.

! Ответы модуля ЧС 1.0 и ЧС 2.0 могут отличаться!

- Описание

  - Входные параметры

  - Результирующие данные

  - Особенности использования

- Примеры вызова PartyWS (SOAP)

  - Пример 1. Поиск по ФИО и ДР для ФЛ. Совпадения найдены

  - Пример 2. Поиск по ФИО, ИНН и месту рождения. По заданным критериям ничего не найдено

  - Пример 3. Поиск по наименованию ЮЛ, ИНН, КПП, ОГРН и адресу. По заданным критериям ничего не найдено

  - Пример 4. Найден в списке ПНП (перечня недействительных паспортов)

- Примеры вызова PartyRA (JSON)

  - Пример 1. Поиск по ДУЛ для ФЛ. Совпадения найдены по ПНП.

  - Пример 2. Поиск по ДУЛ, ФИО и ДР для ФЛ. Совпадения найдены по ПНП и по Стоп-листу.

  - Пример 3. Поиск по ФИО, ДР и паспорту. По заданным критериям ничего не найдено

## Описание


### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
party	WParty	да	Информация по контрагенту
includePartyFields	boolean	нет	Возвращать реквизиты найденных карточек ЧС. По умолчанию false.

### Результирующие данные

Имя параметра	Тип	Описание
party	WMatchedParty[]	Информация по найденным карточкам ЧС
party	WRiskParty	Информация по выявленному уровню риска клиента

## Особенности использования

 Метод предназначен для быстрой проверки контрагента по черным спискам. Работает только с данными, которые были переданы на вход, без привязки к карточкам клиентов в БД CDI.

Метод работает по следующему алгоритму:

1. Произвести стандартизацию переданной записи без сохранения контрагента в системе.
2. Найти дубликаты между ним и «Черными списками».
3. Найти дубликаты между ним и перечнем недействительных паспортов (ПНП).
4. В ответе вернуть всех найденных на шаге 2 дубликатов с указанием:
  - a. правила идентификации дубликатов, по которому был найден каждый из дубликатов (для ПНП всегда правило 51);
  - b. коэффициента соответствия для правила, по которому контрагенты были идентифицированы как дубликаты (для ПНП всегда 100);
  - c. уровня риска (рассчитывается с учетом настроенных параметров, индивидуально для каждого заказчика) (актуально только для модуля ЧС 2.0).

Варианты ответа:

1. Человек найден в ПНП и других ЧС — вернуть список ЧС, в которых он найден + совпадение с ПНП.
2. Человек найден в ПНП и не найден в других ЧС — вернуть совпадение с ПНП.
3. Человек не найден в ПНП и найден в других ЧС — вернуть список ЧС, в которых он найден.
4. Человек не найден в ПНП и не найден в других ЧС — вернуть пустой ответ.

Возвращаемые методом контрагенты отсортированы в порядке убывания коэффициента соответствия.

Если параметр `includePartyFields = true`, то в ответе будут возвращены все реквизиты и атрибуты найденных карточек Черных списков.

Для каждой карточки возвращается её источник (`sourceSystem`) и идентификатор (`rawId`), кроме перечня недействительных паспортов. Для паспортов указывается только **sourceSystem="INVALID\_PASSPORT"**, без идентификатора.

Помимо уровня риска, возвращается и дата присвоения уровня риска (`amlRiskDate`) (дата вызова метода).

## Примеры вызова PartyWS (SOAP)

### Пример 1. Поиск по ФИО и ДР для ФЛ. Совпадения найдены

Запрос:

```


Запрос


<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklist>
      <party type="PHYSICAL">
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
        <field name="birthdate">03.12.1987</field>
        <field name="birthdateRawSource">03.12.1987</field>
      </party>
    </checkBlacklist>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ модуля ЧС 1.0. Найдено 2 карточки ЧС. Для каждой карточки возвращается ее источник (**sourceSystem**) и идентификатор (**rawId**). В данном примере есть совпадение со списком террористов (**sourceSystem=TERRORIST**) и списком ИПДЛ (**sourceSystem=PFA**):

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklistResponse>
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="BLACKLIST" sourceSystem="TERRORIST_CB" rawId="1">
          </party>
        </matchedParty>
        <matchedParty>
          <matchRule>23</matchRule>
          <matchScope>60</matchScope>
          <party type="BLACKLIST" sourceSystem="PFA" rawId="5">
            </party>
          </matchedParty>
        </checkBlacklistResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ ЧС 2.0 (В примере совпадение со списком террористов является высоким уровнем риска HIGH (настраивается посредством параметров сервисов)):

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
```



```

<soapenv:Body>
  <checkBlacklistResponse>
    <party type="PHYSICAL">
      <field name="amlRiskLevel">HIGH</field>
      <field name="amlRiskDate">11.05.2018</field>
    </party>
    <matchedParty>
      <matchRule>1</matchRule>
      <matchScope>100</matchScope>
      <party type="PHYSICAL_LIST" hid="1024" sourceSystem="TERRORIST_CB" rawId="
1">
    </party>
  </matchedParty>
  <matchedParty>
    <matchRule>23</matchRule>
    <matchScope>60</matchScope>
    <party type="PHYSICAL_LIST" hid="1026" sourceSystem="PFA" rawId="
5">
  </party>
</matchedParty>
</checkBlacklistResponse>
</soapenv:Body>
</soapenv:Envelope>

```

*Пример 2. Поиск по ФИО, ИНН и месту рождения. По заданным критериям ничего не найдено*

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;checkBlacklist&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="surname"&gt;МАМОНОВСКИЙ&lt;/field&gt;         &lt;field name="name"&gt;СЕРГЕЙ&lt;/field&gt;         &lt;field name="patronymic"&gt;ВИКТОРОВИЧ&lt;/field&gt;         &lt;field name="inn"&gt;772375622656&lt;/field&gt;         &lt;field name="birthPlace"&gt;г. Москва&lt;/field&gt;       &lt;/party&gt;     &lt;/checkBlacklist&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ модуля ЧС 1.0 и ЧС 2.0 (уровень риска возвращается только в ответах модуля ЧС 2.0):

<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Body&gt;     &lt;checkBlacklistResponse xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;       &lt;party type="PHYSICAL"&gt;         &lt;field name="amlRiskLevel"&gt;MISSED&lt;/field&gt;         &lt;field name="amlRiskDate"&gt;26.09.2022 08:52:25&lt;/field&gt;       &lt;/party&gt;     &lt;/checkBlacklistResponse&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>
--

**Пример 3. Поиск по наименованию ЮЛ, ИНН, КПП, ОГРН и адресу. По заданным критериям ничего не найдено**

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklist>
      <party type="LEGAL">
        <field name="nameRawSource">ООО «Ромашка»</field>
        <field name="kpp">773501001</field>
        <field name="inn">7735542429</field>
        <field name="ogrn">1087746823350</field>
        <attribute type="ADDRESS" rawId="CFT.1">
          <field name="rawSource">Россия, Москва, Турчанинов пер., д.6 стр.2</field>
        </attribute>
        <attribute type="ADDRESS" rawId="CFT.2">
          <field name="rawSource">Россия, Москва, Фрунзенская наб., д.15</field>
        </attribute>
      </party>
    </checkBlacklist>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ ЧС 1.0 и ЧС 2.0 (уровень риска возвращается только в ответах модуля ЧС 2.0):

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Body>
    <checkBlacklistResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <party type="LEGAL">
        <field name="amlRiskLevel">MISSED</field>
        <field name="amlRiskDate">26.09.2022 08:52:25</field>
      </party>
    </checkBlacklistResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

**Пример 4. Найден в списке ПНП (перечня недействительных паспортов)**

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklist>
      <party type="PHYSICAL">
        <field name="surname">МАМОНОВ</field>
        <field name="name">СЕРГЕЙ</field>
        <field name="patronymic">ВИКТОРОВИЧ</field>
        <field name="fullNameRawSource">Иванов Сергей Викторович</field>
      </party>
    </checkBlacklist>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <attribute type="DOCUMENT_PASSPORT" rawId="111">
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="type">PASSPORT</field>
      <field name="documentSeries">6003</field>
      <field name="documentNumber">973081</field>
    </attribute>
  </party>
</checkBlacklist>
</soapenv:Body>
</soapenv:Envelope>

```

Ответ модуля ЧС 1.0 и ЧС 2.0 (Найдено 1 совпадение — с ПНП. Источник (**sourceSystem**) и идентификатор (**rawId**) для паспорта не указываются. Если нужны сами данные, нужно вызвать метод с параметром **includePartyFields**. Уровень риска возвращается только в ответах модуля ЧС 2.0: в примере совпадение со списком недействительных паспортов является средним уровнем риска MEDIUM (настраивается посредством параметров сервисов)).

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Body>
    <checkBlacklistResponse>
      <party type="PHYSICAL">
        <field name="amlRiskLevel">MEDIUM</field>
        <field name="amlRiskDate">11.05.2018</field>
      </party>
      <matchedParty>
        <matchRule>51</matchRule>
        <matchScope>100</matchScope>
        <party type="BLACKLIST" sourceSystem="INVALID_PASSPORT">
          </party>
        </matchedParty>
      </checkBlacklistResponse>
    </soapenv:Body>
  </soapenv:Envelope>

```

**Пример 5. Найден в списках Terrorist и Недействительных паспортов, в ответе вернем ре квизиты найденных карточек ЧС**

Запрос:

```

                                Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header>
  <soapenv:Body>
    <checkBlacklist>
      <_13:party type="PHYSICAL">
        <_13:field name="fullNameRawSource">Манжуров Василий Андреевич</field>
        <field name="name">Василий</field>
        <field name="patronymic">Андреевич</field>
        <field name="surname">Манжуров</field>
        <field name="birthdate">03.08.1995</field>
        <field name="birthdateRawSource">03.08.1995</field>
        <attribute type="DOCUMENT_PASSPORT">
          <field name="qualifierType">PASSPORT_RU</field>
          <field name="documentNumber">0809899063</field>
        </attribute>
      </party>

```

```

<includePartyFields>true</includePartyFields>
</checkBlacklist>
</soapenv:Body>
</soapenv:Envelope>

```

Ответ модуля ЧС 1.0: (найденно два совпадения реквизиты карточки ЧС возвращаются, так как использован параметр includePartyFields)

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <checkBlacklistResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <matchedParty>
        <matchRule>1</matchRule>
        <matchScope>100</matchScope>
        <party type="BLACKLIST" sourceSystem="TERRORIST" rawId="
2016_190_02_00000000000A7BAC">
          <field name="allNames">Манжуров Василий Андреевич, Монжура Василий
Андреевич, Manjurov Vasiliy Andreevich</field>
          <field name="allNamesAuthor">TERRORIST:2016_190_02_00000000000A7BAC<
/field>
          <field name="allBirthdates">03.08.1995, 02.10.1963</field>
          <field name="allBirthdatesAuthor">TERRORIST:2016_190_02_00000000000A7BAC<
/field>
          <field name="allDocumentNumbers">0809899063</field>
          <field name="allDocumentNumbersAuthor">TERRORIST:
2016_190_02_00000000000A7BAC</field>
          <field name="blacklistId">2016_190_02_00000000000A7BAC</field>
          <field name="blacklistName">TERRORIST</field>
          <field name="rule">1</field>
          <field name="matchScope">100</field>
          <field name="blacklistIdAuthor">TERRORIST:2016_190_02_00000000000A7BAC<
/field>
          <field name="surname">МАНЖУРОВ</field>
          <field name="name">ВАСИЛИЙ</field>
          <field name="patronymic">АНДРЕЕВИЧ</field>
          <field name="nameRawSource">Манжуров Василий Андреевич</field>
          <field name="nameAuthor">TERRORIST:2016_190_02_00000000000A7BAC</field>
          <field name="birthdate">03.08.1995</field>
          <field name="birthdateAuthor">TERRORIST:2016_190_02_00000000000A7BAC<
/field>
          <field name="birthPlace">Казахская ССР, Жамбылская область, Шуйский
район, село Тана</field>
          <field name="birthPlaceAuthor">TERRORIST:2016_190_02_00000000000A7BAC<
/field>
          <field name="inn"/>
          <field name="innAuthor">TERRORIST:2016_190_02_00000000000A7BAC</field>
          <field name="documentNumber">0809899063</field>
          <field name="documentNumberAuthor">TERRORIST:
2016_190_02_00000000000A7BAC</field>
          <field name="bankruptcyDate"/>
          <field name="bankruptcyDateAuthor">TERRORIST:
2016_190_02_00000000000A7BAC</field>
          <field name="bankruptcyStatus"/>
          <field name="bankruptcyStatusAuthor">TERRORIST:
2016_190_02_00000000000A7BAC</field>
          <field name="bankruptcyExitDate"/>
          <field name="bankruptcyExitDateAuthor">TERRORIST:
2016_190_02_00000000000A7BAC</field>
          <field name="actualityDateAuthor">TERRORIST:2016_190_02_00000000000A7BAC<
/field>
          <field name="actualityDate">01.11.2021 00:00:00</field>

```

```

<field name="lastChangeTimestamp">1640175447627</field>
<attribute type="ADDRESS" rawId="TERRORIST:2016_190_02_0000000000A7BAC"
>
  <field name="qualityCode">GOOD</field>
  <field name="validationCode">MISSED</field>
  <field name="type">HOME</field>
  <field name="kladrActualityDate"/>
  <field name="postalCode">361425</field>
  <field name="kladrPostalCode"/>
  <field name="country">РОССИЯ</field>
  <field name="district"/>
  <field name="regionType"/>
  <field name="region">РЕСП КАБАРДИНО-БАЛКАРСКАЯ</field>
  <field name="rayonType"/>
  <field name="rayon">Р-Н ЧЕГЕМСКИЙ</field>
  <field name="cityType"/>
  <field name="city"/>
  <field name="cityAreaType"/>
  <field name="cityArea"/>
  <field name="settlementType"/>
  <field name="settlement">С ШАЛУШКА</field>
  <field name="streetType"/>
  <field name="street">УЛ ИМ А.А.ЦИПИНОВА</field>
  <field name="houseNumber">125</field>
  <field name="militaryNumber"/>
  <field name="postalBox"/>
  <field name="korpus"/>
  <field name="stroenie"/>
  <field name="vladenie"/>
  <field name="hostel"/>
  <field name="floor"/>
  <field name="entrance"/>
  <field name="section"/>
  <field name="flat"/>
  <field name="flat2"/>
  <field name="office"/>
  <field name="office2"/>
  <field name="room"/>
  <field name="room2"/>
  <field name="kladrCode"/>
  <field name="okatoCode"/>
  <field name="fiasCode"/>
  <field name="fiasLevel">EMPTY</field>
  <field name="validationExtension"/>
  <field name="popularity">NONE</field>
  <field name="rawSource">РОССИЙСКАЯ ФЕДЕРАЦИЯ, КАБАРДИНО-
БАЛКАРСКАЯ РЕСПУБЛИКА, С. ШАЛУШКА, УЛ. СЕВЕРНАЯ, Д. 125,</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="actualityDate"/>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
</party>
</matchedParty>
<matchedParty>
  <matchRule>51</matchRule>
  <matchScope>100</matchScope>
  <party type="BLACKLIST" sourceSystem="INVALID_PASSPORT">
    <attribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1">
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="issueDate"/>
      <field name="issueAuthority"/>
    </attribute>
  </party>
</matchedParty>

```

```

<field name="departmentCode"/>
<field name="expiryDate"/>
<field name="invalidStatus">INVALID</field>
<field name="popularity">NONE</field>
<field name="qualityCode">EDITED</field>
<field name="type">PASSPORT</field>
<field name="documentSeries">08 09</field>
<field name="documentNumber">899063</field>
<field name="rawSource">номер 0809899063</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="actualityDate">10.11.2022</field>
</attribute>
<source sourceSystem="TERRORIST" rawId="2016_190_02_0000000000A7BAC"/>
</party>
</matchedParty>
</checkBlacklistResponse>
</soap:Body>
</soap:Envelope>

```

Ответ модуля ЧС 2.0:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <checkBlacklistResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <party type="LEGAL">
        <field name="amlRiskLevel">MISSED</field>
        <field name="amlRiskDate">26.09.2022 08:52:25</field>
      </party>
      <matchedParty>
        <matchRule>2</matchRule>
        <matchScope>980</matchScope>
        <party type="PHYSICAL_LIST" hid="7810474" sourceSystem="TERRORIST" rawId="
2016_190_02_0000000000A7BAC">
          <field name="inn"/>
          <field name="innAuthor">TERRORIST:2016_190_02_0000000000A7BAC</field>
          <field name="innQC">EMPTY</field>
          <field name="innRawSource"/>
          <field name="snils"/>
          <field name="snilsAuthor">TERRORIST:2016_190_02_0000000000A7BAC</field>
          <field name="snilsQC">EMPTY</field>
          <field name="snilsRawSource"/>
          <field name="bankruptcyStatus"/>
          <field name="bankruptcyStatusAuthor">TERRORIST:
2016_190_02_0000000000A7BAC</field>
          <field name="bankruptcyStatusDate"/>
          <field name="bankruptcyDate"/>
          <field name="bankruptcyExitDate"/>
          <field name="bankruptcyStage"/>
          <field name="bankruptcyType"/>
          <field name="bankruptcyDecisionDate"/>
          <field name="entryDate">01.04.2019</field>
          <field name="entryDateAuthor">TERRORIST:2016_190_02_0000000000A7BAC<
/field>
          <field name="denialCode"/>
            <field name="denialCodeAuthor"/>
          <field name="additionalInfoAuthor">TERRORIST:2016_190_02_0000000000A7BAC<
/field>
          <field name="createdDate">22.12.2021</field>
          <field name="actualityDateAuthor">TERRORIST:2016_190_02_0000000000A7BAC<
/field>

```

```

<field name="actualityDate">01.11.2021 00:00:00</field>
<field name="lastChangeTimestamp">1640175447627</field>
<attribute type="FIO" hid="31219" rawId="2016_190_02_0000000000A7BAC.
TERRORIST.FIO.1">
  <field name="qualityCode">EDITED</field>
  <field name="surname">Манжуров</field>
  <field name="name">Василий</field>
  <field name="patronymic">Андреевич</field>
  <field name="type">DEFAULT</field>
  <field name="rawSource">Манжуров Василий Андреевич</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
<attribute type="FIO" hid="31222" rawId="2016_190_02_0000000000A7BAC.
TERRORIST.FIO.4">
  <field name="qualityCode">EDITED</field>
  <field name="surname">Монжура</field>
  <field name="name">Василий</field>
  <field name="patronymic">Андреевич</field>
  <field name="type">DEFAULT</field>
  <field name="rawSource">Монжура Василий Андреевич</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
<attribute type="FIO" hid="31223" rawId="2016_190_02_0000000000A7BAC.
TERRORIST.FIO.5">
  <field name="qualityCode">NOT_SURE</field>
  <field name="surname">Манжуров</field>
  <field name="name">Василий</field>
  <field name="patronymic">Андреевич</field>
  <field name="type">DEFAULT</field>
  <field name="rawSource">Manjurov Vasiliy Andreevich</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
<attribute type="SIMPLE_BIRTHDATE" hid="24765" rawId="
2016_190_02_0000000000A7BAC.TERRORIST.BDATE.1">
  <field name="birthdate">03.08.1995</field>
  <field name="accuracy">DAY</field>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
<attribute type="SIMPLE_BIRTHDATE" hid="24765" rawId="
2016_190_02_0000000000A7BAC.TERRORIST.BDATE.2">
  <field name="birthdate">02.10.1963</field>
  <field name="accuracy">DAY</field>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
<attribute type="SIMPLE_BIRTH_PLACE" hid="13083" rawId="
2016_190_02_0000000000A7BAC.TERRORIST.BPLACE.1">
  <field name="birthPlace">Казахская ССР, Жамбылская область, Шуйский
район, село Тана</field>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>

```

```
<attribute type="DOCUMENT_PASSPORT" hid="5827428" rawId="
2016_190_02_0000000000A7BAC.TERRORIST.DOC.1">
  <field name="qualifierType"/>
  <field name="issueDate"/>
  <field name="issueAuthority">Казахская ССР</field>
  <field name="departmentCode"/>
  <field name="expiryDate"/>
  <field name="invalidStatus">ACTUAL</field>
  <field name="popularity">NONE</field>
  <field name="qualityCode">BAD_TYPE</field>
  <field name="type">PASSPORT</field>
  <field name="documentSeries"/>
  <field name="documentNumber">13601334</field>
  <field name="rawSource">•0809899063••Казахская ССР••</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="actualityDate">01.11.2021 00:00:00</field>
  <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
</attribute>
```

```
<attribute type="ADDRESS" hid="7784428" rawId="TERRORIST:
2016_190_02_0000000000A7BAC.ADDRESS.3">
  <field name="qualityCode">GOOD</field>
  <field name="validationCode">MISSED</field>
  <field name="type">HOME</field>
  <field name="kladrActualityDate"/>
  <field name="postalCode">361425</field>
  <field name="kladrPostalCode"/>
  <field name="country">РОССИЯ</field>
  <field name="district"/>
  <field name="regionType"/>
  <field name="region">РЕСП КАБАРДИНО-БАЛКАРСКАЯ</field>
  <field name="rayonType"/>
  <field name="rayon">Р-Н ЧЕГЕМСКИЙ</field>
  <field name="cityType"/>
  <field name="city"/>
  <field name="cityAreaType"/>
  <field name="cityArea"/>
  <field name="settlementType"/>
  <field name="settlement">С ШАЛУШКА</field>
  <field name="streetType"/>
  <field name="street">УЛ ИМ А.А.ЦИПИНОВА</field>
  <field name="houseNumber">125</field>
  <field name="militaryNumber"/>
  <field name="postalBox"/>
  <field name="korporus"/>
  <field name="stroenie"/>
  <field name="vladenie"/>
  <field name="hostel"/>
  <field name="floor"/>
  <field name="entrance"/>
  <field name="section"/>
  <field name="flat"/>
  <field name="flat2"/>
  <field name="office"/>
  <field name="office2"/>
  <field name="room"/>
  <field name="room2"/>
  <field name="kladrCode"/>
  <field name="okatoCode"/>
  <field name="fiasLevel">EMPTY</field>
  <field name="validationExtension"/>
  <field name="popularity">NONE</field>
```



```

    <field name="rawSource">РОССИЙСКАЯ ФЕДЕРАЦИЯ, КАБАРДИНО-
    БАЛКАРСКАЯ РЕСПУБЛИКА, С. ШАЛУШКА, УЛ. СЕВЕРНАЯ, Д. 125,</field>
    <field name="primary">false</field>
    <field name="additionalInfo"/>
    <field name="actualityDate">01.11.2021 00:00:00</field>
    <field name="author">TERRORIST:2016_190_02_0000000000A7BAC</field>
  </attribute>
  <source hid="7810474" sourceSystem="TERRORIST" rawId="
2016_190_02_0000000000A7BAC"/>
</party>
</matchedParty>
<matchedParty>
  <matchRule>51</matchRule>
  <matchScope>100</matchScope>
  <party type="BLACKLIST" sourceSystem="INVALID_PASSPORT">
    <attribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1">
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="issueDate"/>
      <field name="issueAuthority"/>
      <field name="departmentCode"/>
      <field name="expiryDate"/>
      <field name="invalidStatus">INVALID</field>
      <field name="popularity">NONE</field>
      <field name="qualityCode">EDITED</field>
      <field name="type">PASSPORT</field>
      <field name="documentSeries">08 09</field>
      <field name="documentNumber">899063</field>
      <field name="rawSource">номер 0809899063</field>
      <field name="primary">false</field>
      <field name="additionalInfo"/>
      <field name="actualityDate"/>
    </attribute>
  </party>
</matchedParty>
</checkBlacklistResponse>
</soap:Body>
</soap:Envelope>

```

## Примеры вызова PartyRA (JSON)

### Пример 1. Поиск по ДУЛ для ФЛ. Совпадения найдены по ПНП.

Запрос:

```

{
  "includePartyFields": "false",
  "party": {
    "type": "PHYSICAL",
    "attribute": [
      {
        "type": "DOCUMENT_PASSPORT",
        "field": [
          {
            "value": "PASSPORT_RU",
            "name": "qualifierType"
          },
          {
            "name": "documentSeries",

```

```

        "value": "6514"
      },
      {
        "name": "documentNumber",
        "value": "904762"
      }
    ]
  }
}

```

Ответ ЧС 1.0 и ЧС 2.0 (уровень риска возвращается только в ответах модуля ЧС 2.0):

```

{
  "party": {
    "field": [ {
      "value": "MISSED",
      "name": "amlRiskLevel"
    }, {
      "value": "26.09.2022 08:52:25",
      "name": "amlRiskDate"
    } ],
    "hid": null,
    "sourceSystem": null,
    "rawId": null,
    "type": "PHYSICAL"
  },
  "matchedParty": [
    {
      "matchRule": "51",
      "matchScope": 100,
      "party": {
        "hid": null,
        "sourceSystem": "INVALID_PASSPORT",
        "rawId": null,
        "type": "BLACKLIST"
      }
    }
  ]
}

```

**Пример 2. Поиск по ДУЛ, ФИО и ДР для ФЛ. Совпадения найдены по ПНП и по Стоп-листу.**

Запрос:

```

{
  "includePartyFields": "false",
  "party": {
    "type": "PHYSICAL",
    "field": [
      {
        "value": "28.12.1994",
        "name": "birthdateRawSource"
      },
      {
        "name": "fullNameRawSource",

```



```

    "name" : "amlRiskDate"
  } ],
  "hid" : null,
  "sourceSystem" : null,
  "rawId" : null,
  "type" : "PHYSICAL"
},
"matchedParty": [
  {
    "matchRule": "1",
    "matchScope": 100,
    "party": {
      "hid": 7807918,
      "sourceSystem": "TERRORIST",
      "rawId": "2016_190_02_0000000000A7BAC",
      "type": "PHYSICAL_LIST"
    }
  },
  {
    "matchRule": "51",
    "matchScope": 100,
    "party": {
      "hid": null,
      "sourceSystem": "INVALID_PASSPORT",
      "rawId": null,
      "type": "BLACKLIST"
    }
  }
]
}

```

*Пример 3. Поиск по ФИО, ДР и паспорту. По заданным критериям ничего не найдено*

Запрос:

```

{
  "includePartyFields": "false",
  "party": {
    "type": "PHYSICAL",
    "field": [
      {
        "value": "13.05.1965",
        "name": "birthdateRawSource"
      },
      {
        "name": "fullNameRawSource",
        "value": "Задорин Павел Макеев"
      }
    ]
  },
  "attribute": [
    {
      "type": "DOCUMENT_PASSPORT",
      "field": [
        {
          "value": "PASSPORT_RU",
          "name": "qualifierType"
        },
        {
          "name": "documentSeries",

```

```
    "value": "8969"
  },
  {
    "name": "documentNumber",
    "value": "551772"
  }
}
}
```


Ответ:

**<Empty JSON content>**

# PARTY.CHECKBLACKLISTBYRAWID

Метод проверяет контрагента по черным спискам по rawId и коду исходной системы и сразу создает связь между искомым КА и записью ЧС.

Также метод возвращает для паспорта КА статус «обнаружен в перечне недействительных документов» (INVALID), если документ уже был проверен по ПНП и получил такой статус.

 Только для модуля ЧС 2.0

- Описание
  - Входные параметры
  - Результирующие данные
  - Алгоритм
  - Особенности использования
- Логика создания связей «КА — список»
- Примеры вызова SOAP PartyWS.checkBlacklistByRawID
  - Пример 1
  - Пример 2
- Примеры вызова REST PartyRA.checkBlacklistByRawID

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
party	WParty	да	Информация по контрагенту
includePartyFields	boolean	нет	Возвращать реквизиты найденных карточек ЧС. По умолчанию false.

### Результирующие данные

Имя параметра	Тип	Описание
party	WMatchedParty[]	Информация по найденным карточкам ЧС
party	WRiskParty	Информация по выявленному уровню риска клиента

### Алгоритм

Метод работает по следующему алгоритму:

1. Находит по `sourceSystem` и `rawId` искомого контрагента в базе данных.
2. Отправляет онлайн найденного КА в «Фактор» на кросс-сверку.
3. Находит дубликаты между ним и списками.
4. Сохраняет связь между КА и списком, если таковой еще нет, закрывает имеющиеся связи, если по соответствующим правилам не нашлось совпадений.
5. Возвращает в ответе все совпадения искомого КА со списками, включая актуальные связи, с указанием:
  - a. правила идентификации дубликатов, по которому был найден каждый из дубликатов;
  - b. коэффициента соответствия для правила, по которому контрагенты были идентифицированы как дубликат;
  - c. уровня риска (рассчитывается с учетом настроенных параметров, индивидуально для каждого заказчика).

## Особенности использования

Возвращаемые методом контрагенты отсортированы в порядке убывания коэффициента соответствия.

Если параметр `includePartyFields = true`, то в ответе будут возвращены все реквизиты и атрибуты найденных карточек черных списков.

Для каждой карточки возвращается ее источник (`sourceSystem`) и идентификатор (`rawId`), кроме перечня недействительных паспортов. Для паспортов указывается только **`sourceSystem="INVALID_PASSPORT"`**, без идентификатора.

Помимо уровня риска, возвращается и дата присвоения уровня риска (`amlRiskDate`).

## Логика создания связей «КА — список»

Опирается только на золотого — оригинального или слитого — контрагента и его связи.

При запросе по `rawId` влитого КА и найденном совпадении со списком:

- если на золотом КА нет связи со списком — создаем новую;
- если на золотом КА уже есть связь со списком по правилу, по которому есть совпадение, — новую связь не создаем, в ответе возвращаем имеющуюся;
- если на золотом КА есть связь со списком по правилу, по которому не нашлось актуальных совпадений, — закроем неактуальную, создадим связь по актуальному правилу.

Связи между исходными — влитыми — КА и списками игнорируются. При наличии имеющейся связи по актуальному правилу на исходной карточке все равно создается связь на золотой. Неактуальные связи на исходных карточках не закрываются методом при создании новой связи на золотой.

## Примеры вызова SOAP PartyWS.checkBlacklistByRawID

### Пример 1

Нашлось два совпадения: со списком террористов и списком отказников ЦБ.

Запрос:

```
Запрос checkBlacklistByRawID
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklistByRawID>
      <sourceSystem>AL</sourceSystem>
      <rawId>1</rawId>
      <partyType>PHYSICAL</partyType>
    </checkBlacklistByRawID>
  </soapenv:Body>
</soapenv:Envelope>
```

Запрос аналогичен методу getByRawID

Ответ:

```
Ответ checkBlacklistByRawID
<checkBlacklistByRawIDResponse xmlns="http://hflabs.ru/cdi/soap/open/16_13" xmlns:ns2="http://hflabs.ru/cdi/soap/2_13">
  <party hid="1" rawId="1" sourceSystem="AL" type="PHYSICAL">
    <field name="amlRiskLevel">HIGH</field>
    <field name="amlRiskDate">20.03.2023 12:00:00</field>
  </party>
  <matchedPartyAndRelation>
    <matchedParty>
      <matchRule>2</matchRule>
      <matchScope>99</matchScope>
      <party type="PHYSICAL_LIST" hid="1025" sourceSystem="TERRORIST" rawId="2525">
        <relation type="8052" hashCode="715addf5134396540ea9fe9a584949c8">
          <first type="PHYSICAL" hid="1"/>
          <second type="PHYSICAL_LIST" hid="1025"/>
          <field name="status">UNKNOWN</field>
        </relation>
      </matchedParty>
    </matchedPartyAndRelation>
    <matchedPartyAndRelation>
      <matchedParty>
        <matchRule>5</matchRule>
        <matchScope>96</matchScope>
        <party type="PHYSICAL_LIST" hid="1036" sourceSystem="CB" rawId="2536">
          <relation type="8054" hashCode="42814ed1314d74dda6496778446a49c0">
            <first type="PHYSICAL" hid="1"/>
            <second type="PHYSICAL_LIST" hid="1036"/>
          </relation>
        </party>
      </matchedParty>
    </matchedPartyAndRelation>
  </checkBlacklistByRawIDResponse>
```



```
<field name="status">UNKNOWN</field>
  </relation>
</matchedParty>
</matchedPartyAndRelation>
</checkBlacklistByRawIDResponse>
```

Ответ практически полностью повторяет метод checkBlacklist, только после найденной в списке карточки будет ещё возвращаться созданная с ней связь.

## Пример 2

Для паспорта вернули статус статус «обнаружен в перечне недействительных документов».

Запрос:

```
Запрос checkBlacklistByRawID
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklistByRawID>
      <sourceSystem>BT</sourceSystem>
      <rawId>2</rawId>
      <partyType>PHYSICAL</partyType>
    </checkBlacklistByRawID>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
Ответ checkBlacklistByRawID
<checkBlacklistByRawIDResponse xmlns="http://hflabs.ru/cdi/soap/open/16_13" xmlns:ns2="
http://hflabs.ru/cdi/soap/2_13">
  <party hid="2" rawId="2" sourceSystem="BT" type="PHYSICAL">
    <field name="amlRiskLevel">HIGH</field>
    <field name="amlRiskDate">20.03.2023 12:00:00</field>
  </party>
  <matchedParty>
    <matchRule>51</matchRule>
    <matchScope>100</matchScope>
    <party type="BLACKLIST" sourceSystem="INVALID_PASSPORT"/>
  </matchedParty>
</matchedPartyAndRelation>
</checkBlacklistByRawIDResponse>
```

## Примеры вызова REST PartyRA.checkBlacklistByRawID

Запрос:

```
Запрос checkBlacklistByRawID
{
  "sourceSystem" : "AL",
  "rawId" : "1",
  "partyType" : "PHYSICAL",
```

```
}
  "includePartyFields": "false"
}
```

ОТВЕТ:

ОТВЕТ checkBlacklistByRawID

```
{
  "matchedParty": [
    {
      "matchRule": "2",
      "matchScope": 99,
      "party": {
        "hid": 1025,
        "sourceSystem": "TERRORIST",
        "rawId": "2525",
        "relation": [
          {
            "first": {
              "type": "PHYSICAL",
              "hid": 1,
              "rawId": null,
              "sourceSystem": null
            },
            "second": {
              "type": "PHYSICAL_LIST",
              "hid": 1025,
              "rawId": null,
              "sourceSystem": null
            }
          },
          {
            "field": [
              {
                "value": "UNKNOWN",
                "name": "status"
              }
            ],
            "type": 8052,
            "deleted": null,
            "hashCode": "715addf5134396540ea9fe9a584949c8"
          }
        ],
        "type": "PHYSICAL_LIST"
      }
    },
    {
      "matchRule": "5",
      "matchScope": 96,
      "party": {
        "hid": 1036,
        "sourceSystem": "TERRORIST_CB",
        "rawId": "2536",
        "relation": [
          {
            "first": {
              "type": "PHYSICAL",
              "hid": 1,
              "rawId": null,
              "sourceSystem": null
            },
            "second": {
              "type": "PHYSICAL_LIST",
              "hid": 1036,
              "rawId": null,
              "sourceSystem": null
            }
          },
          {
            "field": [
              {

```

```
    "value": "UNKNOWN",
    "name": "status"
  },
  "type": 8054,
  "deleted": null,
  "hashcode": "42814ed1314d74dda6496778446a49c0"
},
"type": "PHYSICAL_LIST"
}
}
```

# PARTY.CHECKBLACKLISTBYHID

Метод проверяет контрагента по черным спискам по хиду и сразу создает связь между искомым КА и записью ЧС.

Также метод возвращает для паспорта КА статус «обнаружен в перечне недействительных документов» (INVALID), если документ уже был проверен по ПНП и получил такой статус.

- Описание
  - Входные параметры
  - Результирующие данные
  - Алгоритм
  - Особенности использования
- Логика создания связей КА — список
- Примеры вызова SOAP PartyWS.checkBlackListByHid
  - Пример 1
  - Пример 2
- Примеры вызова REST PartyRA.checkBlackListByHid
- 

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
party	WParty	да	Информация по контрагенту
includePartyFields	boolean	нет	Возвращать реквизиты найденных карточек ЧС. По умолчанию false.

### Результирующие данные

Имя параметра	Тип	Описание
party	WMatchedParty[]	Информация по найденным карточкам ЧС
party	WRiskParty	Информация по выявленному уровню риска клиента

### Алгоритм

Метод работает по следующему алгоритму:

1. Находит по хиду искомого контрагента в базе данных.
2. Отправляет онлайн найденного КА в «Фактор» на кросс-сверку.
3. Находит дубликаты между ним и списками.

4. Сохраняет связь между КА и списком, если таковой еще нет, закрывает имеющиеся связи, если по соответствующим правилам не нашлось совпадений.
5. Возвращает в ответе все совпадения искомого КА со списками, включая актуальные связи, с указанием:
  - a. правила идентификации дубликатов, по которому был найден каждый из дубликатов;
  - b. коэффициента соответствия для правила, по которому контрагенты были идентифицированы как дубликаты;
  - c. уровня риска (рассчитывается с учетом настроенных параметров, индивидуально для каждого заказчика).

## **Особенности использования**

Возвращаемые методом контрагенты отсортированы в порядке убывания коэффициента соответствия.

Если параметр *includePartyFields* = **true**, то в ответе будут возвращены все реквизиты и атрибуты найденных карточек черных списков.

Для каждой карточки возвращается ее источник (*sourceSystem*) и идентификатор (*rawId*), кроме перечня недействительных паспортов. Для паспортов указывается только **sourceSystem="INVALID\_PASSPORT"**, без идентификатора.

Помимо уровня риска, возвращается и дата присвоения уровня риска (*amlRiskDate*).

## **Логика создания связей КА — список**

Опирается только на золотого — оригинального или слитого — контрагента и его связи.

При запросе по золотому хиду и найденном совпадении со списком:

- если на КА нет связи со списком — создаем новую;
- если на КА уже есть связь со списком по правилу, по которому есть совпадение, — новую связь не создаем, в ответе возвращаем имеющуюся;
- если на КА есть связь со списком по правилу, по которому не нашлось актуальных совпадений, — закроем неактуальную, создадим связь по актуальному правилу.

Связи между исходными — влитыми — КА и списками игнорируются. При наличии имеющейся связи по актуальному правилу на исходной карточке все равно создается связь на золотой. Неактуальные связи на исходных карточках не закрываются методом при создании новой связи на золотой.

## **Примеры вызова SOAP PartyWS.checkBlackListByHid**

### **Пример 1**

Нашлось два совпадения: со списком террористов и списком отказников ЦБ.

Запрос:

```
Запрос checkBlacklistByHid
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlackListByHid>
      <hid>1</hid>
      <partyType>PHYSICAL</partyType>
    </checkBlackListByHid>
  </soapenv:Body>
</soapenv:Envelope>
```

Запрос аналогичен методу getByHid, так как нужно знать только хид, чтобы найти карточку в системе

Ответ:

```
Ответ checkBlacklistByHid
<checkBlackListByHidResponse xmlns="http://hflabs.ru/cdi/soap/open/16_13" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <party hid="1" rawId="1" sourceSystem="BT" type="PHYSICAL">
    <field name="amlRiskLevel">MEDIUM</field>
    <field name="amlRiskDate">01.10.2011 10:00:00</field>
  </party>
  <matchedParty>
    <matchRule>2</matchRule>
    <matchScope>99</matchScope>
    <party type="PHYSICAL_LIST" hid="1025" sourceSystem="TERRORIST" rawId="2525">
      <relation type="8052" hashCode="715addf5134396540ea9fe9a584949c8">
        <first type="PHYSICAL" hid="1"/>
        <second type="PHYSICAL_LIST" hid="1025"/>
        <field name="status">UNKNOWN</field>
      </relation>
    </party>
  </matchedParty>
  <matchedParty>
    <matchRule>5</matchRule>
    <matchScope>96</matchScope>
    <party type="PHYSICAL_LIST" hid="1036" sourceSystem="CB" rawId="2536">
      <relation type="8054" hashCode="42814ed1314d74dda6496778446a49c0">
        <first type="PHYSICAL" hid="1"/>
        <second type="PHYSICAL_LIST" hid="1036"/>
        <field name="status">UNKNOWN</field>
      </relation>
    </party>
  </matchedParty>
</checkBlackListByHidResponse>
```

Ответ практически полностью повторяет метод checkBlackList, только после найденной в списке карточки будет ещё возвращаться созданная с ней связь, по описанному в алгоритме механизму.

## Пример 2

Для паспорта вернули статус статус «обнаружен в перечне недействительных документов».

Запрос:

```
Запрос checkBlacklistByHid
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <checkBlacklistByHid>
      <hid>1</hid>
      <partyType>PHYSICAL</partyType>
    </checkBlacklistByHid>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
Ответ checkBlacklistByHid
<checkBlacklistByHidResponse xmlns="http://hflabs.ru/cdi/soap/open/16_13" xmlns:ns2="http://hflabs.ru/cdi/soap/2_13">
  <party hid="1" rawId="1" sourceSystem="BT" type="PHYSICAL">
    <field name="amlRiskLevel">MEDIUM</field>
    <field name="amlRiskDate">01.10.2011 10:00:00</field>
  </party>
  <matchedParty>
    <matchRule>51</matchRule>
    <matchScope>100</matchScope>
    <party type="BLACKLIST" sourceSystem="INVALID_PASSPORT"/>
  </matchedParty>
</matchedPartyAndRelation>
</checkBlacklistByHidResponse>
```

## Примеры вызова REST PartyRA.checkBlackListByHid

Запрос:

```
Запрос checkBlacklistbyHid
{
  "hid": "12502201",
  "partyType": "PHYSICAL"
}
```

Ответ:

```
Ответ checkBlacklistByHid
{"matchedParty": [{"matchRule": "5",
"matchScope": 96,
"party": {
"hid": 12523276,
"sourceSystem": "TERRORIST",
```

```
"rawId": "502",
"relation": [ {
  "first": {
    "type": "PHYSICAL",
    "hid": 12502201,
    "rawId": null,
    "sourceSystem": null
  },
  "second": {
    "type": "PHYSICAL_LIST",
    "hid": 12523276,
    "rawId": null,
    "sourceSystem": null
  },
  "field": [ {
    "value": "UNKNOWN",
    "name": "status"
  } ],
  "type": 8055,
  "deleted": null,
  "hashCode": "98f15581d1b38f202072ab6096159494"
}],
"type": "PHYSICAL_LIST"
}
}]}
```

В случае, когда контрагент не найден в черном списке, в ответе вернется:

**Ответ checkBlacklistByHid**

**<Empty JSON content>**



# PARTY.CLOSE

Метод позволяет закрыть контрагента по его идентификатору в Едином Клиенте (HID) или по идентификатору в исходной системе (RawId).

- Входные параметры
- Результирующие данные
- Особенности использования
- Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Клиент успешно закрыт, передан hid
  - Пример 2. Клиент успешно закрыт, передан код системы и идентификатор в ней
- Примеры вызова PartyRA (JSON)
  - Пример 1. Клиент успешно закрыт

## Входные параметры

Имя параметра	Тип	Обязательный?	Описание
hid	long	нет	Уникальный идентификатор контрагента в Едином клиенте (HID).
sourceSystem	string	нет	Код исходной системы.
rawId	string	нет	Идентификатора клиента в исходной системе.
partyType	Тип контрагента	да	Тип контрагента

## Результирующие данные

Имя параметра	Тип	Описание
party	WParty	Информация по контрагенту.

## Особенности использования

Для закрытия контрагента достаточно передать одно из полей **HID** или **sourceSystem + rawId**.

Если исходный идентификатор (**sourceSystem + rawId**) противоречит идентификатору HID, то выбор объекта для закрытия осуществляется по HID.

Для записей, участвовавших в слиянии или полученных в результате слияния:

- При вызове метода с идентификатором НID итогового (полученного в результате слияния) контрагента система закроет объединенного контрагента и каждого из его исходных контрагентов.
- При вызове метода с идентификатором НID исходного контрагента система закроет только этого исходного контрагента, предварительно "вылив" (операция разъединения) его из объединенной записи. Если у слитого контрагента было всего две исходные записи, будет выполнено полное разливание для данного слитого контрагента.

## ***Исключительные ситуации***

Если контрагента с указанным идентификатором в системе не существует или он был закрыт ранее, система вернет ошибку.

## ***Примеры вызова PartyWS (SOAP)***

### ***Пример 1. Клиент успешно закрыт, передан hid***

Запрос:

```


Запрос

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <close>
      <hid>1024</hid>
      <partyType>PHYSICAL</partyType>
    </close>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <closeResponse xmlns="http://hflabs.ru/cdi/soap/2_13"/>
  </soap:Body>
</soap:Envelope>
```

### ***Пример 2. Клиент успешно закрыт, передан код системы и идентификатор в ней***

Запрос:

```


Запрос

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
```

```
<soapenv:Header/>
<soapenv:Body>
  <close>
    <sourceSystem>AL</sourceSystem>
    <rawId>1</rawId>
    <partyType>PHYSICAL</partyType>
  </close>
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <closeResponse xmlns="http://hflabs.ru/cdi/soap/2_13"/>
  </soap:Body>
</soap:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Клиент успешно закрыт

Запрос:

```
{ "hid" : "1024" }
```

ОТВЕТ:

```
{}
```

# PARTY.CLOSEATTRIBUTE

Закрывает атрибут клиента по его hid.

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
  - Исключительные ситуации
- Пример вызова PartyWS (SOAP)
- Пример вызова PartyRA (JSON)

## Описание

Метод позволяет закрыть (удалить) у клиента атрибут с указанным идентификатором.

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
partyType	Тип контрагента	Да	Тип контрагента
attributeType	Тип атрибута	Да	Тип атрибута
attributeHid	Целое	Да	Уникальный идентификатор атрибута в CDI

### Результирующие данные

При успешном закрытии атрибута система вернет пустой ответ.

### Особенности использования

Для атрибутов, участвовавших в слиянии и полученных в результате слияния:

- при вызове метода с идентификатором HID итогового (полученного в результате слияния) атрибута система закроет данный атрибут и каждого из его исходных атрибутов.
- при вызове метода с идентификатором HID исходного атрибута система вернет ошибку (т. е. "частичное" закрытие объединенных атрибутов не поддерживается).

### Исключительные ситуации

Если атрибут с указанным HID и типом отсутствует или уже закрыт, система вернет ошибку.

## Пример вызова PartyWS (SOAP)

Запрос:

```
Запрос  
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <closeAttribute>  
      <partyType>PHYSICAL</partyType>  
      <attributeType>PHONE</attributeType>  
      <attributeHid>1</attributeHid>  
    </closeAttribute>  
  </soapenv:Body>  
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <closeAttributeResponse/>  
  </soapenv:Body>  
</soapenv:Envelope>
```

## Пример вызова PartyRA (JSON)

Запрос:

```
{ "partyType" : "PHYSICAL",  
  "attributeType" : "PHONE",  
  "attributeHid" : "1" }
```

ОТВЕТ:

```
{ }
```

# PARTY.CLOSEATTRIBUTEBYRAWID

Метод позволяет закрыть атрибут клиента по его rawId.

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
  - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Контрагент указан по HID
  - Пример 2. Контрагент указан по связке rawId + SourceSystem
- Пример вызова PartyRA (JSON)

## Описание

Метод позволяет закрыть (удалить) у клиента атрибут с указанным идентификатором в исходной системе.

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
partyIdentifier	WPartyType	Да	Контрагент
attributeType	Тип атрибута	Да	Тип атрибута
attributeRawId	String	Да	Уникальный идентификатор атрибута в исходной системе

### Результирующие данные

При успешном закрытии атрибута система вернет пустой ответ.

### Особенности использования

Метод поддерживает только закрытие атрибутов в исходных контрагентах (**MERGED\_STATUS = 0 или 1**).

Не модифицирует staging.

### Исключительные ситуации

Система вернет ошибку в следующих ситуациях:

- Контрагент с указанными параметрами отсутствует, закрыт или является результатом слияния.

- Атрибут с указанным rawId и типом отсутствует или уже закрыт.

## Примеры вызова PartyWS (SOAP)

### Пример 1. Контрагент указан по HID

Запрос:

Запрос
<pre>&lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;closeAttributeByRawId&gt;       &lt;partyIdentifier type="PHYSICAL" hid="1024"/&gt;       &lt;attributeType&gt;PHONE&lt;/attributeType&gt;       &lt;attributeRawId&gt;1&lt;/attributeRawId&gt;     &lt;/closeAttributeByRawId&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt;</pre>

Ответ:

<pre>&lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;closeAttributeByRawIdResponse/&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt;</pre>
--

### Пример 2. Контрагент указан по связке rawId + SourceSystem

Запрос:

Запрос
<pre>&lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;closeAttributeByRawId&gt;       &lt;partyIdentifier type="PHYSICAL" sourceSystem="AL" rawId="1"/&gt;       &lt;attributeType&gt;PHONE&lt;/attributeType&gt;       &lt;attributeRawId&gt;PH.1&lt;/attributeRawId&gt;     &lt;/closeAttributeByRawId&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt;</pre>

Ответ:

<pre>&lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;</pre>
---

```
<closeAttributeByRawIdResponse/>
</soapenv:Body>
</soapenv:Envelope>
```

## ***Пример вызова PartyRA (JSON)***

Запрос:

```
{ "partyIdentifier" : {
  "sourceSystem" : "BT",
  "rawId" : "1",
  "type":"PHYSICAL"},
  "attributeType" : "PHONE",
  "attributeRawId" : "1" }
```

Ответ:

```
{} 
```



# PARTY.GETREFERENCE

Метод позволяет получить все значения из справочника [Классификаторов](#) по выбранному классификатору.

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
  - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Значения классификатора получены
  - Пример 2. Для указанного типа классификатора не найдено доступных значений
  - Пример 3. Значения классификатора получены с учетом неактивных значений
  - Пример 4. Значения классификатора получены с учетом типа контрагента
- Примеры вызова PartyRA (JSON)
  - Пример 1. Значения классификатора получены

## Описание

Данный метод позволяет получить список доступных значений для следующих Классификаторов:

- Тип телефона;
- Тип адреса;
- Тип адреса электронной почты;
- Тип ДУЛ для физического лица;
- Тип лицензии;
- Тип документа для юридического лица;
- Тип документа для индивидуального предпринимателя.

## Входные параметры

Имя параметра	Тип	Обязательный?	Описание
referenceType	WReferenceType	да	Название классификатора в ЕК.
partyType	WPartyType	нет	Тип контрагента.
onlyActual	boolean	нет	Признак актуальности значений классификатора. По умолчанию имеет значение "true".

## Результирующие данные

Имя параметра	Тип	Описание
reference	WReference	Значение классификатора

## Особенности использования

Если в запросе не указан тип контрагента, то система вернет все доступные для данного классификатора значения.

Если в запросе указан тип контрагента, то система вернет только такие значения классификатора, у которых:

- тип контрагента соответствует типу, указанному в запросе;
- тип контрагента в классификаторе не указан.

Если признак актуальности значений классификатора (onlyActual) в запросе не указан или равен "true", то система вернет только активные значения классификатора.

Если признак актуальности значений классификатора (onlyActual) в запросе равен "false", то система вернет как активные, так и уже закрытые ранее значения классификатора.

## Исключительные ситуации

Если по указанному названию классификатора не найдено значений, то система вернет пустое сообщение.

Если контрагента с указанным типом в системе не существует, то система вернет ошибку.

Если по названию классификатора найдены доступные значения, но все они имеют непустой тип контрагента, отличный от указанного в запросе, то система вернет сообщение об ошибке.

Если в запросе не указан признак актуальности значений классификатора (onlyActual) или он равен "true", а в системе все значения данного классификатора закрыты, то есть имеют признак актуальности, равный "false", то система вернет сообщение об ошибке.

## Примеры вызова PartyWS (SOAP)

### Пример 1. Значения классификатора получены

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getReference>
      <referenceType>ADDRESS_TYPE</referenceType>
    </getReference>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</getReference>
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getReferenceResponse>
      <reference type="ADDRESS_TYPE">
        <entry code="OTHER">
          <actual>true</actual>
          <ordinal>0</ordinal>
          <label>Иной</label>
          <description>Иное значение</description>
        </entry>
        <entry code="HOME">
          <partyType>PHYSICAL,INDIVIDUAL</partyType>
          <actual>true</actual>
          <ordinal>1</ordinal>
          <label>Домашний</label>
          <description>Домашний адрес</description>
        </entry>
        <entry code="LOCATION">
          <partyType>LEGAL</partyType>
          <actual>true</actual>
          <ordinal>7</ordinal>
          <label>Фактический</label>
          <description>Фактический адрес</description>
        </entry>
        <entry code="LEGAL">
          <partyType>LEGAL</partyType>
          <actual>true</actual>
          <ordinal>8</ordinal>
          <label>Юридический</label>
          <description>Юридический адрес</description>
        </entry>
      </reference>
    </getReferenceResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

*Пример 2. Для указанного типа классификатора не найдено доступных значений*

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getReference>
      <referenceType>ADDRESS_LEGAL</referenceType>
    </getReference>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getReferenceResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <reference type="ADDRESS_LEGAL"/>
    </getReferenceResponse>
  </soap:Body>
</soap:Envelope>
```

### Пример 3. Значения классификатора получены с учетом неактивных значений

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getReference>
      <referenceType>ADDRESS_TYPE</referenceType>
      <onlyActual>>false</onlyActual>
    </getReference>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getReferenceResponse>
      <reference type="ADDRESS_TYPE">
        <entry code="OTHER">
          <actual>>true</actual>
          <ordinal>0</ordinal>
          <label>Иной</label>
          <description>Иное значение</description>
        </entry>
        <entry code="HOME">
          <partyType>PHYSICAL,INDIVIDUAL</partyType>
          <actual>>true</actual>
          <ordinal>1</ordinal>
          <label>Домашний</label>
          <description>Домашний адрес</description>
        </entry>
        <entry code="LOCATION">
          <partyType>LEGAL</partyType>
          <actual>>true</actual>
          <ordinal>7</ordinal>
          <label>Фактический</label>
          <description>Фактический адрес</description>
        </entry>
        <entry code="LEGAL">
          <partyType>LEGAL</partyType>
        </entry>
      </reference>
    </getReferenceResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        <actual>true</actual>
        <ordinal>8</ordinal>
        <label>Юридический</label>
        <description>Юридический адрес</description>
    </entry>
    <entry code="LEGAL">
        <partyType>PHYSICAL</partyType>
        <actual>false</actual>
        <ordinal>9</ordinal>
        <label>Тестовый</label>
    </entry>
</reference>
</getReferenceResponse>
</soapenv:Body>
</soapenv:Envelope>

```

#### Пример 4. Значения классификатора получены с учетом типа контрагента

Запрос:

```

                Запрос
    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
    http://hflabs.ru/cdi/soap/2_13">
        <soapenv:Header/>
        <soapenv:Body>
            <getReference>
                <referenceType>ADDRESS_TYPE</referenceType>
                <partyType>LEGAL</partyType>
            </getReference>
        </soapenv:Body>
    </soapenv:Envelope>

```

Ответ:

```

    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
    http://hflabs.ru/cdi/soap/2_13">
        <soapenv:Header/>
        <soapenv:Body>
            <getReferenceResponse>
                <reference type="ADDRESS_TYPE">
                    <entry code="OTHER">
                        <actual>true</actual>
                        <ordinal>0</ordinal>
                        <label>Иной</label>
                        <description>Иное значение</description>
                    </entry>
                    <entry code="LOCATION">
                        <partyType>LEGAL</partyType>
                        <actual>true</actual>
                        <ordinal>7</ordinal>
                        <label>Фактический</label>
                        <description>Фактический адрес</description>
                    </entry>
                    <entry code="LEGAL">
                        <partyType>LEGAL</partyType>
                        <actual>true</actual>
                        <ordinal>8</ordinal>
                        <label>Юридический</label>
                    </entry>
                </reference>
            </getReferenceResponse>
        </soapenv:Body>
    </soapenv:Envelope>

```

```
<description>Юридический адрес</description>
</entry>
</reference>
</getReferenceResponse>
</soapenv:Body>
</soapenv:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Значения классификатора получены

Запрос:

```
{ "referenceType" : "EMAIL_TYPE" }
```

Ответ:

```
{ "reference": {
  "entry": [
    {
      "partyType": null,
      "actual": true,
      "ordinal": 0,
      "label": "Иной",
      "description": null,
      "code": "OTHER"
    },
    {
      "partyType": null,
      "actual": true,
      "ordinal": 1,
      "label": "Рабочий",
      "description": null,
      "code": "WORK"
    },
    {
      "partyType": null,
      "actual": true,
      "ordinal": 2,
      "label": "Домашний",
      "description": null,
      "code": "HOME"
    }
  ],
  "type": "EMAIL_TYPE"
}}
```

# PARTY.SAVERELATIONS

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
    - Создание новой связи
    - Закрытие существующей связи
    - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Создание связи между физическим и юридическим лицом
  - Пример 2. Создание нескольких связей
  - Пример 3. Закрытие связи между физическим и юридическим лицом
- Примеры вызова PartyRA (JSON)
  - Пример 1. Создание связи между физическими лицами

Метод позволяет создавать и закрывать взаимосвязи между объектами.

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный	Описание
relation[]	WRelation	Да	Взаимосвязи. Параметр <i>deleted</i> по умолчанию принимает значение <b>FALSE</b> .

### Результирующие данные

При успешном завершении работы метод вернет пустой ответ.

### Особенности использования

Для каждой из связей, указанных в запросе, метод работает по следующему алгоритму

1. Найти в системе указанные во входных параметрах объекты.
2. Создать или закрыть между ними указанную взаимосвязь.

Связываемые объекты в **WRelation** должны быть указаны в том же порядке, что и соответствующие им сущности в описании типов взаимосвязей. То есть, если контрагент А является поручителем контрагента Б, то, в соответствии с МД003. Типы взаимосвязей между

контрагентами, параметры контрагента А должны быть указаны в **//relation@first**, а параметры контрагента Б — в **//relation@second**. Связи между одними и теми же объектами в системе, указанными в разном порядке, считаются двумя разными взаимосвязями.

В запросе для каждой из сторон могут быть переданы либо НID объекта в Едином клиенте, либо исходные система и идентификатор объекта. Причем для одной стороны может быть передан только идентификатор объекта в Едином Клиенте, а для другой исходные идентификатор и система.

Если для одной из сторон связи указаны и НID в Едином клиенте, и исходные система и идентификатор, то поиск объекта в системе осуществляется по НID.

Метод выполняется в одной транзакции, то есть в случае ошибки при работе с любой из указанных в запросе связей никакие изменения не сохраняются.

## Создание новой связи

Для каждой из связей, указанных в запросе с параметром **deleted = false** (умолчательное значение):

1. если указанной связи между заданными объектами не существует, то система создаст между ними новую взаимосвязь;
2. если указанная связь между заданными объектами уже существует — ничего не делать.

## Закрытие существующей связи

Для каждой из связей, отмеченных как закрытая (**deleted = true**):

1. если указанная взаимосвязь найдена — закрыть ее;
2. если указанная взаимосвязь не существует или закрыта — ничего не делать.

При слиянии нескольких исходных записей в одну золотую связи исходных пробрасываются на золотую.

Закрытие связей золотого автоматически закрывает и соответствующие связи исходных.

## Исключительные ситуации

Система вернет ошибку в любом из следующих случаев:

- Если по указанным входным параметрам не найден хотя бы один объект.
- Если хотя бы один объект, указанный во входных параметрах, закрыт.
- Если хотя бы один объект, указанный во входных параметрах, полностью разъединен.
- Если указан неизвестный тип связи.
- Если в процессе создания связи произошла ошибка.
- Если в запросе:

тип первого объекта отличен от типа объекта А для заданного типа связи или (и)

тип второго объекта отличен от типа объекта Б для заданного типа связи.



Во всех перечисленных исключительных ситуациях система завершит работу без сохранения изменений (создание или закрытие связей) для всех связей, указанных в запросе.

## Примеры вызова PartyWS (SOAP)

### Пример 1. Создание связи между физическим и юридическим лицом

Запрос:

```


Запрос

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveRelations>
      <relation type="16">
        <first type="PHYSICAL" sourceSystem="AL" rawId="1"/>
        <second type="LEGAL" sourceSystem="BT" rawId="3"/>
      </relation>
    </saveRelations>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveRelationResponse/>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 2. Создание нескольких связей

В приведенном запросе передаются сразу три связи, причем:

- в первой связи указаны только идентификаторы объектов в Едином Клиенте;
- во второй для одной стороны связи указан идентификатор в Едином Клиенте, а для другой — исходные система и идентификатор;
- для третьей связи заданы только исходные идентификаторы и системы.

Запрос:

```


Запрос

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveRelations>
```

```

<relation type="16">
  <first type="PHYSICAL" hid="11025"/>
  <second type="LEGAL" hid="2024"/>
</relation>
<relation type="16">
  <first type="PHYSICAL" hid="1024"/>
  <second type="LEGAL" sourceSystem="BT" rawId="3"/>
</relation>
<relation type="16">
  <first type="PHYSICAL" sourceSystem="AL" rawId="1"/>
  <second type="LEGAL" sourceSystem="BT" rawId="3"/>
</relation>
</saveRelations>
</soapenv:Body>
</soapenv:Envelope>

```

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveRelationResponse/>
  </soapenv:Body>
</soapenv:Envelope>

```

### Пример 3. Заккрытие связи между физическим и юридическим лицом

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;saveRelations&gt;       &lt;relation type="16" deleted="true"&gt;         &lt;first type="PHYSICAL" hid="11025"/&gt;         &lt;second type="LEGAL" hid="2024"/&gt;       &lt;/relation&gt;     &lt;/saveRelations&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <saveRelationResponse/>
  </soapenv:Body>
</soapenv:Envelope>

```

## Примеры вызова PartyRA (JSON)

### Пример 1. Создание связи между физическими лицами

Запрос:

```
{  
  "relation": [{  
    "first": {  
      "type": "PHYSICAL",  
      "hid": 1065  
    },  
    "second": {  
      "type": "PHYSICAL",  
      "hid": 1066  
    },  
    "type": 2,  
    "deleted": null  
  }]  
}
```

Ответ:

```
{
```

# PARTY.GETRELATIONS

Метод позволяет получить **взаимосвязи** объекта с другими объектами по идентификатору в «Едином клиенте». Возвращает связи между золотыми контрагентами.

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
  - Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Найден клиент со связями по hid
  - Пример 2. Найден клиент со связями заданного типа
  - Пример 3. Найден клиент со связями по RawId
  - Пример 4. Для указанного объекта связи не найдены
- Примеры вызова PartyRA (JSON)
  - Пример 1. Найден клиент со связями по RawId

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
essence	Связанный объект	Да	Взаимосвязи.
relationType	Код связи	Нет	Тип взаимосвязи между объектами

### Результирующие данные

Имя параметра	Тип	Описание
relation[]	WRelation	Взаимосвязи

### Особенности использования

Система вернет для объекта, указанного в запросе, все **взаимосвязи**, в которых он является стороной А (**//relation@first**) или стороной Б (**//relation@second**).

В ответе для взаимосвязей **//relation** будет указана только противоположная запрашиваемому контрагенту сторона взаимосвязи. При этом для связанного объекта в ответе будет указан только его тип и уникальный идентификатор в CDI (HID). А если связь между объектами подтверждаемая, то в ответе вернётся также статус подтверждения связи (пример 1).

В ответе для взаимосвязей **//party/relation** будет указана только противоположная запрашиваемому контрагенту сторона взаимосвязи.

Если в запросе указан параметр *relationType*, то система вернет только те взаимосвязи, тип которых соответствует указанному в запросе.

Если исходный идентификатор (**sourceSystem + rawId**) противоречит идентификатору НІД, то поиск объекта в системе осуществляется по НІД.

Если указанный контрагент связан с контрагентом, который был объединен, то система вернет связь с объединенным контрагентом.

Если контрагент был объединен в результате слияния, то при вызове метода с идентификатором НІД исходного или объединенного контрагента система вернет:

- для связей с черными списками — только связи объединенного контрагента;
- для остальных типов связей — связи объединенного контрагента и связи всех его исходных контрагентов (без дублирования одинаковых).

### *Исключительные ситуации*

Если по заданному объекту не найдено связей (с заданным типом), то система вернет пустой ответ.

Если указанный контрагент был полностью разъединен, то система вернет ошибку.

Если указанный контрагент был закрыт, то система вернет ошибку.

## **Примеры вызова PartyWS (SOAP)**

### *Пример 1. Найден клиент со связями по hid*

Для связи 500 также возвращен статус, т.к. связь подтверждаемая

Запрос:

Запрос
<pre>&lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;getRelations&gt;       &lt;essence type="PHYSICAL" hid="1024"/&gt;     &lt;/getRelations&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt;</pre>

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getRelationsResponse>
      <relation type="1" hashCode="518fbfc8e87918dff5feb24bd78982d6">
        <first type="PHYSICAL" hid="2024"/>
      </relation>
      <relation type="500" hashCode="dcbf76d60b76e3b767c060c4eff9d868">
        <first type="PHYSICAL" hid="2025"/>
        <field name="status">UNKNOWN</field>
      </relation>
      <relation type="11" hashCode="12bf76d60b76e3b767c060c4eff9d860">
        <second type="LEGAL" hid="3024"/>
      </relation>
    </getRelationsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Пример 2. Найден клиент со связями заданного типа

Запрос:

Запрос
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns=" http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;getRelations&gt;       &lt;essence type="PHYSICAL" hid="1024"/&gt;       &lt;relationType&gt;1&lt;/relationType&gt;     &lt;/getRelations&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Ответ:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getRelationsResponse>
      <relation type="1" hashCode="518fbfc8e87918dff5feb24bd78982d6">
        <first type="PHYSICAL" hid="2024"/>
      </relation>
      <relation type="1" hashCode="dcbf76d60b76e3b767c060c4eff9d868">
        <first type="PHYSICAL" hid="2025"/>
      </relation>
    </getRelationsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Пример 3. Найден клиент со связями по RawId

Запрос:

#### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getRelations>
      <essence type="PHYSICAL" sourceSystem="BT" rawId="1"/>
    </getRelations>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getRelationsResponse>
      <relation type="1" hashCode="dcbf76d60b76e3b767c060c4eff9d868">
        <first type="PHYSICAL" hid="2024"/>
      </relation>
      <relation type="11" hashCode="518fbfc8e87918dff5feb24bd78982d6">
        <second type="LEGAL" hid="3024"/>
      </relation>
    </getRelationsResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

*Пример 4. Для указанного объекта связи не найдены*

Запрос:

#### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getRelations>
      <essence type="PHYSICAL" hid="5024"/>
    </getRelations>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getRelationsResponse/>
  </soapenv:Body>
</soapenv:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1. Найден клиент со связями по RawId

Запрос:

```
{  
  "essence": {  
    "sourceSystem": "BT",  
    "rawId": "1",  
    "type": "PHYSICAL"}  
}
```

Ответ:

```
{  
  "relation": [{  
    "first": null,  
    "second": {  
      "type": "PHYSICAL",  
      "hid": 1066,  
      "rawId": null,  
      "sourceSystem": null  
    },  
    "type": 1,  
    "hashCode": "518fbfc8e87918dff5feb24bd78982d6",  
    "deleted": null  
  }]  
}
```



# PARTY.GETSOURCEATTRIBUTES

Метод позволяет получить по идентификатору атрибута в Едином клиенте информацию об атрибутах, из которых он был составлен .

- Описание
  - Входные параметры
  - Результирующие данные
  - Особенности использования
- Исключительные ситуации
- Примеры вызова PartyWS (SOAP)
  - Пример 1. Атрибут слитого контрагента составлен из двух исходных
  - Пример 2. Запрос по исходному контрагенту без атрибутов-дубликатов
  - Пример 3. Атрибута с указанным идентификатором не существует

## Описание

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
partyType	Тип контрагента	да	Тип контрагента
attributeType	Тип атрибута	да	Тип атрибута
attributeHid	long	да	Уникальный идентификатор атрибута в Едином клиенте (HID).

### Результирующие данные

Имя параметра	Тип	Описание
attribute	WAttribute	Информация по атрибуту.

### Особенности использования

Если для атрибута, по которому выполняется запрос, присутствует hidden-копия (схлопнутый атрибут), для неё также будут возвращены исходные атрибуты.

### Исключительные ситуации

Если атрибута с указанным идентификатором в системе не существует, то система вернет ошибку.

Если атрибута с указанным типом в системе не существует, то система вернет ошибку.

Если атрибут с указанным типом и идентификатором существует в системе, но принадлежит контрагенту с *partyType*, не равным переданному в запросе, система вернет ошибку.

Если указанного типа контрагента в системе не существует, то система вернет ошибку.

Если указанный атрибут был закрыт, то система вернет ошибку.

Если переданный в запросе *attributeHid* принадлежит контрагенту, не являющемуся результирующим, то система вернет ошибку.

## Примеры вызова PartyWS (SOAP)

### Пример 1. Атрибут слитого контрагента составлен из двух исходных

Запрос:

```
Запрос  
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <getSourceAttributes>  
      <partyType>PHYSICAL</partyType>  
      <attributeType>PHONE</attributeType>  
      <attributeHid>1026</attributeHid>  
    </getSourceAttributes>  
  </soapenv:Body>  
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">  
  <soap:Body>  
    <getSourceAttributesResponse xmlns="http://hflabs.ru/cdi/soap/2_13">  
      <attribute type="PHONE" hid="1024" rawId="PHONE.1024">  
        <field name="qualityCode">GOOD</field>  
        <field name="type">OTHER</field>  
        <field name="operator"/>  
        <field name="timezone">UTC+4</field>  
        <field name="countryCode">7</field>  
        <field name="cityCode">495</field>  
        <field name="number">1234567</field>  
        <field name="extension"/>  
        <field name="rawSource">4951234567</field>  
        <field name="primary">>false</field>  
        <field name="additionalInfo"/>  
        <field name="actualityDate">13.04.2002</field>  
        <field name="author">AL:1</field>  
      </attribute>  
      <attribute type="PHONE" hid="1025" rawId="PHONE.1025">  
        <field name="qualityCode">GOOD</field>  
        <field name="type">OTHER</field>  
        <field name="operator"/>  
        <field name="timezone">UTC+4</field>  
        <field name="countryCode">7</field>  
      </attribute>  
    </getSourceAttributesResponse>  
  </soap:Body>  
</soap:Envelope>
```

```

<field name="cityCode">495</field>
<field name="number">1234567</field>
<field name="extension"/>
<field name="rawSource">4951234567</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="actualityDate">10.04.2002</field>
<field name="author">AL:2</field>
</attribute>
</getSourceAttributesResponse>
</soap:Body>
</soap:Envelope>

```

## Пример 2. Запрос по исходному контрагенту без атрибутов-дубликатов

Запрос:

**Запрос**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getSourceAttributes>
      <partyType>PHYSICAL</partyType>
      <attributeType>PHONE</attributeType>
      <attributeHid>1024</attributeHid>
    </getSourceAttributes>
  </soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getSourceAttributesResponse xmlns="http://hflabs.ru/cdi/soap/2_13">
      <attribute type="PHONE" hid="1024" rawId="PHONE.1024">
        <field name="qualityCode">GOOD</field>
        <field name="type">OTHER</field>
        <field name="operator"/>
        <field name="timezone">UTC+4</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">1234567</field>
        <field name="extension"/>
        <field name="rawSource">4951234567</field>
        <field name="primary">>false</field>
        <field name="additionalInfo"/>
        <field name="actualityDate">13.04.2002</field>
        <field name="author">AL:1</field>
      </attribute>
    </getSourceAttributesResponse>
  </soap:Body>
</soap:Envelope>

```

## Пример 3. Атрибута с указанным идентификатором не существует

Запрос:

### Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <getSourceAttributes>
      <partyType>PHYSICAL</partyType>
      <attributeType>PHONE</attributeType>
      <attributeHid>1027</attributeHid>
    </getSourceAttributes>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:PARTY_NOT_FOUND</faultcode>
      <faultstring>No 'Phone' found with HID 1027 on the date "</faultstring>
      <detail>
        <errorMessage xmlns="http://hflabs.ru/cdi/soap/2_13">
          <errorType>IncorrectDateException</errorType>
          <errorMessage>Error getSourceAttributes for PHONE (HID = 1027) with partyType
PHYSICAL: No 'Phone' found with HID 1027 on the date "</errorMessage>
        </errorMessage>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

# PARTY.MERGESOURCE

❗ С релиза 18.1 данный метод не используется.

Устарел, больше не поддерживается.

Метод позволяет создать нового исходного контрагента и принудительно объединить его с указанным результирующим контрагентом.

## Примеры вызова PartyWS (SOAP)

### Пример 1

Запрос:

```
Запрос
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <mergeSource>
      <party type="PHYSICAL" sourceSystem="GM" rawId="3"/>
      <mergeTo type="PHYSICAL" hid="1026"/>
    </mergeSource>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <soapenv:Header/>
  <soapenv:Body>
    <mergeSourceResponse>
      <party type="PHYSICAL" hid="1026">
        <field name="actualityDate">22.02.2012</field>
        <field name="surname">КУРКО</field>
        <field name="name">ОЛЕГ</field>
        <field name="patronymic">ИГОРЕВИЧ</field>
        ...
        <source hid="1024" sourceSystem="AL" rawId="1"/>
        <source hid="1025" sourceSystem="BT" rawId="2"/>
        <source hid="1027" sourceSystem="GM" rawId="3"/>
      </party>
    </mergeSourceResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Примеры вызова PartyRA (JSON)

### Пример 1

Запрос:

```
{
  "party" : {
    "sourceSystem" : "GM",
    "rawId" : "3",
    "type" : "PHYSICAL"
  },
  "mergeTo" : {
    "type": "PHYSICAL",
    "hid": "1026"
  }
}
```

Ответ:

```
{
  "party": {
    "field": [
      {
        "value": "22.02.2012",
        "name": "actualityDate"
      },
      {
        "value": "КУРКО",
        "name": "surname"
      },
      {
        "value": "ОЛЕГ",
        "name": "name"
      },
      {
        "value": "ИГОРЕВИЧ",
        "name": "patronymic"
      },
      ...
    ],
    "hid": 1026,
    "sourceSystem": null,
    "rawId": null,
    "source": [
      {
        "hid": 1024,
        "sourceSystem": "AL",
        "rawId": "1"
      },
      {
        "hid": 1025,
        "sourceSystem": "BT",
        "rawId": "2"
      },
      {

```

```
    "hid": 1027,  
    "sourceSystem": "GM",  
    "rawId": "3"  
  },  
  ],  
  "type": "PHYSICAL"  
}
```

# PARTYWS И PARTYRA (WSDL, WADL И СХЕМЫ)

Файл	Описание
<a href="#">partyra.wadl</a>	WADL сервиса PartyRA
<a href="#">partyws.wsdl</a>	WSDL сервиса PartyWS
<a href="#">partyws-model.xsd</a>	Модель данных контрагентов в системе
<a href="#">partyws-msg.xsd</a>	Модель данных сервисов PartyWS и PartyRA
<a href="#">common-msg.xsd</a>	Модель данных базовых сообщений



# TASKWS И TASKRA

Интерфейс **TaskWS** предоставляет методы, позволяющие управлять [задачами обработки данных](#). Взаимодействие осуществляется по протоколу SOAP.

Аналогичные методы предоставляет интерфейс **TaskRA**, при этом взаимодействие осуществляется с помощью REST-сервисов.

Интерфейсы включают следующие методы:

- [Task.execute](#) — Метод позволяет запустить задачу с указанием ее параметров.
- [Task.status](#) — Метод позволяет получить статус запущенной задачи по ее идентификатору.
- [TaskWS и TaskRA \(WSDL и схемы\)](#)

# TASK.EXECUTE

Метод позволяет запустить задачу с указанием ее параметров.

- Входные параметры
- Результирующие данные
- Особенности использования
- Примеры вызова TaskWS (SOAP)
  - Пример 1. Задача успешно запущена
  - Пример 2. Запуск задачи с несколькими параметрами
- Примеры вызова TaskRA (JSON)
  - Пример 1. Задача успешно запущена с указанными параметрами
  - Пример 2. Запуск задачи с несколькими параметрами

## Входные параметры

Имя параметра	Тип	Обязательный?	Описание
name	string	да	Название задачи.
parameters	WTaskParameter	нет	Параметры выполнения задачи.

## Результирующие данные

Имя параметра	Тип	Описание
id	string	Идентификатор запущенной задачи.

## Особенности использования

Для запуска задачи достаточно передать ее название. Если параметры не указаны, то применяются параметры по умолчанию, либо сохраненные ранее.

## Примеры вызова TaskWS (SOAP)

### Пример 1. Задача успешно запущена

Запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/task/15_3">  
<soapenv:Header/>
```

```
<soapenv:Body>
  <executeTaskRequest>
    <name>findDuplicates</name>
  </executeTaskRequest>
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <executeTaskResponse xmlns:ns2="http://hflabs.ru/cdi/soap/2_13" xmlns="http://hflabs.ru/cdi/task/15_3">
      <id>33729439-6ee9-49d4-a3ba-21c3cf21e463</id>
    </executeTaskResponse>
  </soap:Body>
</soap:Envelope>
```

## Пример 2. Запуск задачи с несколькими параметрами

Запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/task/15_3">
  <soapenv:Header/>
  <soapenv:Body>
    <executeTaskRequest>
      <name>diagnosticLogs</name>
      <parameters name="sendEmail">
        <value>true</value>
      </parameters>
      <parameters name="sendTo">
        <value>support_cdi@hflabs.ru</value>
      </parameters>
      <parameters name="profiles">
        <value>logs</value>
      </parameters>
      <parameters name="params">
        <value>lowerBound=2015-08-08;upperBound=2015-08-11</value>
      </parameters>
    </executeTaskRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <executeTaskResponse xmlns:ns2="http://hflabs.ru/cdi/soap/2_13" xmlns="http://hflabs.ru/cdi/task/15_3">
      <id>33729439-6ee9-49d4-a3ba-21c3cf21e463</id>
    </executeTaskResponse>
  </soap:Body>
</soap:Envelope>
```

## Примеры вызова TaskRA (JSON)

### Пример 1. Задача успешно запущена с указанными параметрами

Запрос:

```
{
  "name" : "allEnginesRebuildByName",
  "parameters": [
    {
      "name": "enginesToRebuild",
      "value": "GRAPH, PHYSICAL"
    }
  ]
}
```

Ответ:

```
{
  "id": "a4870c9b-5b67-4256-b53c-7653439c907e"
}
```

### Пример 2. Запуск задачи с несколькими параметрами

Запрос:

```
{
  "name" : "diagnosticLogs",
  "parameters": [
    {
      "name": "sendEmail",
      "value": "true"
    },
    {
      "name": "sendTo",
      "value": "support_cdi@hflabs.ru"
    },
    {
      "name": "profiles",
      "value": "logs"
    },
    {
      "name": "params",
      "value": "lowerBound=2015-08-08;upperBound=2015-08-11"
    }
  ]
}
```

Ответ:

```
{
  "id": "3c644526-6b8f-4676-869f-9a39b0be2f29"
}
```

# TASK.STATUS

Метод позволяет получить статус запущенной задачи по ее идентификатору.

- Входные параметры
- Результирующие данные
- Примеры вызова TaskWS (SOAP)
  - Пример 1. Получен статус задачи
  - Пример 2. Задача с указанным идентификатором не существует
- Примеры вызова TaskRA (JSON)
  - Пример 1. Получен статус задачи

## Входные параметры

Имя параметра	Тип	Обязательный?	Описание
id	string	да	Идентификатор задачи.

## Результирующие данные

Имя параметра	Тип	Описание
state	WTaskState (Литера )	Статус задачи.
startDate	dateTime	Дата и время запуска задачи.
endDate	dateTime	Дата и время окончания выполнения задачи.
description	string	Описание завершения задачи — текст ошибки, либо ОК при успешном выполнении.

## Примеры вызова TaskWS (SOAP)

### Пример 1. Получен статус задачи

Запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/task/15_3">
  <soapenv:Header/>
  <soapenv:Body>
    <getTaskStatusRequest>a1b038c3-952f-4dd8-8ea2-c65c34c2de72<
  /getTaskStatusRequest>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getTaskStatusResponse xmlns:ns2="http://hflabs.ru/cdi/soap/2_13" xmlns="http://hflabs.ru/cdi/task/15_3">
      <state>SKIPPED</state>
      <startDate>2015-05-05T17:05:35</startDate>
      <endDate>2015-05-05T17:05:35</endDate>
      <description>Job 'deDuplicationWithHouseholdSearchTaskPerformer' is skipped, because following jobs are running: 'deDuplicationWithHouseholdSearchTaskPerformer'</description>
    </getTaskStatusResponse>
  </soap:Body>
</soap:Envelope>
```

## Пример 2. Задача с указанным идентификатором не существует

Запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/task/15_3">
  <soapenv:Header/>
  <soapenv:Body>
    <getTaskStatusRequest>a1b038c3-952f-4dd8-8ea2-c65c34c2de72</getTaskStatusRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode xmlns:ns1="http://hflabs.ru/cdi/soap/2_13">ns1:INVALID_REQUEST</faultcode>
      <faultstring>Task with ID a1b038c3-952f-4dd8-8ea2-c65c34c2de72' not found</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## Примеры вызова TaskRA (JSON)

### Пример 1. Получен статус задачи

Запрос:

```
GET:cdi/soap/services/15_3/TaskRA/status/a4870c9b-5b67-4256-b53c-7653439c907e
```

ОТВЕТ:

```
{  
  "state": "FINISHED",  
  "startDate": 1430838976209,  
  "endDate": 1430839004264,  
  "description": "OK"  
}
```

# TASKWS И TASKRA (WSDL И СХЕМЫ)

Файл	Описание
<a href="#">taskws.wsdl</a>	WSDL сервиса TaskWS
<a href="#">taskws-msg.xsd</a>	Модель данных сервисов TaskWS и TaskRA
<a href="#">taskws-model.xsd</a>	Модель данных задач управления данными в системе



# CLEANSERVICE

Интерфейс **CleanService** позволяет стандартизовать клиентские данные в онлайн-режиме.

## SOAP:

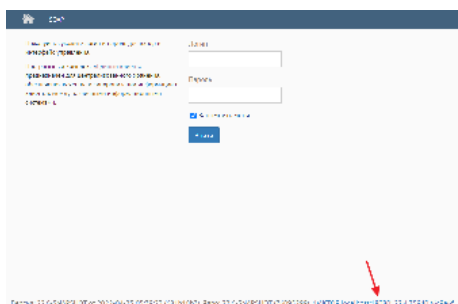
- **Endpoint address:** `http://[сервер]:[порт]/factor-service-<имя_заказчика>/api/services/ws/CleanService`
- **WSDL:** `http://[сервер]:[порт]/factor-service-<имя_заказчика>/api/services/ws/CleanService?wsdl`

## REST:

- **Endpoint address:** `http://[сервер]:[порт]/factor-service-<имя_заказчика>/api/services/rs/CleanService`
- **WADL:** `http://[сервер]:[порт]/factor-service-<имя_заказчика>/api/services/rs/CleanService?_wadl`

**i** Ссылку до сервиса CleanService на вашем контуре можно скопировать прямо из интерфейса системы. Для этого надо открыть страницу версий Фактора ? Сервисы ? **CleanService**

1. Перейти на страницу версий Фактора (в подвале сайта ссылка):



2. Справа сверху нажать ссылку «Сервисы»:

Factor 22.4.75940#6c9a6f93 2022-04-20		Сервисы
Версии справочников		
agent		2022-04-16
changeMobileProvider		2022-04-06
doubtful		2022-04-03

3. Нас интересует раздел **CleanServicePortType** (остальные — внутренние технические сервисы Фактора). Там можно получить прямую ссылку на WSDL сервиса:

Available SOAP services:

<b>CdiBlackListServicePortType</b> <ul style="list-style-type: none"><li>getBlackListOfflineStatus</li><li>blackListSearch</li><li>runBlackListOffline</li></ul>	<b>Endpoint address:</b> <a href="http://10.0.18.47:18080/factor-service-cdi-demo/api/services/ws/CdiBlackListService">http://10.0.18.47:18080/factor-service-cdi-demo/api/services/ws/CdiBlackListService</a> <b>WSDL :</b> <a href="http://blacklist.cdi.soap.cleaner.hflabs.ru/CdiBlackListService">http://blacklist.cdi.soap.cleaner.hflabs.ru/CdiBlackListService</a> <b>Target namespace:</b> <a href="http://blacklist.cdi.soap.cleaner.hflabs.ru">http://blacklist.cdi.soap.cleaner.hflabs.ru</a>
<b>CleanServicePortType</b> <ul style="list-style-type: none"><li>doXmlClean</li><li>doClean</li><li>doBatchClean</li></ul>	<b>Endpoint address:</b> <a href="http://10.0.18.47:18080/factor-service-cdi-demo/api/services/ws/CleanService">http://10.0.18.47:18080/factor-service-cdi-demo/api/services/ws/CleanService</a> <b>WSDL :</b> <a href="http://clean.soap.cleaner.hflabs.ru/CleanService">http://clean.soap.cleaner.hflabs.ru/CleanService</a> <b>Target namespace:</b> <a href="http://clean.soap.cleaner.hflabs.ru">http://clean.soap.cleaner.hflabs.ru</a>
<b>DedupServicePortType</b> <ul style="list-style-type: none"><li>dumpDeDuplicationData</li><li>retrieveDeDuplicationStatus</li><li>findOnlineDuplicates</li><li>findIncrementalCrossDuplicates</li></ul>	<b>Endpoint address:</b> <a href="http://10.0.18.47:18080/factor-service-cdi-demo/api/services/ws/DedupService">http://10.0.18.47:18080/factor-service-cdi-demo/api/services/ws/DedupService</a> <b>WSDL :</b> <a href="http://generated.dedup.soap.cleaner.hflabs.ru/DedupService">http://generated.dedup.soap.cleaner.hflabs.ru/DedupService</a> <b>Target namespace:</b> <a href="http://generated.dedup.soap.cleaner.hflabs.ru">http://generated.dedup.soap.cleaner.hflabs.ru</a>

Для REST ссылку можно найти на этой же странице в списке: «Available RESTful services». Выбираем ссылку для сервиса — **CleanService**.

## Методы

- **doClean** — унифицированный метод, который позволяет стандартизовать любой из компонентов, составляющих клиентскую запись (одиночные реквизиты, адреса, документ, телефоны, email). Метод принимает на входе массив значений полей и имя *мэппинга*, которым следует стандартизовать указанные поля.

Обработка одной записи.

Метод doClean позволяет отправить одну клиентскую запись на стандартизацию в «Фактор».

Формат и правила стандартизации данных задаются с помощью мэппинга, имя которого необходимо указать в параметре запроса *mapping*. Имя мэппинга — константа, которую сообщает сотрудник HFLabs. Доступных мэппингов может быть несколько, например, один для адресов и один для ФИО. В одном запросе — только один мэппинг.

## Входные данные

**String[] data** — массив данных для обработки;

**String mapping** — название мэппинга.

## Результирующие данные

**String[] data** — массив обработанных данных.

- **doBatchClean** — метод аналогичный doClean, но позволяющий выполнить стандартизацию сразу же нескольких записей.

Обработка нескольких записей в одном запросе

Метод `doBatchClean` позволяет отправить на стандартизацию в «Фактор» несколько клиентских записей в одном запросе.

Формат и правила стандартизации данных задаются с помощью мэппинга, имя которого необходимо указать в параметре запроса `mapping`. Имя мэппинга — константа, которую сообщает сотрудник HFLabs. Доступных мэппингов может быть несколько, например, один для адресов и один для ФИО. В одном запросе — только один мэппинг.

❗ Рекомендуем формировать запросы максимум на 400 записей.

При большом количестве записей «Фактор» будет долго занят и помешает процессам, использующим `doClean`.

При пачках более 1000 записей корректную работу не гарантируем.

## ***Входные данные***

**`String[][] data`** — массив данных для обработки;

**`String mapping`** — название мэппинга.

## ***Результирующие данные***

**`String[][] data`** — массив обработанных данных.

## ***Примеры***

В последующих разделах приведено описание параметров вызова для каждого из компонентов контрагента.

- `CleanService.address` — Стандартизация адреса
- `CleanService.email` — Стандартизация адреса электронной почты
- `CleanService.legal` — Стандартизация реквизитов контрагента юридического лица
- `CleanService.passport` — Стандартизация ДУЛ
- `CleanService.phone` — Стандартизация телефона
- `CleanService.physical` — Стандартизация реквизитов контрагента
- `CleanService.soap-name-declension` — Падежные формы ФИО

# CLEANSERVICE.ADDRESS

- Стандартизация адреса

- Входные параметры

- Результирующие данные

- Формат входных данных

- Формат результирующих данных

- Примеры вызова SOAP

- Пример 1. Адрес с частичным объединением компонентов

- Пример 2. Адрес одной строкой

- Пример 3. Несколько адресов

- Примеры вызова JSON

- Пример 1. Адрес одной строкой

## Стандартизация адреса

### *Входные параметры*

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	Массив значений полей адреса для стандартизации.
mapping	string	да	clean-address

### *Результирующие данные*

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных полей адреса.

### *Формат входных данных*

Поле	Описание	Комментарий
1 id	Идентификатор	Копируется без изменений.
2 additionalInfo	Дополнительная информация	Копируется без изменений.
3 postalCode	Почтовый индекс ИЛИ адрес целиком	Если поле содержит адрес целиком, остальные (ниже) поля заполнять не надо.

4	country	Наименование страны	
5	regionType	Тип региона	
6	region	Наименование региона	Если поле наименования уже содержит тип, заполнять его ещё раз не нужно.
7	rayonType	Тип района в регионе	
8	rayon	Наименование района в регионе	
9	cityType	Тип города	
10	city	Наименование города	
11	settlementType	Тип населенного пункта	
12	settlement	Наименование населенного пункта	
13	streetType	Тип улицы	
14	street	Наименование улицы	
15	houseNumber	Номер дома	
16	militaryNumber	Номер военной части	
17	postalBox	Номер абонентского ящика	
18	korpus	Корпус	
19	stroenie	Строение	
20	vladenie	Владение	
21	hostel	Общежитие	Номер или просто указание на то, что это общежитие («ОБЩ»).
22	floor	Этаж	
23	entrance	Номер подъезда	
24	section	Секция	
25	flat	Первая квартира	
26	flat2	Вторая квартира	
27	office	Первый офис	
28	office2	Второй офис	
29	room	Первая комната	
30	room2	Вторая комната	
31	cityAreaType	Тип района города	
32	cityArea	Район города	

## Формат результирующих данных

Поле	Описание	Комментарий
1 id	Идентификатор	
additi onalIn 2 fo	Дополнительная информация	
rawSo 3 urce	Исходный адрес одной строкой	
fullAd dress 4	Результирующий адрес одной строкой	Адрес собирается по универсальной маске из полученных значений компонент как в ФИАС, без дополнительных преобразований и перестановок.
postal 5 Code	Почтовый индекс	Больше подходит для почтовых рассылок.
kladrP ostalC 6 ode	Почтовый индекс по КЛАДР	Больше подходит для отчётности ФНС.
7 country	Наименование страны	
region 8 Type	Тип региона	
9 region	Наименование региона	
rayon 10 Type	Тип района в регионе	
11 rayon	Наименование района в регионе	
cityTy 12 pe	Тип города	
13 city	Наименование города	
settle mentT 14 ype	Тип населенного пункта	
settle 15 ment	Наименование населенного пункта	
street 16 Type	Тип улицы	
17 street	Наименование улицы	
house Numb 18 er	Номер дома	
militar yNum 19 ber	Номер военной части	
postal	Номер абонентского ящика	

20	Box		
21	korpus	Корпус	
22	stroenie	Строение	
23	vladen	Владение	
24	hostel	Общежитие	Номер или просто указание на то, что это общежитие («ОБЩ»).
25	floor	Этаж	
26	entrance	Номер подъезда	
27	section	Секция	
28	flat	Первая квартира	
29	flat2	Вторая квартира	
30	office	Первый офис	
31	office2	Второй офис	
32	room	Первая комната	
33	room2	Вторая комната	
34	district	Федеральный округ	
35	cityAreaType	Тип района города	С релиза 16.9.
36	cityArea	Район города	С релиза 16.9.
37		резерв	
38		резерв	
39		резерв	
40	qualityCode	Код полноты адреса (Литера)	
41	validationCode	Код проверки адреса (Литера)	
42	validationExtension	Расширение валидатора	Сомнительная часть адреса.
43	qualityDate	Дата актуальности ФИАС	
44	kladrCode	Код КЛАДР адреса	
45	fiasId	Идентификатор ФИАС	Идентификатор ФИАС самого подходящего объекта из ФИАС.

46	fiasLevel	Уровень распознавания по ФИАС (Литера)	Уровень, до которого адрес сопоставлен с ФИАС.
47	okatoCode	Код ОКАТО	
48	fnsNumber	Код ИФНС для ЮЛ	
49	timezoneUTC	Часовой пояс по UTC	
50	timezoneMSK	Часовой пояс по Москве	
51	fiasRegionId	Идентификатор ФИАС региона	С релиза 16.4.
52	fiasRayonId	Идентификатор ФИАС района	С релиза 16.4.
53	fiasCityId	Идентификатор ФИАС города	С релиза 16.4.
54	fiasNpId	Идентификатор ФИАС населённого пункта	С релиза 16.4.
55	fiasStreetId	Идентификатор ФИАС улицы	С релиза 16.4.
56	fiasExtraIdE	Идентификатор ФИАС дополнительной территории	С релиза 16.4.
57	fiasHouseId	Идентификатор ФИАС дома	С релиза 16.4.
58	fiasCityAreaId	Идентификатор ФИАС по району города	С релиза 16.9.
59	fiasExtraSubIdE	Идентификатор ФИАС по доподтерритории	С релиза 16.9.
60	popularity	Популярность (Литера)	С релиза 17.8.
61	flatArea	Площадь квартиры	С релиза 20.11, заполняется только при наличии лицензии на модуль «Расширенная информация по адресам (площади и стоимости)».
62	flatPrice	Стоимость квартиры	С релиза 20.11, заполняется только при наличии лицензии на модуль «Расширенная информация по адресам (площади и стоимости)».
63	geoLatitude	Координаты: широта	С релиза 20.11, заполняется только при наличии лицензии на модуль «Геокодирование».
64	geoLongitude	Координаты: долгота	С релиза 20.11, заполняется только при наличии лицензии на модуль «Геокодирование».
65	geoQualityCode	Код точности координат	С релиза 20.11, заполняется только при наличии лицензии на модуль «Геокодирование».
	geoBelwayH	Признак нахождения внутри кольцевой	С релиза 20.11, только для Москвы и Санкт-Петербурга. Заполняется только при наличии лицензии на модуль



66	it		«Геокодирование».
67	geoBel twayDi stance	Расстояние от кольцевой в километрах	С релиза 20.11, только для Москвы и Санкт-Петербурга. Заполняется только при наличии лицензии на модуль «Геокодирование».
68	oksmC ode	ОКСМ	С релиза 20.11. Для заполнения требуется изменение настроек.
69	oktmo Nearest	Ближайший ОКТМО. Если есть точный код, возвращаем точный, если нет, то ближайший, если нет, то от родителя	С релиза 20.11.
70	liter	Литера	С релиза 20.11. Для заполнения требуется изменение настроек.
71	constr uction	Сооружение	С релиза 20.11. Для заполнения требуется изменение настроек.
72		резерв	
73		резерв	
74		резерв	
75		резерв	
76		резерв	
77		резерв	
78		резерв	
79		резерв	
80	compa rtment1	Первое помещение	С релиза 20.11.
81	compa rtment2	Второе помещение	С релиза 20.11.
82	taxOffi ceCod eFl	Код ИФНС для ФЛ	С релиза 20.11.
83	mosco wCity Area	Административный округ	С релиза 20.11, только для Москвы (ЦАО, ВАО и др.).
84	fiasCo de	Иерархический код адреса в ФИАС (СС+PPP+ГГГ+ППП+СССС+УУУУ+ДДДД)	С релиза 20.11.
85		резерв под здание	
86	fiasIdF lat	Идентификатор ФИАС квартиры/помещения	Будет заполняться у тех, кто штатно использует full-фиас
87	fiasFla tCadN umber	Кадастровый номер квартиры/помещения	Будет заполняться у тех, кто штатно использует full-фиас

## Примеры вызова SOAP

### Пример 1. Адрес с частичным объединением компонентов

В примере типы компонентов объединены со значениями — "г Москва".

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleener.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <!-- 1 --> <data>id</data>
      <!-- 2 --> <data>additional info</data>
      <!-- 3 --> <data/>
      <!-- 4 --> <data>Россия</data>
      <!-- 5 --> <data>г Москва</data>
      <!-- 6 --> <data/>
      <!-- 7 --> <data/>
      <!-- 8 --> <data/>
      <!-- 9 --> <data>г Москва</data>
      <!-- 10 --> <data/>
      <!-- 11 --> <data/>
      <!-- 12 --> <data/>
      <!-- 13 --> <data>Озерковская</data>
      <!-- 14 --> <data/>
      <!-- 15 --> <data>50</data>
      <!-- 16 --> <data/>
      <!-- 17 --> <data/>
      <!-- 18 --> <data/>
      <!-- 19 --> <data>1</data>
      <!-- 20 --> <data/>
      <!-- 21 --> <data/>
      <!-- 22 --> <data/>
      <!-- 23 --> <data/>
      <!-- 24 --> <data/>
      <!-- 25 --> <data/>
      <!-- 26 --> <data/>
      <!-- 27 --> <data>201</data>
      <!-- 28 --> <data/>
      <!-- 29 --> <data/>
      <!-- 30 --> <data/>
      <!-- 21 --> <data/>
      <!-- 32 --> <data/>
      <mapping>clean-address</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 2. Адрес одной строкой

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleener.hflabs.ru">
```

```

<soapenv:Header/>
<soapenv:Body>
  <cle:doCleanRequest>
    <data>id</data>
    <data>additional info</data>
    <data>москва озерковская д. 50 стр1 оф 201</data>
    <mapping>clean-address</mapping>
  </cle:doCleanRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Ответ (идентичен для примеров 1 и 2)

```

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>москва озерковская д. 50 стр1 оф 201,</data>
      <data>115054, РОССИЯ, Г МОСКВА, Г МОСКВА, НАБ ОЗЕРКОВСКАЯ, дом 50, стр. 1,
офис 201,</data>
      <data>115054</data>
      <data>115054</data>
      <data>РОССИЯ</data>
      <data>Г</data>
      <data>МОСКВА</data>
      <data></data>
      <data></data>
      <data>Г</data>
      <data>МОСКВА</data>
      <data></data>
      <data></data>
      <data>НАБ</data>
      <data>ОЗЕРКОВСКАЯ</data>
      <data>50</data>
      <data></data>
      <data></data>
      <data></data>
      <data>1</data>
      <data></data>
      <data></data>
      <data></data>
      <data></data>
      <data></data>
      <data></data>
      <data></data>
      <data>201</data>
      <data></data>
      <data></data>
      <data></data>
      <data>ЦЕНТРАЛЬНЫЙ</data>
      <data></data>
      <data></data>
      <data></data>
      <data></data>
      <data>GOOD</data>
      <data>VALIDATED</data>
      <data></data>
      <data>09.06.2023</data>
      <data>770000000021080007</data>
    </ns2:doCleanResponse>
  </soap:Body>
</soap:Envelope>

```

```

<data>f139f20f-fcf1-4632-853c-e01593d5d8c9</data>
<data>FIAS_HOUSE</data>
<data>45286560000</data>
<data>7705</data>
<data>UTC+3</data>
<data>MSK+0</data>
<data>0c5b2444-70a0-4932-980c-b4dc0d3f02b5</data>
<data/>
<data/>
<data/>
<data>8c50bf05-c9f7-4960-b23d-81497ad86d0c</data>
<data/>
<data>f139f20f-fcf1-4632-853c-e01593d5d8c9</data>
<data/>
<data/>
<data>NONE</data>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data>643</data>
<data>45376000</data>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data/>
<data>7705</data>
<data>Центральный</data>
<data>77000000000000021080007</data>
<data/>
<data/>
<data/>
</ns2:doCleanResponse>
</soap:Body>
</soap:Envelope>

```

### Пример 3. Несколько адресов

Запрос

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doBatchCleanRequest>
      <data>
        <dataFields></dataFields>
        <dataFields></dataFields>
        <dataFields>Сама, Ново-Вокзальный,10,5</dataFields>
      </data>
    </cle:doBatchCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>

```







```

</dataFields/>
<dataFields/>
<dataFields/>
<dataFields/>
<dataFields>7705</dataFields>
<dataFields>Центральный</dataFields>
<dataFields>7700000000000021080007</dataFields>
<dataFields/>
<dataFields/>
<dataFields/>
</data>
</ns2:doBatchCleanResponse>
</soap:Body>
</soap:Envelope>

```

## Примеры вызова JSON

### Пример 1. Адрес одной строкой

Запрос

```

{"doCleanRequest":
  {
    "mapping":"clean-address",
    "data":[
      "id",
      "additional info",
      "москва озерковская д. 50 стр1 оф 201"
    ]
  }
}

```

Ответ

```

{
  "doCleanResponse": {
    "data": [
      "id",
      "additional info",
      "москва озерковская д. 50 стр1 оф 201, ",
      "115054, РОССИЯ, Г МОСКВА, Г МОСКВА, НАБ ОЗЕРКОВСКАЯ, дом 50, стр. 1, офис
201, ",
      "115054",
      "115054",
      "РОССИЯ",
      "Г",
      "МОСКВА",
      "",
      "",
      "Г",
      "МОСКВА",
      "",
      "",
      "НАБ",
      "ОЗЕРКОВСКАЯ",
      "50",
      "",
      ""
    ]
  }
}

```





```
"7705",
"Центральный",
"77000000000000021080007",
,,,
,,,
,,,
}
}
```

### **i** Примечание

При запросе через doBatchCleanRequest, если запросить один адрес, то в ответе возвращается единичное значение, а не массив с одним элементом:

```
Для одного адреса
{ "doBatchCleanResponse":
  { "data":
    { "dataFields":[ ] }
  }
}
```

```
Для нескольких адресов
{ "doBatchCleanResponse":
  { "data":
    [
      { "dataFields":[ ] },
      { "dataFields":[ ] },
    ]
  }
}
```

# CLEANSERVICE.EMAIL

- Стандартизация адреса электронной почты
  - Входные параметры
  - Результирующие данные
  - Формат входных данных
  - Формат результирующих данных
  - Примеры вызова SOAP
    - Пример 1. Стандартизация адреса электронной почты
    - Пример 2. Временный емейл
  - Примеры вызова JSON
    - Пример 1. Стандартизация адреса электронной почты

## Стандартизация адреса электронной почты

### *Входные параметры*

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	Массив значений полей email для стандартизации.
mapping	string	да	clean-email

### *Результирующие данные*

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных полей email.

### *Формат входных данных*

Поле	Описание	Комментарий
1 id	Идентификатор	Копируется без изменений.
2 additionalInfo	Дополнительная информация	Копируется без изменений.
3 email	Адрес электронной почты	

## Формат результирующих данных

Поле	Описание	Комментарий
1 id	Идентификатор	
2 additionalInfo	Дополнительная информация	
3 rawSource	Исходный адрес электронной почты	
4 email	Результирующий адрес электронной почты	
5 qualityCode	Код качества email (Литера)	
6 popularity	Популярность (Литера)	С релиза 17.8.

## Примеры вызова SOAP

### Пример 1. Стандартизация адреса электронной почты

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>ivanov@gmail.com</data>
      <mapping>clean-email</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>ivanov@gmail.com</data>
      <data>IVANOV@GMAIL.COM</data>
      <data>EDITED</data>
      <data>NONE</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 2. Временный емейл

Запрос

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="
http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>xjf91838@0-mail.com</data>
      <mapping>clean-email</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

ОТВЕТ

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>xjf91838@0-mail.com</data>
      <data>XJF91838@0-MAIL.COM</data>
      <data>GOOD_DISPOSABLE</data>
      <data>NONE</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Примеры вызова JSON

### Пример 1. Стандартизация адреса электронной почты

Запрос

```

{"doCleanRequest":
  {
    "mapping": "clean-email",
    "data": [
      "id",
      "additional info",
      "ivanov@gmail.com"
    ]
  }
}

```

ОТВЕТ

```

{"doCleanResponse": {"data": [
  "id",
  "additional info",
  "ivanov@gmail.com",
  "IVANOV@GMAIL.COM",
  "EDITED",
  "NONE"
]}}

```

# CLEANSERVICE.LEGAL

- Стандартизация реквизитов контрагента юридического лица
  - Входные параметры
  - Результирующие данные
- Формат входных данных
- Формат результирующих данных
  - Примеры вызова SOAP
    - Пример 1. Полный набор информации
  - Примеры вызова JSON
    - Пример 1. Полный набор информации

## Стандартизация реквизитов контрагента юридического лица

### *Входные параметры*

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	Массив значений реквизитов для стандартизации.
mapping	string	да	clean-legal

### *Результирующие данные*

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных реквизитов.

## Формат входных данных

Поле	Описание	Комментарий
1 id	Идентификатор	Копируется без изменений
2 additionalInfo	Дополнительная информация	Копируется без изменений
3 fullName	Полное наименование (включая ОПФ)	
4 opf	ОПФ	

5	shortName	Краткое наименование (без ОПФ)
6	foreignName	Иностранное наименование
7	residenceFlg	Флаг резидента
8	inn	ИНН
9	kpp	КПП
10	ogrn	ОГРН
11	okved	ОКВЭД
12	okpo	ОКПО

## Формат результирующих данных

Поле	Описание	
1	id	Идентификатор
2	additionalInfo	Дополнительная информация
3	fullNameRawSource	Исходное полное наименование
4	opfRawSource	Исходное ОПФ
5	name	Полное наименование без ОПФ
6	opf	Сокращенная ОПФ
7	nameStatus	Статус распознавания (литера)
8	shortName	Краткое наименование
9	foreignName	Иностранное наименование
10	residenceFlg	Флаг резидента
11	innRawSource	Исходный ИНН
12	inn	ИНН
13	innQualityCode	Код качества ИНН (литера)
14	kppRawSource	Исходный КПП
15	kpp	КПП
16	kppQualityCode	Код качества КПП (Литера)
17	ogrnRawSource	Исходный ОГРН
18	ogrn	ОГРН
19	ogrnQualityCode	Код качества ОГРН (литера)
20	okvedRawSource	Исходный ОКВЭД
21	okved	ОКВЭД

22	okvedQualityCode	Код качества ОКВЭД (литера)
23	okpoRawSource	Исходный ОКПО
24	okpo	ОКПО
25	okpoQualityCode	Код качества ОКПО (литера)
26	egrulStatus	Всегда NOT_CHECKED.

Во входных параметрах метода все реквизиты необязательные. Если значение реквизита неизвестно или отсутствует, его необходимо передавать как <data/>

(если следующие по порядку поля заполнены), либо вообще не указывать (если следующие по порядку поля все пустые).

## Примеры вызова SOAP

### Пример 1. Полный набор информации

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>СПОРТМАСТЕР</data>
      <data>ООО</data>
      <data>СПОРТ</data>
      <data>SPORTMASTER</data>
      <data></data>
      <data>7728551528</data>
      <data>772801001</data>
      <data>1057747320278</data>
      <data>63.12</data>
      <data>0140351061</data>
      <mapping>clean-legal</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>СПОРТМАСТЕР</data>
      <data>ООО</data>
      <data>СПОРТМАСТЕР</data>
      <data>ООО</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>
```



```

<data>EDITED</data>
<data>СПОРТ</data>
<data>SPORTMASTER</data>
<data>UNKNOWN</data>
<data>7728551528</data>
<data>7728551528</data>
<data>GOOD</data>
<data>772801001</data>
<data>772801001</data>
<data>GOOD</data>
<data>1057747320278</data>
<data>1057747320278</data>
<data>GOOD</data>
<data>63.12</data>
<data>63.12</data>
<data>GOOD</data>
<data>0140351061</data>
<data>0140351061</data>
<data>GOOD</data>
<data>NOT_CHECKED</data>
</ns2:doCleanResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Примеры вызова JSON

### Пример 1. Полный набор информации

Запрос:

```

{"doCleanRequest":
{
  "mapping":"clean-legal",
  "data":[
    "id",
    "additional info",
    "СПОРТМАСТЕР",
    "ООО",
    "СПОРТ",
    "SPORTMASTER",
    "",
    "7728551528",
    "772801001",
    "1057747320278",
    "63.12",
    "0140351061"
  ]
}
}

```

Ответ:

```

{"doCleanResponse": {"data": [
  "id",
  "additional info",
  "СПОРТМАСТЕР",
  "ООО",

```

**"СПОРТМАСТЕР",  
"000",  
"EDITED",  
"СПОРТ",  
"SPORTMASTER",  
"UNKNOWN",  
"7728551528",  
"7728551528",  
"GOOD",  
"772801001",  
"772801001",  
"GOOD",  
"1057747320278",  
"1057747320278",  
"GOOD",  
"63.12",  
"63.12",  
"GOOD",  
"0140351061",  
"0140351061",  
"GOOD",  
"NOT\_CHECKED"**

**}}}**

# CLEANSERVICE.PASSPORT

- Стандартизация ДУЛ

Входные параметры

Результирующие данные

Формат входных данных

Формат результирующих данных

Примеры вызова SOAP

Пример 1. Серия и номер в одном поле

Пример 2. Невалидный паспорт — поиск по ПНД (перечню недействительных документов)

Пример 3. Истек срок действия — заполнена expirationDate

Пример 4. Истек срок действия — определено по дате рождения

Примеры вызова JSON

Пример 1. Серия и номер в одном поле

## Стандартизация ДУЛ

### *Входные параметры*

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	Массив значений полей документа для стандартизации.
mapping	string	да	clean-document

### *Результирующие данные*

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных полей документа.

### *Формат входных данных*

Поле	Описание	Комментарий
1 id	Идентификатор	Копируется без изменений.
2 additionalInfo	Дополнительная информация	Копируется без изменений.

3	documentType	Тип документа, удостоверяющего личность (Литера)	
4	documentSeries	Серия документа	Серия документа ИЛИ серия и номер документа одной строкой.
5	documentNumber	Номер документа	
6	issueDate	Дата выдачи документа	
7	issueAuthority	Орган, выдавший документ	
8	departmentCode	Код подразделения органа, выдавшего документ	
9	expirationDate	Дата окончания действия документа	
10	birthdate	Дата рождения владельца документа	

## Формат результирующих данных

	Поле	Описание	Комментарий
1	id	Идентификатор	
2	additionalInfo	Дополнительная информация	
3	documentType	Тип документа, удостоверяющего личность (Литера)	
4	rawSource	Исходные данные одной строкой	
5	documentSeries	Серия документа	
6	documentNumber	Номер документа	
7	issueDate	Дата выдачи документа	
8	issueAuthority	Орган, выдавший документ	
9	departmentCode	Код подразделения органа, выдавшего документ	
10	expirationDate	Дата окончания действия документа	
11	qualityCode	Код качества документа (Литера)	
12	qualityReason	Причина невалидности документа (плохого кода качества)	Список литеральных кодов через запятую.

13	outOfDateFlag	Признак документа с истекшим сроком действия	Если срок действия документа закончился, вернет значение <b>Obsolete</b> , иначе пустоту.  Выставляется либо по дате окончания (все типы), либо по дате выдачи (права + загран), либо по комбинации даты выдачи и ДР (паспорт РФ /СССР)
14	invalidStatus	Признак недействительного паспорта (перечень МВД)	INVALID для недействительного паспорта, иначе пусто.
15	popularity	Популярность (Литера)	С релиза 17.8.

Номер и серию можно передавать в поле **documentSeries**, либо отдельно (серию в **documentSeries**, номер в **documentNumber**).

Дату выдачи, орган выдачи и код подразделения необходимо передавать в отдельных полях.

## Примеры вызова SOAP

### Пример 1. Серия и номер в одном поле

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>45 15 170222</data>
      <data/>
      <data>12.03.2012</data>
      <data>ОВД Мневники</data>
      <data>301-028</data>
      <data/>
      <data>01.01.1990</data>
      <mapping>clean-document</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ


```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>45 15 170222 , выдан 12.03.2012 ОВД Мневники 301-028</data>
      <data>45 15</data>
      <data>170222</data>
      <data>12.03.2012</data>
      <data>ОВД МНЕВНИКИ</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

<data>301-028</data>
<data>01.01.2035</data>
<data>EDITED</data>
<data/>
<data>>false</data>
<data/>
<data>NONE</data>
</ns2:doCleanResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Пример 2. Невалидный паспорт — поиск по ПНД (перечню недействительных документов)

 Чтобы проверка по ПНД начала работать — подключите справочник по инструкции.

Дополнительных работ со стороны ХФЛабс делать не нужно — проверка по ПНД входит в модуль «Документы»

Запрос

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>4507 654343</data>
      <data/>
      <data>12.03.2012</data>
      <data>ОВД Мневники</data>
      <data>301-028</data>
      <data/>
      <data>01.01.1990</data>
      <mapping>clean-document</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ

*Паспорт найден в перечне ПНД — Не действителен (ЗАМЕНЕН НА НОВЫЙ)*

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>4507 654343 , выдан 12.03.2012 ОВД Мневники 301-028</data>
      <data>45 07</data>
      <data>654343</data>
      <data>12.03.2012</data>
      <data>ОВД Мневники</data>
      <data>301-028</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<data>01.01.2035</data>
<data>EDITED</data>
<data/>
<data>>false</data>
<data>INVALID</data>
<data>NONE</data>
</ns2:doCleanResponse>
</soapenv:Body>
</soapenv:Envelope>

```

### Пример 3. Истек срок действия — заполнена expirationDate

Запрос

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="
http://clean.soap.cleener.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>45 15 170222</data>
      <data/>
      <data>12.03.2012</data>
      <data>ОВД Мневники</data>
      <data>301-028</data>
      <data>01.01.2000</data> <!-- expirationDate уже наступило -->
      <data/>
      <mapping>clean-document</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

Ответ. В поле **outOfDateFlag** — **Obsolete**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleener.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>45 15 170222 , выдан 12.03.2012 ОВД Мневники 301-028, истекает 01.01.2000
    </data>
      <data>45 15</data>
      <data>170222</data>
      <data>12.03.2012</data>
      <data>ОВД МНЕВНИКИ</data>
      <data>301-028</data>
      <data>01.01.2000</data>
      <data>EDITED</data>
      <data>Obsolete by out_of_date</data>
      <data>Obsolete</data>
      <data/>
      <data>NONE</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Пример 4. Истек срок действия — определено по дате рождения

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleener.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>45 15 170222</data>
      <data/>
      <data>12.03.1975</data> <!-- Выдан паспорт в 75 году -->
      <data>ОВД Мневники</data>
      <data>301-028</data>
      <data/>
      <data>01.01.1960</data> <!-- А дата рождения — 60. То есть паспорт выдан в 15 лет, а сейчас ему уже далеко за 20, пора менять! -->
      <mapping>clean-document</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ. В поле **outOfDateFlag** — **Obsolete**

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleener.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>PASSPORT_RU</data>
      <data>45 15 170222 , выдан 12.03.1975 ОВД Мневники 301-028</data>
      <data>45 15</data>
      <data>170222</data>
      <data>12.03.1975</data>
      <data>ОВД МНЕВНИКИ</data>
      <data>301-028</data>
      <data>01.01.1980</data>
      <data>EDITED</data>
      <data>Obsolete by out_of_date</data>
      <data>Obsolete</data>
      <data/>
      <data>NONE</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Примеры вызова JSON

### Пример 1. Серия и номер в одном поле

Запрос



```
{"doCleanRequest":  
{  
  "mapping":"clean-document",  
  "data":[  
    "id",  
    "additional info",  
    "PASSPORT_RU",  
    "4507 654343",  
    "",  
    "12.03.2012",  
    "ОВД Мневники",  
    "301-028",  
    "",  
    "01.01.1990",  
  ]  
}  
}
```

Ответ

```
{"doCleanResponse": {"data": [  
  "id",  
  "additional info",  
  "PASSPORT_RU",  
  "4507 654343 , выдан 12.03.2012 ОВД Мневники 301-028",  
  "45 07",  
  "654343",  
  "12.03.2012",  
  "ОВД МНЕВНИКИ",  
  "301-028",  
  "01.01.2035",  
  "EDITED",  
  "",  
  "false",  
  "",  
  "NONE"  
  ]}}
```

# ПРИЧИНА НЕВАЛИДНОСТИ ДОКУМЕНТА

qualityReason	Комментарий
INVALID by PNP	найден в справочнике недействительных документов
NOT_VALID_UNDER_MIN_AGE	просроченный паспорт, определенный при сравнении даты выдачи с датой рождения
NOT_SURE by empty date	пустая дата выдачи
NOT_SURE by date	невалидная дата выдачи (в будущем или сильно в прошлом)
Prolonged by out_of_date	<i>Для паспорта:</i> указана дата истечения, она меньше текущей, но укладывается в срок 90 дней <i>Для водительского удостоверения:</i> дата истечения меньше текущей, но находится в промежутке с 01.01.2022 по 31.12.2023, а также прошло менее 3 лет по отношению к указанной дате окончания
Obsolete by out_of_date	указана дата истечения, она меньше текущей и не укладывается в срок 90 дней
NOT_SURE by sn	некорректный формат серии/номера или недействительная серия
NOT_SURE by code	неверный формат кода подразделения
NOT_SURE by delivery	меньше 14 лет для паспорта РФ, меньше 16 для паспорта СССР,
NOT_SURE by series issue mismatch	дата выдачи не соответствует серии паспорта

Причин может быть несколько, перечисляются через запятую:

**NOT\_SURE by sn,NOT\_SURE by date,NOT\_SURE by code**

# CLEANSERVICE.PHONE

- Стандартизация телефона
  - Входные параметры
  - Результирующие данные
  - Формат входных данных
  - Формат результирующих данных
  - Примеры вызова SOAP
    - Пример 1. Гранулярный телефон
    - Пример 2. Телефон одной строкой
  - Примеры вызова JSON
    - Пример 1. Запрос (гранулярный телефон)

## Стандартизация телефона

### Входные параметры

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	Массив значений полей телефона для стандартизации.
mapping	string	да	clean-phone

### Результирующие данные

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных полей телефона.

### Формат входных данных

	Поле	Описание	Комментарий
1	id	Идентификатор	Копируется без изменений.
2	additionalInfo	Дополнительная информация	Копируется без изменений.
3	phoneType	Тип телефона (Литера)	
	countryCode	Код страны	Код страны ИЛИ номер одной строкой. Во втором случае остальные

4		телефонные поля заполнять не нужно.
5	cityCode	Код города
6	phoneNumber	Номер телефона
7	phoneExtension	Добавочный номер
8	addressRegion	Регион Вспомогательная адресная информация, которая используется при подборе и проверке кода города телефона.
9	addressRayon	Район
10	addressCity	Город
11	addressSettlement	Населённый пункт
12	addressCountry	Страна

## Формат результирующих данных

Поле	Описание	Комментарий
1	id	Идентификатор
2	additionalInfo	Дополнительная информация
3	phoneType	Тип телефона (Литера)
4	rawSource	Исходные данные одной строкой
5	phoneCountryCode	Код страны
6	phoneCityCode	Код города
7	phoneNumber	Номер телефона
8	phoneExtension	Добавочный номер
9	phoneComment	Комментарий Дополнительная строковая информация, выделенная из телефона.
10	qualityCode	Код качества телефона (Литера)
11	numberProfile	Характеристика номера телефона (Литера)
12	operator	Оператор Оператор, которому распределён диапазон по справочнику Россвязи.
13	operatorRegion	Регион Регион, в котором распределён диапазон по справочнику Россвязи.
14	movedNumberFlag	Признак перенесённого телефона
15	codeCountry	Страна По коду страны.

16	codeRegion	Регион	По коду города.
17	codeRayon	Район	По коду города.
18	codeCity	Город	По коду города.
19	codeSettlement	Населённый пункт	По коду города.
20	timezoneUTC	Часовой пояс по UTC	
21	timezoneMSK	Часовой пояс по Москве	
22	popularity	Популярность (Литера)	С релиза 17.8.

## Примеры вызова SOAP

### Пример 1. Гранулярный телефон

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>OTHER</data>
      <data>7</data>
      <data>495</data>
      <data>2235443</data>
      <mapping>clean-phone</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

### Пример 2. Телефон одной строкой

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>OTHER</data>
      <data>74952235443</data>
      <mapping>clean-phone</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>OTHER</data>
      <data>74952235443</data>
      <data>7</data>
      <data>495</data>
      <data>2235443</data>
      <data/>
      <data/>
      <data>GOOD</data>
      <data>PROBABLY_MOBILE</data>
      <data>ПАО "Мобильные ТелеСистемы"</data>
      <data>Москва</data>
      <data/>
      <data>РОССИЯ</data>
      <data>МОСКВА</data>
      <data/>
      <data>МОСКВА</data>
      <data/>
      <data>UTC+3</data>
      <data>MSK+0</data>
      <data>NONE</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Примеры вызова JSON

### Пример 1. Запрос (гранулярный телефон)

```

{"doCleanRequest":
  {
    "mapping":"clean-phone",
    "data":[
      "id",
      "additional info",
      "OTHER",
      "7",
      "495",
      "2235443"
    ]
  }
}

```

Ответ

```

{"doCleanResponse": {"data": [
  "id",
  "additional info",
  "OTHER",
  "74952235443",
  "7",
  "495",
  "2235443",
  "",

```

```
""  
,  
"GOOD",  
"PROBABLY_MOBILE",  
"ПАО "Мобильные ТелеСистемы"",  
"Москва",  
""  
,  
"7",  
"МОСКВА",  
""  
,  
"МОСКВА",  
""  
,  
"UTC+3",  
"MSK+0",  
"NONE"  
}}}
```

# CLEANSERVICE.PHYSICAL

- Стандартизация реквизитов контрагента
  - Входные параметры
  - Результирующие данные
- Формат входных данных
- Формат результирующих данных
  - Примеры вызова SOAP
    - Пример 1. Полный набор информации
  - Примеры вызова JSON
    - Пример 1. Полный набор информации

## Стандартизация реквизитов контрагента

### *Входные параметры*

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	Массив значений реквизитов для стандартизации.
mapping	string	да	clean-physical

### *Результирующие данные*

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных реквизитов.

## Формат входных данных

Поле	Описание	Комментарий
1 id	Идентификатор	Копируется без изменений.
2 additionalInfo	Дополнительная информация	Копируется без изменений.
3 surname	Фамилия	Фамилия или ФИО одной строкой.
4 name	Имя	
5 patronymic	Отчество	



6	gender	Пол	Принимаются значения «MALE / FEMALE», «М / Ж», «М / F»
7	birthdate	Дата рождения	
8	inn	ИНН	
9	snils	СНИЛС	Не стандартизируется, если нет лицензии на очистку документов

## Формат результирующих данных

Поле	Описание
1 id	Идентификатор
2 additionalInfo	Дополнительная информация
3 fioRawSource	Исходное ФИО одной строкой
4 genderRawSource	Исходный пол
5 surname	Фамилия
6 name	Имя
7 patronymic	Отчество
8 gender	Пол (Литера)
9 fioQualityCode	Код качества ФИО (Литера)
10 birthdateRawSource	Исходная дата рождения
11 birthdate	Дата рождения
12 birthdateQualityCode	Код качества ДР (Литера)
13 innRawSource	Исходный ИНН
14 inn	ИНН
15 innQualityCode	Код качества ИНН (Литера)
16 snilsRawSource	Исходный СНИЛС
17 snils	СНИЛС
18 snilsQualityCode	Код качества СНИЛС (Литера)

Во входных параметрах метода все реквизиты необязательные. Если значение реквизита неизвестно или отсутствует, его необходимо передавать как <data/>

(если следующие по порядку поля заполнены), либо вообще не указывать (если следующие по порядку поля все пустые).

## Примеры вызова SOAP

### Пример 1. Полный набор информации

Запрос

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>id</data>
      <data>additional info</data>
      <data>Мамонов</data>
      <data>Сергей</data>
      <data>Викторович</data>
      <data>MALE</data>
      <data>03.12.1987</data>
      <data>519300081166</data>
      <data>123-456-789 00</data>
      <mapping>clean-physical</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>id</data>
      <data>additional info</data>
      <data>Мамонов Сергей Викторович</data>
      <data>MALE</data>
      <data>МАМОНОВ</data>
      <data>СЕРГЕЙ</data>
      <data>ВИКТОРОВИЧ</data>
      <data>MALE</data>
      <data>EDITED</data>
      <data>03.12.1987</data>
      <data>03.12.1987</data>
      <data>EDITED</data>
      <data>519300081166</data>
      <data>519300081166</data>
      <data>GOOD</data>
      <data>123-456-789 00</data>
      <data/>
      <data>NOT_VALID_CHECK_SUM</data>
    </ns2:doCleanResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Примеры вызова JSON

### Пример 1. Полный набор информации

Запрос:

```
{"doCleanRequest":  
  {  
    "mapping":"clean-physical",  
    "data":[  
      "id",  
      "additional info",  
      "Мамонов",  
      "Сергей",  
      "Викторович",  
      "MALE",  
      "03.12.1987",  
      "519300081166",  
      "123-456-789 00"  
    ]  
  }  
}
```

Ответ:

```
{"doCleanResponse": {"data": [  
  "id",  
  "additional info",  
  "Мамонов Сергей Викторович",  
  "MALE",  
  "МАМОНОВ",  
  "СЕРГЕЙ",  
  "ВИКТОРОВИЧ",  
  "MALE",  
  "EDITED",  
  "03.12.1987",  
  "03.12.1987",  
  "EDITED",  
  "519300081166",  
  "519300081166",  
  "GOOD",  
  "123-456-789 00",  
  "",  
  "NOT_VALID_CHECK_SUM"  
  ]}  
}}
```

# CLEANSERVICE.SOAP-NAME-DECLENSION

- Падежные формы ФИО
  - Входные параметры
  - Результирующие данные
  - Формат входных данных
    - Входные параметры
    - Результирующие данные
  - Код качества склонения
  - Примеры вызова SOAP
  - Примеры вызова REST

## Падежные формы ФИО

### *Входные параметры*

Имя параметра	Тип	Обязательный?	Описание
data	string[]	да	ФИО для стандартизации
mapping	string	да	soap-name-declension

### *Результирующие данные*

Имя параметра	Тип	Описание
data	string[]	Массив стандартизованных полей

### *Формат входных данных*

#### Входные параметры

N	Имя параметра	Обязательный?	Описание
1	Исходное ФИО	да	ФИО одной строкой
2	Пол	нет	Пол

#### Результирующие данные

N	Параметр	Описание
---	----------	----------

1	Исх. ФИО	Исходные ФИО
2	Фамилия	
3	Имя	
4	Отчество	
5	Пол	
6	Код качества ФИО	
7	Фамилия	родительный падеж
8	Имя	
9	Отчество	
10	Фамилия	дательный падеж
11	Имя	
12	Отчество	
13	Фамилия	творительный падеж
14	Имя	
15	Отчество	
16	Код качества склонения фамилии	Код качества склонения компоненты ФИО
17	Код качества склонения имени	
18	Код качества склонения отчества	

## Код качества склонения

Значение кода качества	Комментарий
GOOD	Ручная проверка не требуется. Падежные формы найдены в словаре склонений.
NOT_SURE	На проверку. Падежные формы не найдены в словаре склонений. Либо удалось применить правила, либо нет и слово признано несклоняемым.
UNDEF	Результат пуст. Исходный пол неизвестен, исходные данные меньше 3 символов или пустые.

## Примеры вызова SOAP

Запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cle="http://clean.soap.cleaner.hflabs.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <cle:doCleanRequest>
      <data>Иванов Андрей Васильевич</data>
      <mapping>soap-name-declension</mapping>
    </cle:doCleanRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</cle:doCleanRequest>
</soapenv:Body>
</soapenv:Envelope>
```

ОТВЕТ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:doCleanResponse xmlns:ns2="http://clean.soap.cleaner.hflabs.ru">
      <data>Иванов Андрей Васильевич</data>
      <data>Иванов</data>
      <data>Андрей</data>
      <data>Васильевич</data>
      <data>MALE</data>
      <data>EDITED</data>
      <data>Иванова</data>
      <data>Андрея</data>
      <data>Васильевича</data>
      <data>Иванову</data>
      <data>Андрею</data>
      <data>Васильевичу</data>
      <data>Ивановым</data>
      <data>Андреем</data>
      <data>Васильевичем</data>
      <data>GOOD</data>
      <data>GOOD</data>
      <data>GOOD</data>
    </ns2:doCleanResponse>
  </soap:Body>
</soap:Envelope>
```

## Примеры вызова REST

Запрос:

```
{
  "doCleanRequest":{
    "mapping":"soap-name-declension",
    "data":[
      "Иванов Андрей Васильевич",
      "мужчина"
    ]
  }
}
```

Ответ:

```
{
  "doCleanResponse":{
    "data":[
      "Иванов Андрей Васильевич",
      "Иванов",
      "Андрей",
      "Васильевич",
      "MALE",
      "EDITED",
      "Иванова",

```

**"Андрея",  
"Васильевича",  
"Иванову",  
"Андрею",  
"Васильевичу",  
"Ивановым",  
"Андреем",  
"Васильевичем",  
"GOOD",  
"GOOD",  
"GOOD"**

**]  
}  
}**

# БЕЗОПАСНОСТЬ

Для аутентификации при вызове веб-сервисов PartyWS и PartyRA применяется HTTP Basic authentication. В качестве учетной записи можно использовать любого пользователя с правами Менеджера данных.

Пример вызова веб-сервиса с использованием HTTP Basic authentication:

```
POST http://localhost:8088/cdi/soap/services/2_12/PartyWS HTTP/1.1  
Accept-Encoding: gzip,deflate  
Content-Type: text/xml;charset=UTF-8  
SOAPAction: "http://hflabs.ru/cdi/soap/PartyWS/getByHID"  
User-Agent: Jakarta Commons-HttpClient/3.1  
Content-Length: 262  
Authorization: Basic b3BlcmF0b3I6MQ==  
Host: localhost:8088  
  
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://hflabs.ru/cdi/soap/2_12">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <getByHID>  
      <hid>1024</hid>  
    </getByHID>  
  </soapenv:Body>  
</soapenv:Envelope>
```

При вызове веб-сервиса CleanService аутентификация не требуется.



# ОСОБЕННОСТИ СЕРВИСОВ

Полезная информация для интеграции с нами:

Регистр значейний флагов, которые мы возвращаем в ответе зависит от типа.

Для boolean возвращаем в нижнем регистре — true/false

Для threestate — в верхнем TRUE/FALSE/UNKNOWN

# ОПИСАНИЕ JMS-ИНТЕРФЕЙСА

- Расширения JMS-сообщения
- Ограничения сервиса
- Примеры использования
- Структура JMS сообщения
- JMS.FAQ

# РАСШИРЕНИЯ JMS-СООБЩЕНИЯ

- Создание и удаление связей
- Детализация изменения полей
- Детализация закрытия контрагента
- Передача прошлых ФИО и ДУЛ

Единый Клиент поддерживает изменение структуры и наполнения сообщения через настройку параметров в разделе конфигурация. Есть возможность расширить формат для следующих событий:

- Создание и удаление связей - в сообщении добавятся **rawId** и **sourceSystem** карточки
- Детализация изменения полей - в сообщении, при изменении полей, будет добавлена информация о старом значении и новом, присвоенном, значении поля
- Детализация закрытия контрагента - в сообщении, при закрытии контрагента, будет содержаться полная карточка КА

## Создание и удаление связей

Для расширения сообщений о создании/удалении связей, отправляемых в JMS очередь, информацией о **rawId** и **sourceSystem** карточки необходимо:

1. В АРМ Администратора -> Конфигурация -> [Параметры сервисов](#) для параметра **services.relations.fillSourceIdByDefault** установить значение - **true**
2. Перезапустить сервис Единого Клиента

Пример:

XML
<pre>&lt;jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;jms:action type="CREATE_RELATION"&gt;     &lt;relation type="1" hashCode="ac1af1a4cd07cee8f06fdbee1ab8b820"&gt;       &lt;first type="PHYSICAL" hid="1"/&gt;       &lt;second type="PHYSICAL" hid="2"/&gt;     &lt;/relation&gt;   &lt;/jms:action&gt; &lt;/jms:event&gt;</pre>
JSON
<pre>{   "action": [     {       "attribute": null,       "changes": null,       "relation": {         "first": {           "type": "PHYSICAL",</pre>

```

    "hid": 1,
    "rawId": null,
    "sourceSystem": null,
  },
  "second": {
    "type": "PHYSICAL",
    "hid": 2,
    "rawId": null,
    "sourceSystem": null
  },
  "type": 1,
  "deleted": null,
  "hashCode": "ac1af1a4cd07cee8f06fdbee1ab8b820"
},
"type": "CREATE_RELATION"
}
],
"stamp": 1695313143893
}

```

#### XML

```

<jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CREATE_RELATION">
    <relation type="1" hashCode="ac1af1a4cd07cee8f06fdbee1ab8b820">
      <first type="PHYSICAL" hid="1" rawId="test1" sourceSystem="AL"/>
      <second type="PHYSICAL" hid="2" rawId="test2" sourceSystem="BI"/>
    </relation>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": null,
      "relation": {
        "first": {
          "type": "PHYSICAL",
          "hid": 1,
          "rawId": "test1",
          "sourceSystem": "AL"
        },
        "second": {
          "type": "PHYSICAL",
          "hid": 2,
          "rawId": "test2",
          "sourceSystem": "BI"
        },
        "type": 1,
        "deleted": null,
        "hashCode": "ac1af1a4cd07cee8f06fdbee1ab8b820"
      },
      "type": "CREATE_RELATION"
    }
  ],
  "stamp": 1695313143893
}

```

⚠ Важно: При большом объеме запросов на создание связей включение данной настройки может повлиять на скорость работы системы, поэтому настройку нужно включить сначала в тестовом режиме. При обнаружении проблем со скоростью отклика отключить расширенное оповещение: выставить для параметра **services.relations.fillSourceIdByDefault** значение **false** и перезапустить сервис «Единого клиента» (**HFLABS CDI**)

## Детализация изменения полей

Для включения в сообщения информации о каждом изменившемся поле, необходимо:

1. В АРМ Администратора -> Конфигурация -> [Параметры сервисов](#) для параметра **jms.events.changeDetailsEnabled** установить значение - **true**
2. Перезапустить сервис Единого Клиента

Данная настройка добавит информацию о каждом изменившемся поле в событии. Пример:

XML
<pre>&lt;jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13"&gt;   &lt;jms:action type="UPDATE"&gt;     &lt;party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"&gt;       &lt;field name="surname"&gt;ВЯХОРЕВ-НЕВСКИЙ&lt;/field&gt;       &lt;field name="name"&gt;АЛЕКСАНДР&lt;/field&gt;       &lt;field name="patronymic"&gt;АЛЕКСАНДРОВИЧ&lt;/field&gt;       &lt;field name="fullNameQC"&gt;EDITED&lt;/field&gt;       &lt;field name="fullNameAuthor"&gt;AL:1&lt;/field&gt;       &lt;field name="fullNameRawSource"&gt;Вяхорев-Невский Александр Александрович&lt;/field&gt;     &lt;/party&gt;     &lt;attribute type="PHONE" hid="1024" rawId="AL.PH.1"&gt;       &lt;field name="author"&gt;AL:1&lt;/field&gt;       &lt;field name="type"&gt;HOME&lt;/field&gt;       &lt;field name="countryCode"&gt;7&lt;/field&gt;       &lt;field name="cityCode"&gt;495&lt;/field&gt;       &lt;field name="number"&gt;2171522&lt;/field&gt;       &lt;field name="qualityCode"&gt;GOOD&lt;/field&gt;       &lt;field name="rawSource"&gt;8-495-217-15-22&lt;/field&gt;     &lt;/attribute&gt;   &lt;/jms:action&gt; &lt;/jms:event&gt;</pre>
JSON
<pre>{   "action": [     {       "attribute": null,       "party": [         {           "field": [             {               "value": "ВЯХОРЕВ-НЕВСКИЙ",               "name": "surname"             },             {               "value": "АЛЕКСАНДР",</pre>

```

    "name": "name"
  },
  {
    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев-Невский Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  }
],
"type": "PHYSICAL"
}

```

```

    ],
    "relation": null,
    "type": "CREATE"
  }
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change type="UPDATE" essence="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
        <jms:detail fieldName="fullNameRawSource">
          <jms:oldValue>Вяхорев Александр Александрович</jms:oldValue>
          <jms:newValue>Вяхорев-Невский Александр Александрович</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="surname">
          <jms:oldValue>ВЯХОРЕВ</jms:oldValue>
          <jms:newValue>ВЯХОРЕВ-НЕВСКИЙ</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ-НЕВСКИЙ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев-Невский Александр Александрович</field>
    </party>
    <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
      <field name="author">AL:1</field>
      <field name="type">HOME</field>
      <field name="countryCode">7</field>
      <field name="cityCode">495</field>
      <field name="number">2171522</field>
      <field name="qualityCode">GOOD</field>
      <field name="rawSource">8-495-217-15-22</field>
    </attribute>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ-НЕВСКИЙ",
              "name": "surname"
            },
            {

```

```

    "value": "АЛЕКСАНДР",
    "name": "name"
  },
  {
    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев-Невский Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  }
],
"type": "PHYSICAL"

```



```

    }
  ],
  "changes": {
    "change": [
      {
        "detail": [
          {
            "oldValue": "Вяхорев Александр Александрович",
            "newValue": "Вяхорев-Невский Александр Александрович",
            "fieldName": "fullNameRawSource"
          },
          {
            "oldValue": "ВЯХОРЕВ",
            "newValue": "ВЯХОРЕВ-НЕВСКИЙ",
            "fieldName": "fullNameRawSource"
          }
        ]
      },
      {
        "type": "UPDATE",
        "essence": "PHYSICAL",
        "hid": 1024,
        "rawId": "1",
        "sourceSystem": "AL"
      }
    ]
  },
  "relation": null,
  "type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Детализация закрытия контрагента

Расширение сообщения о закрытии КА подключается по запросу

## Передача прошлых ФИО и ДУЛ

Для включения в сообщения информации о прошлых ФИО и ДУЛ, необходимо:

1. В АРМ Администратора -> Конфигурация -> [Параметры сервисов](#) для параметра **jms.events.attributes.pastValues** установить значение - **true**
2. Перезапуск сервиса Единого Клиента не требуется

Данная настройка добавит информацию о прошлых ФИО и ДУЛ в событиях

- UPDATE;
- MERGE;
- CLOSE при подключенном расширенном сообщении о закрытии клиента;
- UNMERGE.

```

<ns2:event stamp="2023-10-02T14:07:47" xmlns="http://hflabs.ru/cdi/soap/2_13" xmlns:ns2="
http://hflabs.ru/cdi/jms/2_27">
  <ns2:action type="MERGE">

```

```

<party type="PHYSICAL" hid="1024" sourceSystem="QLF" rawId="1"/>
</ns2:action>
<ns2:action type="MERGE">
  <party type="PHYSICAL" hid="1027" sourceSystem="QLF" rawId="2"/>
</ns2:action>
<ns2:action type="CREATE">
  <party type="PHYSICAL" hid="1028">
    <field name="surname">Виноградов</field>
    <field name="name">Вадим</field>
    <field name="patronymic">Петрович</field>
    <field name="gender">MALE</field>
    <field name="genderRawSource">UNKNOWN</field>
    <field name="fullNameQC">CONFIRMED_MANUALLY</field>
    <field name="fullNameAuthor">admin_performer</field>
    <field name="fullNameRawSource">Виноградов Егор Петрович</field>
    <field name="actualityDateAuthor">admin_performer</field>
    <field name="actualityDate">02.10.2023 14:06:26</field>
    <field name="lastChangeTimestamp">1696244867969</field>
    <attribute type="DOCUMENT_PASSPORT" hid="1028" rawId="DOCUMENT_PASSPORT.
1024">
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="issueDate">01.04.1994</field>
      <field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
      <field name="departmentCode">111-437</field>
      <field name="expiryDate">01.04.2024</field>
      <field name="invalidStatus">ACTUAL</field>
      <field name="popularity">NONE</field>
      <field name="qualityCode">CONFIRMED_MANUALLY</field>
      <field name="type">PASSPORT</field>
      <field name="documentSeries">72 06</field>
      <field name="documentNumber">821258</field>
      <field name="rawSource">72 06•821255•01.04.1994•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-433•01.04.2024</field>
      <field name="primary">true</field>
      <field name="additionalInfo">паспорт</field>
      <field name="actualityDate">02.10.2023 14:02:35</field>
      <field name="author">admin_performer</field>
    </attribute>
    <pastAttribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1024">
      <field name="qualifierType">PASSPORT_RU</field>
      <field name="issueDate">01.04.1994</field>
      <field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
      <field name="departmentCode">111-437</field>
      <field name="expiryDate">01.04.2024</field>
      <field name="invalidStatus">ACTUAL</field>
      <field name="popularity">NONE</field>
      <field name="qualityCode">CONFIRMED_MANUALLY</field>
      <field name="type">PASSPORT</field>
      <field name="documentSeries">72 06</field>
      <field name="documentNumber">821255</field>
      <field name="rawSource">72 06•821255•01.04.1994•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-433•01.04.2024</field>
      <field name="primary">false</field>
      <field name="additionalInfo"></field>
      <field name="actualityDate">02.10.2023 14:02:35</field>
      <field name="author">admin_performer</field>
    </pastAttribute>
    <pastAttribute type="FIO">
      <field name="qualityCode">EDITED</field>
      <field name="surname">Виноградов</field>
      <field name="name">Егор</field>
      <field name="patronymic">Петрович</field>

```

```
<field name="type">DEFAULT</field>
<field name="rawSource">Виноградов Егор Петрович</field>
<field name="primary">>false</field>
<field name="additionalInfo"></field>
<field name="actualityDate">02.10.2023 14:02:16</field>
<field name="author">QLF:1</field>
</pastAttribute>
<source hid="1024" sourceSystem="QLF" rawId="1"/>
<source hid="1027" sourceSystem="QLF" rawId="2"/>
</party>
</ns2:action>
</ns2:event>
```

```
<ns2:event stamp="2023-10-02T14:19:33" xmlns="http://hflabs.ru/cdi/soap/2_13" xmlns:ns2="
http://hflabs.ru/cdi/jms/2_27">
  <ns2:action type="CLOSE">
    <party type="PHYSICAL" hid="1033" sourceSystem="QLF" rawId="3">
      <field name="surname">Степанов</field>
      <field name="name">Антон</field>
      <field name="patronymic">Петрович</field>
      <field name="gender">MALE</field>
      <field name="fullNameRawSource">Степанов Егор Петрович</field>
      <field name="actualityDateAuthor">admin_performer</field>
      <field name="actualityDate">02.10.2023 14:19:03</field>
      <field name="lastChangeTimestamp">1696245573329</field>
      <attribute type="DOCUMENT_PASSPORT" hid="1029" rawId="DOCUMENT_PASSPORT.
1029">
        <field name="qualifierType">PASSPORT_RU</field>
        <field name="issueDate">01.04.1991</field>
        <field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
        <field name="departmentCode">111-422</field>
        <field name="expiryDate">01.04.2023</field>
        <field name="invalidStatus">OUTDATED</field>
        <field name="popularity">NONE</field>
        <field name="qualityCode">CONFIRMED_MANUALLY</field>
        <field name="type">PASSPORT</field>
        <field name="documentSeries">72 07</field>
        <field name="documentNumber">821244</field>
        <field name="rawSource">72 08•821244•01.04.1991•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-422•01.04.2023</field>
        <field name="primary">>true</field>
        <field name="additionalInfo">нацпопт</field>
        <field name="actualityDate">02.10.2023 14:19:15</field>
        <field name="author">admin_performer</field>
      </attribute>
      <pastAttribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1029">
        <field name="qualifierType">PASSPORT_RU</field>
        <field name="issueDate">01.04.1991</field>
        <field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
        <field name="departmentCode">111-422</field>
        <field name="expiryDate">01.04.2023</field>
        <field name="invalidStatus">ACTUAL</field>
        <field name="popularity">NONE</field>
        <field name="qualityCode">CODE_SERIES_CLASH</field>
        <field name="type">PASSPORT</field>
        <field name="documentSeries">72 08</field>
        <field name="documentNumber">821244</field>
        <field name="rawSource">72 08•821244•01.04.1991•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-422•01.04.2023</field>
        <field name="primary">>false</field>
        <field name="additionalInfo"></field>
```

```
<field name="actualityDate">02.10.2023 14:19:15</field>
<field name="author">QLF:3</field>
</pastAttribute>
<pastAttribute type="FIO">
  <field name="qualityCode">EDITED</field>
  <field name="surname">Степанов</field>
  <field name="name">Егор</field>
  <field name="patronymic">Петрович</field>
  <field name="type">DEFAULT</field>
  <field name="rawSource">Степанов Егор Петрович</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"></field>
  <field name="actualityDate">02.10.2023 14:19:03</field>
  <field name="author">QLF:3</field>
</pastAttribute>
</party>
</ns2:action>
</ns2:event>
```

Если у клиента более 10 прошлых атрибутов, выгружаются десять последних по времени актуальности атрибутов суммарно для ФИО и ДУЛ.

# ОГРАНИЧЕНИЯ СЕРВИСА

## Ограничения

По умолчанию не генерируются события по объектам с partyType EGR и BLACKLIST.

## ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ

- JMS. Создание контрагента
- JMS. Изменение контрагента
- JMS. Слияние контрагентов
- JMS. Закрытие контрагента
- JMS. Изменение значения поля
- JMS. Разливание контрагентов
- JMS. Создание связей
- JMS. Закрытие связей
- JMS. Восстановление на версию
- JMS. Проверка по черным спискам
- JMS. Настройка отправки прошлых значений ФИО и ДУЛ

# JMS. СОЗДАНИЕ КОНТРАГЕНТА

Исходное состояние:

- Контрагент с sourceSystem="AL" и rawId="1" отсутствует

Событие:

- Создание контрагента:  
ФИО: Вяхорев Александр Александрович  
Телефон: 7-495-2171522  
sourceSystem="AL" и rawId="1"

## XML

```
<jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CREATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>
```

## JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            }
          ]
        }
      ]
    }
  ]
}
```

```

{
  "value": "АЛЕКСАНДРОВИЧ",
  "name": "patronymic"
},
{
  "value": "EDITED",
  "name": "fullNameQC"
},
{
  "value": "AL:1",
  "name": "fullNameAuthor"
},
{
  "value": "Вяхорев Александр Александрович",
  "name": "fullNameRawSource"
}
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  }
],
"type": "PHYSICAL"
}
],
"changes": null,

```



```
"relation": null,  
"type": "CREATE"  
}  
],  
"stamp": 1695250119193  
}
```

# JMS. ИЗМЕНЕНИЕ КОНТРАГЕНТА

- Обновление из исходной системы
- Редактирование
- Изменение объединенного контрагента
- Новый атрибут
- Схлопывание атрибута
- Закрывание атрибута
- Изменение другого типа контрагента, не ФЛ
- Создание связи между атрибутами
- Закрывание связи между атрибутами

Единый клиент отправляет два вида сообщений об изменении контрагента:

1. стандартное, которое содержит только актуальное состояние карточки (используется по умолчанию);
2. **расширенное**, в котором содержатся детали изменения (доступно, начиная с релиза 16.7, подключается по запросу).

## Обновление из исходной системы

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522

Событие:

- ФИО изменилось: Вяхорев-Невский Александр Александрович

```
XML
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ-НЕВСКИЙ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев-Невский Александр Александрович<
/field>
    <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
      <field name="author">AL:1</field>
```

```

    <field name="type">HOME</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="number">2171522</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-217-15-22</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

## JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ-НЕВСКИЙ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев-Невский Александр Александрович",
              "name": "fullNameRawSource"
            }
          ],
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                },
                {
                  "value": "HOME",
                  "name": "type"
                },
                {
                  "value": "7",
                  "name": "countryCode"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}

```

```

    },
    {
      "value": "495",
      "name": "cityCode"
    },
    {
      "value": "2171522",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "8-495-217-15-22",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1024,
  "rawId": "AL.PH.1",
  "deleted": null
}
],
"type": "PHYSICAL"
}
],
"relation": null,
"type": "CREATE"
}
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change type="UPDATE" essence="PHYSICAL" hid="1024" sourceSystem="AL"
rawId="1">
        <jms:detail fieldName="fullNameRawSource">
          <jms:oldValue>Вяхорев Александр Александрович</jms:oldValue>
          <jms:newValue>Вяхорев-Невский Александр Александрович</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="surname">
          <jms:oldValue>ВЯХОРЕВ</jms:oldValue>
          <jms:newValue>ВЯХОРЕВ-НЕВСКИЙ</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ-НЕВСКИЙ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев-Невский Александр Александрович<
/field>
    <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
      <field name="author">AL:1</field>
    </attribute>
  </party>
</jms:event>

```

```

    <field name="type">HOME</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="number">2171522</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-217-15-22</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ-НЕВСКИЙ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев-Невский Александр Александрович",
              "name": "fullNameRawSource"
            }
          ],
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                },
                {
                  "value": "HOME",
                  "name": "type"
                },
                {
                  "value": "7",
                  "name": "countryCode"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}

```

```

    },
    {
      "value": "495",
      "name": "cityCode"
    },
    {
      "value": "2171522",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "8-495-217-15-22",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1024,
  "rawId": "AL.PH.1",
  "deleted": null
}
],
"type": "PHYSICAL"
}
],
"changes": {
  "change": [
    {
      "detail": [
        {
          "oldValue": "Вяхорев Александр Александрович",
          "newValue": "Вяхорев-Невский Александр Александрович",
          "fieldName": "fullNameRawSource"
        },
        {
          "oldValue": "ВЯХОРЕВ",
          "newValue": "ВЯХОРЕВ-НЕВСКИЙ",
          "fieldName": "fullNameRawSource"
        }
      ]
    },
    {
      "type": "UPDATE",
      "essence": "PHYSICAL",
      "hid": 1024,
      "rawId": "1",
      "sourceSystem": "AL"
    }
  ]
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Редактирование

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522

Событие:

- Отредактировали телефон: 7-495-2171523

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">MorozovNA</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171523</field>
        <field name="qualityCode">CONFIRMED_MANUALLY</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            }
          ]
        }
      ]
    }
  ]
}
```

```

    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "MorozovNA",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171523",
        "name": "number"
      },
      {
        "value": "CONFIRMED_MANUALLY",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ],
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  }
],
"type": "PHYSICAL"
}
],
"relation": null,
"type": "UPDATE"

```



```
}  
],  
"stamp": 1695250119193  
}
```

#### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <jms:action type="UPDATE">  
    <jms:changes>  
      <jms:change type="UPDATE" essence="PHONE" hid="1024" rawId="AL.PH.1">  
        <jms:detail fieldName="number">  
          <jms:oldValue>2171522</jms:oldValue>  
          <jms:newValue>2171523</jms:newValue>  
        </jms:detail>  
        <jms:detail fieldName="qualityCode">  
          <jms:oldValue>GOOD</jms:oldValue>  
          <jms:newValue>CONFIRMED_MANUALLY</jms:newValue>  
        </jms:detail>  
      </jms:change>  
    </jms:changes>  
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">  
      <field name="surname">ВЯХОРЕВ</field>  
      <field name="name">АЛЕКСАНДР</field>  
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>  
      <field name="fullNameQC">EDITED</field>  
      <field name="fullNameAuthor">AL:1</field>  
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>  
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">  
        <field name="author">MorozovNA</field>  
        <field name="type">HOME</field>  
        <field name="countryCode">7</field>  
        <field name="cityCode">495</field>  
        <field name="number">2171523</field>  
        <field name="qualityCode">CONFIRMED_MANUALLY</field>  
        <field name="rawSource">8-495-217-15-22</field>  
      </attribute>  
    </party>  
  </jms:action>  
</jms:event>
```

#### JSON

```
{  
  "action": [  
    {  
      "attribute": null,  
      "party": [  
        {  
          "field": [  
            {  
              "value": "ВЯХОРЕВ",  
              "name": "surname"  
            },  
            {  
              "value": "АЛЕКСАНДР",  
              "name": "name"  
            },  
            {  
              "value": "АЛЕКСАНДРОВИЧ",  
              "name": "patronymic"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```

    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "MorozovNA",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171523",
        "name": "number"
      },
      {
        "value": "CONFIRMED_MANUALLY",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ],
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  }
],
"type": "PHYSICAL"
}
],
"changes": {
  "change": [
    {

```

```

    "detail": [
      {
        "oldValue": "2171522",
        "newValue": "2171523",
        "fieldName": "number"
      },
      {
        "oldValue": "GOOD",
        "newValue": "CONFIRMED_MANUALLY",
        "fieldName": "qualityCode"
      }
    ],
    "type": "UPDATE",
    "essence": "PHYSICAL",
    "hid": 1024,
    "rawId": "1",
    "sourceSystem": "AL"
  }
]
},
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Изменение объединенного контрагента

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-916-8342323
- Контрагенты AL:1 и BT:1 слились в объединенного контрагента HID = 1026.

Событие:

- В результате обновления из BT у объединенного контрагента изменился телефон: 7-495-2171523

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <action type="UPDATE">
    <party type="PHYSICAL" hid="1026">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1026" rawId="AL.PH.1">

```

```

    <field name="author">AL:1</field>
    <field name="type">MOBILE</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="number">2171523</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-217-15-23</field>
  </attribute>
  <attribute type="PHONE" hid="1027" rawId="BT.PH.1">
    <field name="author">BT:1</field>
    <field name="type">HOME</field>
    <field name="countryCode">7</field>
    <field name="cityCode">916</field>
    <field name="number">8342323</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">7-916-8342323</field>
  </attribute>
  <source hid="1024" sourceSystem="AL" rawId="1"/>
  <source hid="1025" sourceSystem="BT" rawId="1"/>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ],
          "hid": 1026,
          "sourceSystem": null,
          "rawId": null,
          "attribute": [
            {

```

```

"field": [
  {
    "value": "AL:1",
    "name": "author"
  },
  {
    "value": "MOBILE",
    "name": "type"
  },
  {
    "value": "7",
    "name": "countryCode"
  },
  {
    "value": "495",
    "name": "cityCode"
  },
  {
    "value": "2171523",
    "name": "number"
  },
  {
    "value": "GOOD",
    "name": "qualityCode"
  },
  {
    "value": "8-495-217-15-23",
    "name": "rawSource"
  }
],
"type": "PHONE",
"hid": 1026,
"rawId": "AL.PH.1",
"deleted": null
},
{
  "field": [
    {
      "value": "BT:1",
      "name": "author"
    },
    {
      "value": "HOME",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "916",
      "name": "cityCode"
    },
    {
      "value": "8342323",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    }
  ],
  "name": "BT"
}

```

```

        "value": "7-916-8342323",
        "name": "rawSource"
    }
    ],
    "type": "PHONE",
    "hid": 1027,
    "rawId": "BT.PH.1",
    "deleted": null
}
},
"source": [
{
    "field": [
        {
            "value": "false",
            "name": "archive"
        }
    ],
    "hid": 1024,
    "sourceSystem": "AL",
    "rawId": "1"
},
{
    "field": [
        {
            "value": "false",
            "name": "archive"
        }
    ],
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
}
],
"type": "PHYSICAL"
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <action type="UPDATE">
    <jms:changes>
      <jms:change type="UPDATE" essence="PHONE" hid="1027" rawId="BT.PH.1">
        <jms:detail fieldName="rawSource">
          <jms:oldValue>7-916-8342322</jms:oldValue>
          <jms:newValue>7-495-2171523</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="cityCode">
          <jms:oldValue>916</jms:oldValue>
          <jms:newValue>495</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="number">
          <jms:oldValue>8342322</jms:oldValue>
          <jms:newValue>2171523</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
  </action>
</jms:event>

```

```

    </jms:detail>
  </jms:change>
</jms:changes>
  <party type="PHYSICAL" hid="1026">
    <field name="surname">ВЯХОРЕВ</field>
    <field name="name">АЛЕКСАНДР</field>
    <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
    <field name="fullNameQC">EDITED</field>
    <field name="fullNameAuthor">AL:1</field>
    <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
    <attribute type="PHONE" hid="1026" rawId="AL.PH.1">
      <field name="author">AL:1</field>
      <field name="type">MOBILE</field>
      <field name="countryCode">7</field>
      <field name="cityCode">495</field>
      <field name="number">2171523</field>
      <field name="qualityCode">GOOD</field>
      <field name="rawSource">8-495-217-15-23</field>
    </attribute>
    <attribute type="PHONE" hid="1027" rawId="BT.PH.1">
      <field name="author">BT:1</field>
      <field name="type">HOME</field>
      <field name="countryCode">7</field>
      <field name="cityCode">916</field>
      <field name="number">8342323</field>
      <field name="qualityCode">GOOD</field>
      <field name="rawSource">7-916-8342323</field>
    </attribute>
    <source hid="1024" sourceSystem="AL" rawId="1"/>
    <source hid="1025" sourceSystem="BT" rawId="1"/>
  </party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    },
    {
      "value": "Вяхорев Александр Александрович",
      "name": "fullNameRawSource"
    }
  ],
  "hid": 1026,
  "sourceSystem": null,
  "rawId": null,
  "attribute": [
    {
      "field": [
        {
          "value": "AL:1",
          "name": "author"
        },
        {
          "value": "MOBILE",
          "name": "type"
        },
        {
          "value": "7",
          "name": "countryCode"
        },
        {
          "value": "495",
          "name": "cityCode"
        },
        {
          "value": "2171523",
          "name": "number"
        },
        {
          "value": "GOOD",
          "name": "qualityCode"
        },
        {
          "value": "8-495-217-15-23",
          "name": "rawSource"
        }
      ],
      "type": "PHONE",
      "hid": 1026,
      "rawId": "AL.PH.1",
      "deleted": null
    },
    {
      "field": [
        {
          "value": "BT:1",
          "name": "author"
        },
        {
          "value": "HOME",
          "name": "type"
        },
        {
          "value": "7",
          "name": "countryCode"
        },
        {
          "value": "916",

```



```

        "name": "cityCode"
    },
    {
        "value": "8342323",
        "name": "number"
    },
    {
        "value": "GOOD",
        "name": "qualityCode"
    },
    {
        "value": "7-916-8342323",
        "name": "rawSource"
    }
],
"type": "PHONE",
"hid": 1027,
"rawId": "BT.PH.1",
"deleted": null
}
],
"source": [
    {
        "field": [
            {
                "value": "false",
                "name": "archive"
            }
        ]
    },
    "hid": 1024,
    "sourceSystem": "AL",
    "rawId": "1"
},
    {
        "field": [
            {
                "value": "false",
                "name": "archive"
            }
        ]
    },
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
}
],
"type": "PHYSICAL"
}
],
"changes": {
    "change": [
        {
            "detail": [
                {
                    "oldValue": "7-916-8342322",
                    "newValue": "7-495-2171523",
                    "fieldName": "rawSource"
                },
                {
                    "oldValue": "916",
                    "newValue": "495",
                    "fieldName": "cityCode"
                }
            ]
        }
    ]
}

```

```

    {
      "oldValue": "8342322",
      "newValue": "2171523",
      "fieldName": "number"
    }
  ],
  "type": "UPDATE",
  "essence": "PHONE",
  "hid": 1027,
  "rawId": "BT.PH.1",
  "sourceSystem": null
}
]
},
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Новый атрибут

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522

Событие:

- Добавили мобильный телефон: 7-916-8342323

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="PHONE" hid="1025" rawId="AL.PH.2">

```

```

    <field name="author">AL:1</field>
    <field name="type">MOBILE</field>
    <field name="countryCode">7</field>
    <field name="cityCode">916</field>
    <field name="number">8342323</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">7-916-8342323</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ]
        },
        {
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                },
                {
                  "value": "HOME",
                  "name": "type"
                },
                {
                  "value": "7",

```

```

    "name": "countryCode"
  },
  {
    "value": "495",
    "name": "cityCode"
  },
  {
    "value": "2171522",
    "name": "number"
  },
  {
    "value": "GOOD",
    "name": "qualityCode"
  },
  {
    "value": "8-495-217-15-22",
    "name": "rawSource"
  }
],
"type": "PHONE",
"hid": 1024,
"rawId": "AL.PH.1",
"deleted": null
},
{
  "field": [
    {
      "value": "AL:1",
      "name": "author"
    },
    {
      "value": "MOBILE",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "916",
      "name": "cityCode"
    },
    {
      "value": "8342323",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "7-916-8342323",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1025,
  "rawId": "AL.PH.2",
  "deleted": null
}
],

```

```

    "type": "PHYSICAL"
  }
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="PHONE" hid="1025" rawId="AL.PH.2">
        <field name="author">AL:1</field>
        <field name="type">MOBILE</field>
        <field name="countryCode">7</field>
        <field name="cityCode">916</field>
        <field name="number">8342323</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">7-916-8342323</field>
      </attribute>
    </party>
    <jms:changes>
      <jms:change type="CREATE" essence="PHONE" hid="1025" rawId="AL.PH.2"/>
    </jms:changes>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {

```

```

    "value": "АЛЕКСАНДР",
    "name": "name"
  },
  {
    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  },
  {
    "field": [

```

```

    {
      "value": "AL:1",
      "name": "author"
    },
    {
      "value": "MOBILE",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "916",
      "name": "cityCode"
    },
    {
      "value": "8342323",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "7-916-8342323",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1025,
  "rawId": "AL.PH.2",
  "deleted": null
}
],
"type": "PHYSICAL"
}
],
"changes": {
  "change": [
    {
      "type": "CREATE",
      "essence": "PHONE",
      "hid": 1025,
      "rawId": "AL.PH.2",
      "sourceSystem": null
    }
  ]
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Схлопывание атрибута

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522

Событие:

- Добавили дублирующий телефон: 74952171522

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            }
          ]
        }
      ]
    }
  ]
}
```



```

    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ],
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  }
],
"type": "PHYSICAL"
}
],
"relation": null,
"type": "UPDATE"

```

```
}
],
"stamp": 1695250119193
}
```

#### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
    </party>
    <jms:changes>
      <jms:change type="CLOSE" essence="PHONE" hid="1025" rawId="AL.PH.2"/>
    </jms:changes>
  </jms:action>
</jms:event>
```

#### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            }
          ]
        }
      ]
    }
  ]
}
```

```

    },
    {
      "value": "Вяхорев Александр Александрович",
      "name": "fullNameRawSource"
    }
  ],
  "hid": 1024,
  "sourceSystem": "AL",
  "rawId": "1",
  "attribute": [
    {
      "field": [
        {
          "value": "AL:1",
          "name": "author"
        },
        {
          "value": "HOME",
          "name": "type"
        },
        {
          "value": "7",
          "name": "countryCode"
        },
        {
          "value": "495",
          "name": "cityCode"
        },
        {
          "value": "2171522",
          "name": "number"
        },
        {
          "value": "GOOD",
          "name": "qualityCode"
        },
        {
          "value": "8-495-217-15-22",
          "name": "rawSource"
        }
      ],
      "type": "PHONE",
      "hid": 1024,
      "rawId": "AL.PH.1",
      "deleted": null
    }
  ],
  "type": "PHYSICAL"
}
],
"changes": {
  "change": [
    {
      "type": "CLOSE",
      "essence": "PHONE",
      "hid": 1025,
      "rawId": "AL.PH.2",
      "sourceSystem": null
    }
  ]
}
],
"relation": null,

```

```
"type": "UPDATE"
}
],
"stamp": 1695250119193
}
```

## Закрытие атрибута

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522

Событие:

- Закрыли (удалили) телефон.

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            }
          ]
        }
      ]
    }
  ]
}
```

```

    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "AL:1",
      "name": "fullNameAuthor"
    },
    {
      "value": "Вяхорев Александр Александрович",
      "name": "fullNameRawSource"
    }
  ],
  "hid": 1024,
  "sourceSystem": "AL",
  "rawId": "1",
  "attribute": null,
  "type": "PHYSICAL"
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
    </party>
    <jms:changes>
      <jms:change type="CLOSE" essence="PHONE" hid="1024" rawId="AL.PH.1"/>
    </jms:changes>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",

```

```

    "name": "name"
  },
  {
    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": null,
"type": "PHYSICAL"
},
],
"changes": {
  "change": [
    {
      "type": "CLOSE",
      "essence": "PHONE",
      "hid": 1024,
      "rawId": "AL.PH.1",
      "sourceSystem": null
    }
  ]
},
"relation": null,
"type": "UPDATE"
},
],
"stamp": 1695250119193
}

```

## ***Изменение другого типа контрагента, не ФЛ***

Исходное состояние:

- Договор HID = 1024 (AL:1)
- Номер: 1110111

Событие:

- Исправили опечатку в номере — на 111-111

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="AGREEMENT" hid="1024" sourceSystem="AL" rawId="1">
      <field name="number">111-111</field>
      <field name="numberAuthor">AL:1</field>
      <field name="date">18.03.1967</field>
      <field name="dateAuthor">AL:1</field>
      <field name="description">Приложение к договору № 1</field>
      <field name="descriptionAuthor">AL:1</field>
      <field name="actualityDateAuthor">AL:1</field>
      <field name="actualityDate">18.03.2011</field>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "111-111",
              "name": "number"
            },
            {
              "value": "AL:1",
              "name": "numberAuthor"
            },
            {
              "value": "18.03.1967",
              "name": "date"
            },
            {
              "value": "AL:1",
              "name": "dateAuthor"
            },
            {
              "value": "Приложение к договору № 1",
              "name": "description"
            },
            {
              "value": "AL:1",
              "name": "descriptionAuthor"
            },
            {
              "value": "AL:1",
              "name": "actualityDateAuthor"
            },
            {
              "value": "18.03.2011",
              "name": "actualityDate"
            }
          ],
          "hid": 1024,
          "sourceSystem": "AL",

```

```

    "rawId": "1",
    "attribute": null,
    "type": "AGREEMENT"
  }
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="AGREEMENT" hid="1024" sourceSystem="AL" rawId="1">
      <field name="number">111-111</field>
      <field name="numberAuthor">AL:1</field>
      <field name="date">18.03.1967</field>
      <field name="dateAuthor">AL:1</field>
      <field name="description">Приложение к договору № 1</field>
      <field name="descriptionAuthor">AL:1</field>
      <field name="actualityDateAuthor">AL:1</field>
      <field name="actualityDate">18.03.2011</field>
    </party>
    <jms:changes>
      <jms:change type="UPDATE" essence="AGREEMENT" hid="1024" sourceSystem="
AL" rawId="1">
        <jms:detail fieldName="number">
          <jms:oldValue>1110111</jms:oldValue>
          <jms:newValue>111-111</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "111-111",
              "name": "number"
            },
            {
              "value": "AL:1",
              "name": "numberAuthor"
            },
            {
              "value": "18.03.1967",
              "name": "date"
            },
            {

```



```

        "value": "AL:1",
        "name": "dateAuthor"
    },
    {
        "value": "Приложение к договору № 1",
        "name": "description"
    },
    {
        "value": "AL:1",
        "name": "descriptionAuthor"
    },
    {
        "value": "AL:1",
        "name": "actualityDateAuthor"
    },
    {
        "value": "18.03.2011",
        "name": "actualityDate"
    }
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": null,
"type": "AGREEMENT"
}
],
"changes": {
    "change": [
        {
            "detail": [
                {
                    "oldValue": "1110111",
                    "newValue": "111-111",
                    "fieldName": "number"
                }
            ]
        },
        {
            "type": "UPDATE",
            "essence": "AGREEMENT",
            "hid": 1024,
            "rawId": "1",
            "sourceSystem": null
        }
    ]
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Создание связи между атрибутами

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович

- Телефон: 7-495-2171522
- Аккаунт: lucky

Событие:

- Между атрибутами телефон - аккаунт добавили связь.

```

XML
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="ACCOUNT" hid="1024" rawId="ACC.1">
        <field name="type">SKYPE</field>
        <field name="account">lucky</field>
        ...
        <field name="rawSource">lucky</field>
      </attribute>
      <attributeRelation type="1000" hid="1024">
        <first type="PHONE" rawId="PH.1" hid="1024"/>
        <second type="ACCOUNT" rawId="ACC.1" hid="1024"/>
      </attributeRelation>
    </party>
  </jms:action>
</jms:event>

```

```

JSON
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            }
          ]
        }
      ]
    }
  ]
}

```

```

{
  "value": "АЛЕКСАНДРОВИЧ",
  "name": "patronymic"
},
{
  "value": "EDITED",
  "name": "fullNameQC"
},
{
  "value": "AL:1",
  "name": "fullNameAuthor"
},
{
  "value": "Вяхорев Александр Александрович",
  "name": "fullNameRawSource"
}
],
"hid": 1024,
"sourceSystem": "AL",
"rawId": "1",
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  },
  {
    "field": [
      {
        "value": "SKYPE",
        "name": "type"
      }
    ]
  }
]

```

```

    },
    {
      "value": "lucky",
      "name": "account"
    },
    {
      "value": "lucky",
      "name": "rawSource"
    }
  ],
  "type": "ACCOUNT",
  "hid": 1024,
  "rawId": "ACC.1",
  "deleted": null
}
],
"attributRrelation" : {
  "first" : {
    "type" : "PHONE",
    "hid" : 1024,
    "rawId" : "PH.1",
    "sourceSystem" : null
  },
  "second" : {
    "type" : "ACCOUNT",
    "hid" : 1024,
    "rawId" : "ACC.1",
    "sourceSystem" : null
  },
  "type" : 1000,
  "deleted" : null,
  "hashCode" : null
},
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>

```

```

    <field name="rawSource">8-495-217-15-22</field>
  </attribute>
  <attribute type="ACCOUNT" hid="1024" rawId="ACC.1">
    <field name="type">SKYPE</field>
    <field name="account">lucky</field>
    ...
    <field name="rawSource">lucky</field>
  </attribute>
  <attributeRelation type="1000" hid="1024">
    <first type="PHONE" rawId="PH.1" hid="1024"/>
    <second type="ACCOUNT" rawId="ACC.1" hid="1024"/>
  </attributeRelation>
</party>
<jms:changes>
  <jms:change type="CREATE" essence="ATTRIBUTE_RELATION" hid="1024"/>
</jms:changes>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ]
        },
        {
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}

```

```

    },
    {
      "value": "HOME",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "495",
      "name": "cityCode"
    },
    {
      "value": "2171522",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "8-495-217-15-22",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1024,
  "rawId": "AL.PH.1",
  "deleted": null
},
{
  "field": [
    {
      "value": "SKYPE",
      "name": "type"
    },
    {
      "value": "lucky",
      "name": "account"
    },
    {
      "value": "lucky",
      "name": "rawSource"
    }
  ],
  "type": "ACCOUNT",
  "hid": 1024,
  "rawId": "ACC.1",
  "deleted": null
}
],
"attributRrelation" : {
  "first" : {
    "type" : "PHONE",
    "hid" : 1024,
    "rawId" : "PH.1",
    "sourceSystem" : null
  },
  "second" : {
    "type" : "ACCOUNT",
    "hid" : 1024,

```

```

    "rawId" : "ACC.1",
    "sourceSystem" : null
  },
  "type" : 1000,
  "deleted" : null,
  "hashCode" : null
},
"type": "PHYSICAL"
}
],
"changes": {
  "change": [
    {
      "type": "CREATE",
      "essence": "ATTRIBUTE_RELATION",
      "hid": 1024,
      "rawId": null,
      "sourceSystem": null
    }
  ]
},
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}

```

## Закрытие связи между атрибутами

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522
- Связанный с телефоном аккаунт: lucky

Событие:

- Закрыли связь между телефоном и аккаунтом.

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="PH.1">
        <field name="author">AL:1</field>

```

```

    <field name="type">HOME</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="number">2171522</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-217-15-22</field>
  </attribute>
  <attribute type="ACCOUNT" hid="1024" rawId="ACC.1">
    <field name="type">SKYPE</field>
    <field name="account">lucky</field>
    ...
    <field name="rawSource">lucky</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ],
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                },
                {

```



```

        "value": "HOME",
        "name": "type"
    },
    {
        "value": "7",
        "name": "countryCode"
    },
    {
        "value": "495",
        "name": "cityCode"
    },
    {
        "value": "2171522",
        "name": "number"
    },
    {
        "value": "GOOD",
        "name": "qualityCode"
    },
    {
        "value": "8-495-217-15-22",
        "name": "rawSource"
    }
    ],
    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
},
{
    "field": [
        {
            "value": "SKYPE",
            "name": "type"
        },
        {
            "value": "lucky",
            "name": "account"
        },
        {
            "value": "lucky",
            "name": "rawSource"
        }
    ],
    "type": "ACCOUNT",
    "hid": 1024,
    "rawId": "ACC.1",
    "deleted": null
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695250119193
}
}

```

XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="ACCOUNT" hid="1024" rawId="ACC.1">
        <field name="type">SKYPE</field>
        <field name="account">lucky</field>
        ...
        <field name="rawSource">lucky</field>
      </attribute>
    </party>
    <jms:changes>
      <jms:change type="CLOSE" essence="ATTRIBUTE_RELATION" hid="1024">
        <jms:detail fieldName="typeId">
          <jms:oldValue>1000</jms:oldValue>
        </jms:detail>
        <jms:detail fieldName="firstAttributeType">
          <jms:oldValue>PHONE</jms:oldValue>
        </jms:detail>
        <jms:detail fieldName="firstAttributeHid">
          <jms:oldValue>1024</jms:oldValue>
        </jms:detail>
        <jms:detail fieldName="firstAttributeRawId">
          <jms:oldValue>PH.1</jms:oldValue>
        </jms:detail>
        <jms:detail fieldName="secondAttributeType">
          <jms:oldValue>ACCOUNT</jms:oldValue>
        </jms:detail>
        <jms:detail fieldName="secondAttributeHid">
          <jms:oldValue>1024</jms:oldValue>
        </jms:detail>
        <jms:detail fieldName="secondAttributeRawId">
          <jms:oldValue>ACC.1</jms:oldValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [

```

```

{
  "field": [
    {
      "value": "ВЯХОРЕВ",
      "name": "surname"
    },
    {
      "value": "АЛЕКСАНДР",
      "name": "name"
    },
    {
      "value": "АЛЕКСАНДРОВИЧ",
      "name": "patronymic"
    },
    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "AL:1",
      "name": "fullNameAuthor"
    },
    {
      "value": "Вяхорев Александр Александрович",
      "name": "fullNameRawSource"
    }
  ],
  "hid": 1024,
  "sourceSystem": "AL",
  "rawId": "1",
  "attribute": [
    {
      "field": [
        {
          "value": "AL:1",
          "name": "author"
        },
        {
          "value": "HOME",
          "name": "type"
        },
        {
          "value": "7",
          "name": "countryCode"
        },
        {
          "value": "495",
          "name": "cityCode"
        },
        {
          "value": "2171522",
          "name": "number"
        },
        {
          "value": "GOOD",
          "name": "qualityCode"
        },
        {
          "value": "8-495-217-15-22",
          "name": "rawSource"
        }
      ]
    }
  ],
}

```

```

    "type": "PHONE",
    "hid": 1024,
    "rawId": "AL.PH.1",
    "deleted": null
  },
  {
    "field": [
      {
        "value": "SKYPE",
        "name": "type"
      },
      {
        "value": "lucky",
        "name": "account"
      },
      {
        "value": "lucky",
        "name": "rawSource"
      }
    ],
    "type": "ACCOUNT",
    "hid": 1024,
    "rawId": "ACC.1",
    "deleted": null
  }
],
"changes": {
  "change": [
    {
      "detail": [
        {
          "oldValue": "1000",
          "fieldName": "typeId"
        },
        {
          "oldValue": "PHONE",
          "fieldName": "firstAttributeType"
        },
        {
          "oldValue": "1024",
          "fieldName": "firstAttributeHid"
        },
        {
          "oldValue": "PH.1",
          "fieldName": "firstAttributeRawId"
        },
        {
          "oldValue": "ACCOUNT",
          "fieldName": "secondAttributeType"
        },
        {
          "oldValue": "1024",
          "fieldName": "secondAttributeHid"
        },
        {
          "oldValue": "ACC.1",
          "fieldName": "secondAttributeRawId"
        }
      ]
    },
    {
      "type": "CLOSE",
      "essence": "ATTRIBUTE_RELATION",
      "hid": 1024,

```

```
    "rawId": null,  
    "sourceSystem": null  
  }  
  ],  
  },  
  "relation": null,  
  "type": "UPDATE"  
}  
],  
"stamp": 1695250119193  
}  
]
```

# JMS. СЛИЯНИЕ КОНТРАГЕНТОВ

- Слияние двух контрагентов
- Слияние 2+1 контрагентов
- Слияние 2+2 контрагентов
- Слияние 2+1 контрагентов с изменением данных на объединенном контрагенте

## Слияние двух контрагентов

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-916-8342323

Событие:

- Контрагенты AL:1 и BT:1 сливаются в объединенного контрагента HID = 1026.

```
XML
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="MERGE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"/>
  </jms:action>
  <jms:action type="MERGE">
    <party type="PHYSICAL" hid="1025" sourceSystem="BT" rawId="1"/>
  </jms:action>
  <jms:action type="CREATE">
    <party type="PHYSICAL" hid="1026">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1026" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">MOBILE</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="PHONE" hid="1027" rawId="BT.PH.1">
        <field name="author">BT:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>
```

```
<field name="cityCode">916</field>
<field name="number">8342323</field>
<field name="qualityCode">GOOD</field>
<field name="rawSource">7-916-8342323</field>
</attribute>
<source hid="1024" sourceSystem="AL" rawId="1"/>
<source hid="1025" sourceSystem="BT" rawId="1"/>
</party>
</jms:action>
</jms:event>
```

## JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "type": "PHYSICAL"
        }
      ],
      "changes": null,
      "relation": null,
      "type": "MERGE"
    },
    {
      "attribute": null,
      "party": [
        {
          "hid": 1025,
          "sourceSystem": "BT",
          "rawId": "1",
          "type": "PHYSICAL"
        }
      ],
      "changes": null,
      "relation": null,
      "type": "MERGE"
    },
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
```

```

    "name": "fullNameQC"
  },
  {
    "value": "AL:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1026,
"sourceSystem": null,
"rawId": null,
"attribute": [
  {
    "field": [
      {
        "value": "AL:1",
        "name": "author"
      },
      {
        "value": "MOBILE",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1026,
    "rawId": "AL.PH.1",
    "deleted": null
  },
  {
    "field": [
      {
        "value": "BT:1",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      }
    ]
  }
]

```



```

        "value": "7",
        "name": "countryCode"
    },
    {
        "value": "916",
        "name": "cityCode"
    },
    {
        "value": "8342323",
        "name": "number"
    },
    {
        "value": "GOOD",
        "name": "qualityCode"
    },
    {
        "value": "7-916-8342323",
        "name": "rawSource"
    }
    ],
    "type": "PHONE",
    "hid": 1027,
    "rawId": "BT.PH.1",
    "deleted": null
}
],
"source": [
    {
        "field": [
            {
                "value": "false",
                "name": "archive"
            }
        ],
        "hid": 1024,
        "sourceSystem": "AL",
        "rawId": "1"
    },
    {
        "field": [
            {
                "value": "flase",
                "name": "archive"
            }
        ],
        "hid": 1025,
        "sourceSystem": "BT",
        "rawId": "1"
    }
    ],
    "type": "PHYSICAL"
}
],
"changes": null,
"relation": null,
"type": "CREATE"
}
],
"stamp": 1695282318938
}

```

## Слияние 2+1 контрагентов

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-916-8342323
- Оба контрагента слились в объединенного контрагента HID = 1026.
- Контрагент HID = 1027 (BT:2); ФИО: Вяхорев Александр Александрович

Событие:

- Контрагент BT:2 вливается в объединенного контрагента HID = 1026.

```
XML
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="MERGE">
    <party type="PHYSICAL" hid="1027" sourceSystem="BT" rawId="2"/>
  </jms:action>
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1026">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">BT:2</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1026" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">MOBILE</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="PHONE" hid="1027" rawId="BT.PH.1">
        <field name="author">BT:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">916</field>
        <field name="number">8342323</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">7-916-8342323</field>
      </attribute>
      <source hid="1024" sourceSystem="AL" rawId="1"/>
      <source hid="1025" sourceSystem="BT" rawId="1"/>
      <source hid="1027" sourceSystem="BT" rawId="2"/>
    </party>
  </jms:action>
</jms:event>
```

## JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "hid": 1027,
          "sourceSystem": "BT",
          "rawId": "2",
          "type": "PHYSICAL"
        }
      ],
      "changes": null,
      "relation": null,
      "type": "MERGE"
    },
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ],
          "hid": 1026,
          "sourceSystem": null,
          "rawId": null,
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                },
                {
                  "value": "MOBILE",
                  "name": "type"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

```

    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "495",
      "name": "cityCode"
    },
    {
      "value": "2171522",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "8-495-217-15-22",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1026,
  "rawId": "AL.PH.1",
  "deleted": null
},
{
  "field": [
    {
      "value": "BT:1",
      "name": "author"
    },
    {
      "value": "HOME",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "916",
      "name": "cityCode"
    },
    {
      "value": "8342323",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "7-916-8342323",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1027,
  "rawId": "BT.PH.1",
  "deleted": null
}

```



- Контрагент НID = 1027 (ВТ:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-916-8342323
- Контрагент НID = 1028 (ВТ:2); ФИО: Вяхорев Александр Александрович
- Контрагенты ВТ:1 и ВТ:2 слились в объединенного контрагента НID = 1029.

Событие:

- Контрагент НID = 1026 вливается в объединенного контрагента НID = 1029.

```

XML
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="MERGE">
    <party type="PHYSICAL" hid="1026"/>
  </jms:action>
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1029">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">ВТ:2</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1111" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">MOBILE</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="PHONE" hid="1122" rawId="BT.PH.1">
        <field name="author">BT:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">916</field>
        <field name="number">8342323</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">7-916-8342323</field>
      </attribute>
      <source hid="1024" sourceSystem="AL" rawId="1"/>
      <source hid="1025" sourceSystem="AL" rawId="2"/>
      <source hid="1027" sourceSystem="BT" rawId="1"/>
      <source hid="1028" sourceSystem="BT" rawId="2"/>
    </party>
  </jms:action>
</jms:event>

```

```

JSON
{
  "action": [
    {
      "attribute": null,
      "party": [
        {

```

```

    "hid": 1026,
    "sourceSystem": null,
    "rawId": null,
    "type": "PHYSICAL"
  }
],
"changes": null,
"relation": null,
"type": "MERGE"
},
{
  "attribute": null,
  "party": [
    {
      "field": [
        {
          "value": "ВЯХОРЕВ",
          "name": "surname"
        },
        {
          "value": "АЛЕКСАНДР",
          "name": "name"
        },
        {
          "value": "АЛЕКСАНДРОВИЧ",
          "name": "patronymic"
        },
        {
          "value": "EDITED",
          "name": "fullNameQC"
        },
        {
          "value": "BT:1",
          "name": "fullNameAuthor"
        },
        {
          "value": "Вяхорев Александр Александрович",
          "name": "fullNameRawSource"
        }
      ]
    },
    {
      "hid": 1029,
      "sourceSystem": null,
      "rawId": null,
      "attribute": [
        {
          "field": [
            {
              "value": "AL:1",
              "name": "author"
            },
            {
              "value": "MOBILE",
              "name": "type"
            },
            {
              "value": "7",
              "name": "countryCode"
            },
            {
              "value": "495",
              "name": "cityCode"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    {
      "value": "2171522",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "8-495-217-15-22",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1111,
  "rawId": "AL.PH.1",
  "deleted": null
},
{
  "field": [
    {
      "value": "BT:1",
      "name": "author"
    },
    {
      "value": "HOME",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "916",
      "name": "cityCode"
    },
    {
      "value": "8342323",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "7-916-8342323",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1122,
  "rawId": "BT.PH.1",
  "deleted": null
}
],
"source": [
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ]
  }
]

```



```

    ],
    "hid": 1024,
    "sourceSystem": "AL",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1025,
    "sourceSystem": "AL",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1027,
    "sourceSystem": "BT",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1028,
    "sourceSystem": "BT",
    "rawId": "2"
  }
],
"type": "PHYSICAL"
}
],
"changes": null,
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

## ***Слияние 2+1 контрагентов с изменением данных на объединенном контрагенте***

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр А; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр А; Телефон: 7-916-8342323

- Оба контрагента слились в объединенного контрагента HID = 1026.
- Контрагент HID = 1027 (BT:2); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522

Событие:

- Контрагент BT:2 вливается в объединенного контрагента HID = 1026, на объединенном контрагенте изменяется отчество и телефон (выигрывает из исходной BT:2).

```

XML
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="MERGE">
    <party type="PHYSICAL" hid="1027" sourceSystem="BT" rawId="2"/>
  </jms:action>
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1026">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">BT:2</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1026" rawId="BT.2.PH.1">
        <field name="author">BT:2</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
      <attribute type="PHONE" hid="1027" rawId="BT.1.PH.1">
        <field name="author">BT:1</field>
        <field name="type">MOBILE</field>
        <field name="countryCode">7</field>
        <field name="cityCode">916</field>
        <field name="number">8342323</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">7-916-8342323</field>
      </attribute>
      <source hid="1024" sourceSystem="AL" rawId="1"/>
      <source hid="1025" sourceSystem="BT" rawId="1"/>
      <source hid="1027" sourceSystem="BT" rawId="2"/>
    </party>
  </jms:action>
</jms:event>

```

```

JSON
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "hid": 1027,
          "sourceSystem": "BT",

```

```

    "rawId": "2",
    "type": "PHYSICAL"
  }
],
"changes": null,
"relation": null,
"type": "MERGE"
},
{
  "attribute": null,
  "party": [
    {
      "field": [
        {
          "value": "ВЯХОРЕВ",
          "name": "surname"
        },
        {
          "value": "АЛЕКСАНДР",
          "name": "name"
        },
        {
          "value": "АЛЕКСАНДРОВИЧ",
          "name": "patronymic"
        },
        {
          "value": "EDITED",
          "name": "fullNameQC"
        },
        {
          "value": "BT:2",
          "name": "fullNameAuthor"
        },
        {
          "value": "Вяхорев Александр Александрович",
          "name": "fullNameRawSource"
        }
      ]
    },
    {
      "hid": 1026,
      "sourceSystem": null,
      "rawId": null,
      "attribute": [
        {
          "field": [
            {
              "value": "BT:2",
              "name": "author"
            },
            {
              "value": "HOME",
              "name": "type"
            },
            {
              "value": "7",
              "name": "countryCode"
            },
            {
              "value": "495",
              "name": "cityCode"
            },
            {
              "value": "2171522",

```

```

    "name": "number"
  },
  {
    "value": "GOOD",
    "name": "qualityCode"
  },
  {
    "value": "8-495-217-15-22",
    "name": "rawSource"
  }
],
"type": "PHONE",
"hid": 1026,
"rawId": "BT.2.PH.1",
"deleted": null
},
{
  "field": [
    {
      "value": "BT:1",
      "name": "author"
    },
    {
      "value": "MOBILE",
      "name": "type"
    },
    {
      "value": "7",
      "name": "countryCode"
    },
    {
      "value": "916",
      "name": "cityCode"
    },
    {
      "value": "8342323",
      "name": "number"
    },
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    {
      "value": "7-916-8342323",
      "name": "rawSource"
    }
  ],
  "type": "PHONE",
  "hid": 1027,
  "rawId": "BT.1.PH.1",
  "deleted": null
}
],
"source": [
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ]
  },
  {
    "hid": 1024,

```

```

    "sourceSystem": "AL",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1027,
    "sourceSystem": "BT",
    "rawId": "2"
  }
],
"type": "PHYSICAL"
}
],
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="MERGE">
    <party type="PHYSICAL" hid="1027" sourceSystem="BT" rawId="2"/>
  </jms:action>
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change type="UPDATE" essence="PHYSICAL" hid="1026" rawId="">
        <jms:detail fieldName="patronymic">
          <jms:oldValue>A</jms:oldValue>
          <jms:newValue>Александрович</jms:newValue>
        </jms:detail>
      </jms:change>
      <jms:change type="UPDATE" essence="PHONE" hid="1026" rawId="BT.2.PH.
1">
        <jms:detail fieldName="mergeAuthor">
          <jms:oldValue>AL:1</jms:oldValue>
          <jms:newValue>BT:2</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="rawId">
          <jms:oldValue>AL.1.PH.1</jms:oldValue>
          <jms:newValue>BT.2.PH.1</jms:newValue>
        </jms:detail>

```

```

        </jms:change>
</jms:changes>
<party type="PHYSICAL" hid="1026">
  <field name="surname">ВЯХОРЕВ</field>
  <field name="name">АЛЕКСАНДР</field>
  <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
  <field name="fullNameQC">EDITED</field>
  <field name="fullNameAuthor">BT:2</field>
  <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
  <attribute type="PHONE" hid="1026" rawId="BT.2.PH.1">
    <field name="author">BT:2</field>
    <field name="type">HOME</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="number">2171522</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-217-15-22</field>
  </attribute>
  <attribute type="PHONE" hid="1027" rawId="BT.1.PH.1">
    <field name="author">BT:1</field>
    <field name="type">MOBILE</field>
    <field name="countryCode">7</field>
    <field name="cityCode">916</field>
    <field name="number">8342323</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">7-916-8342323</field>
  </attribute>
  <source hid="1024" sourceSystem="AL" rawId="1"/>
  <source hid="1025" sourceSystem="BT" rawId="1"/>
  <source hid="1027" sourceSystem="BT" rawId="2"/>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "hid": 1027,
          "sourceSystem": "BT",
          "rawId": "2",
          "type": "PHYSICAL"
        }
      ],
      "relation": null,
      "type": "MERGE"
    },
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",

```

```

    "name": "name"
  },
  {
    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "BT:2",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1026,
"sourceSystem": null,
"rawId": null,
"attribute": [
  {
    "field": [
      {
        "value": "BT:2",
        "name": "author"
      },
      {
        "value": "HOME",
        "name": "type"
      },
      {
        "value": "7",
        "name": "countryCode"
      },
      {
        "value": "495",
        "name": "cityCode"
      },
      {
        "value": "2171522",
        "name": "number"
      },
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      {
        "value": "8-495-217-15-22",
        "name": "rawSource"
      }
    ]
  },
  {
    "type": "PHONE",
    "hid": 1026,
    "rawId": "BT.2.PH.1",
    "deleted": null
  },
  {
    "field": [
      {

```

```

    "value": "BT:1",
    "name": "author"
  },
  {
    "value": "MOBILE",
    "name": "type"
  },
  {
    "value": "7",
    "name": "countryCode"
  },
  {
    "value": "916",
    "name": "cityCode"
  },
  {
    "value": "8342323",
    "name": "number"
  },
  {
    "value": "GOOD",
    "name": "qualityCode"
  },
  {
    "value": "7-916-8342323",
    "name": "rawSource"
  }
],
"type": "PHONE",
"hid": 1027,
"rawId": "BT.1.PH.1",
"deleted": null
}
],
"source": [
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1024,
    "sourceSystem": "AL",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ],
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
  },
  {
    "field": [
      {
        "value": "false",
        "name": "archive"
      }
    ]
  }
]

```



```

    }
    ],
    "hid": 1027,
    "sourceSystem": "BT",
    "rawId": "2"
  }
],
"type": "PHYSICAL"
}
],
"changes": {
  "change": [
    {
      "type": "UPDATE",
      "essence": "PHYSICAL",
      "hid": 1026,
      "rawId": "",
      "sourceSystem": null,
      "detail": [
        {
          "oldValue": "A",
          "newValue": "Александрович",
          "fieldName": "patronymic"
        }
      ]
    }
  ],
  "change": [
    {
      "type": "UPDATE",
      "essence": "PHONE",
      "hid": 1026,
      "rawId": "BT.2.PH.1",
      "sourceSystem": null,
      "detail": [
        {
          "oldValue": "AL:1",
          "newValue": "BT:2",
          "fieldName": "mergeAuthor"
        },
        {
          "oldValue": "AL.1.PH.1",
          "newValue": "BT.2.PH.1",
          "fieldName": "rawId"
        }
      ]
    }
  ]
},
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

# JMS. ЗАКРЫТИЕ КОНТРАГЕНТА

- Удаление исходного контрагента
- Удаление исходного контрагента, который был с кем-то объединен
- Удаление исходного контрагента, который был с кем-то объединен

Единый клиент отправляет два вида сообщений о закрытии контрагента:

1. стандартное, которое содержит только идентификаторы закрытого контрагента;
2. **расширенное**, в котором содержится полная карточка закрытого клиента (доступно, начиная с релиза 16.7, подключается по запросу).

## Удаление исходного контрагента

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- ФИО: Вяхорев Александр Александрович
- Телефон: 7-495-2171522

Событие: контрагента удалили в исходной системе.

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CLOSE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"/>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "type": "PHYSICAL"
        }
      ]
    },
    {
      "changes": null,
      "relation": null,
      "type": "CLOSE"
    }
  ]
}
```

```
"stamp": 1695314258989
}
```

#### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CLOSE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"/>
    <field name="surname">ВЯХОРЕВ</field>
    <field name="name">АЛЕКСАНДР</field>
    <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
    <field name="fullNameQC">EDITED</field>
    <field name="fullNameAuthor">BT:2</field>
    <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
  </party>
</jms:action>
</jms:event>
```

#### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "BT:2",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ]
        },
        {
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "type": "PHYSICAL"
        }
      ]
    },
    {
      "relation": null,
      "type": "CLOSE"
    }
  ]
}
```

```
}  
],  
"stamp": 1695250119193  
}
```

## Удаление исходного контрагента, который был с кем-то объединен

Исходное состояние:

- Контрагент НID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент НID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович
- Контрагент НID = 1026 (BT:2); ФИО: Вяхорев Александр Александрович
- Все три контрагента слились в объединенного контрагента НID = 1027.

Событие: контрагента AL:1 удалили в исходной системе.

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">  
  <jms:action type="CLOSE">  
    <party hid="1024" sourceSystem="AL" rawId="1"/>  
  </jms:action>  
  <jms:action type="UPDATE">  
    <party type="PHYSICAL" hid="1027">  
      <field name="surname">ВЯХОРЕВ</field>  
      <field name="name">АЛЕКСАНДР</field>  
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>  
      <field name="fullNameQC">EDITED</field>  
      <field name="fullNameAuthor">BT:2</field>  
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>  
      <source hid="1025" sourceSystem="BT" rawId="1"/>  
      <source hid="1026" sourceSystem="BT" rawId="2"/>  
    </party>  
  </jms:action>  
</jms:event>
```

### JSON

```
{  
  "action": [  
    {  
      "attribute": null,  
      "party": [  
        {  
          "field": [  
            {  
              "value": "ВЯХОРЕВ",  
              "name": "surname"  
            },  
            {  
              "value": "АЛЕКСАНДР",  
              "name": "name"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```

    },
    {
      "value": "АЛЕКСАНДРОВИЧ",
      "name": "patronymic"
    },
    {
      "value": "EDITED",
      "name": "fullNameQC"
    },
    {
      "value": "BT:2",
      "name": "fullNameAuthor"
    },
    {
      "value": "Вяхорев Александр Александрович",
      "name": "fullNameRawSource"
    }
  ],
  "hid": 1027,
  "sourceSystem": null,
  "rawId": null,
  "source": [
    {
      "field": [
        {
          "value": "false",
          "name": "archive"
        }
      ]
    },
    {
      "hid": 1025,
      "sourceSystem": "BT",
      "rawId": "1"
    },
    {
      "field": [
        {
          "value": "false",
          "name": "archive"
        }
      ]
    },
    {
      "hid": 1026,
      "sourceSystem": "BT",
      "rawId": "2"
    }
  ],
  "type": "PHYSICAL"
},
],
"relation": null,
"type": "UPDATE"
},
{
  "attribute": null,
  "party": [
    {
      "hid": 1024,
      "sourceSystem": "AL",
      "rawId": "1",
      "type": null
    }
  ]
},
],
"changes": null,

```

```

    "relation": null,
    "type": "CLOSE"
  }
],
"stamp": 1695250119193
}

```

## Удаление исходного контрагента, который был с кем-то объединен

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович
- Оба контрагента слились в объединенного контрагента HID = 1026.

Событие: контрагента AL:1 удалили в исходной системе.

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CLOSE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1"/>
  </jms:action>
  <jms:action type="CLOSE">
    <party type="PHYSICAL" hid="1026"/>
  </jms:action>
  <jms:action type="UNMERGE">
    <party type="PHYSICAL" hid="1025" sourceSystem="BT" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">BT:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
    </party>
  </jms:action>
</jms:event>

```

### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",

```

```

    "name": "name"
  },
  {
    "value": "АЛЕКСАНДРОВИЧ",
    "name": "patronymic"
  },
  {
    "value": "EDITED",
    "name": "fullNameQC"
  },
  {
    "value": "BT:1",
    "name": "fullNameAuthor"
  },
  {
    "value": "Вяхорев Александр Александрович",
    "name": "fullNameRawSource"
  }
],
"hid": 1025,
"sourceSystem": "BT",
"rawId": "1",
"type": "PHYSICAL"
},
"relation": null,
"type": "UNMERGE"
},
{
  "attribute": null,
  "party": [
    {
      "hid": 1024,
      "sourceSystem": "AL",
      "rawId": "1",
      "type": "PHYSICAL"
    }
  ]
},
"changes": null,
"relation": null,
"type": "CLOSE"
},
{
  "attribute": null,
  "party": [
    {
      "hid": 1026,
      "sourceSystem": null,
      "rawId": null,
      "type": "PHYSICAL"
    }
  ]
},
"changes": null,
"relation": null,
"type": "CLOSE"
}
},
"stamp": 1695250119193
}

```

# JMS. ИЗМЕНЕНИЕ ЗНАЧЕНИЯ ПОЛЯ

Единый клиент умеет отправлять в JMS изменения по конкретному полю (подключается как CR).

Нужно, когда система следит за одним полем и другие изменения ее не волнуют. В ответе передается только это поле, без прочих реквизитов и атрибутов.

## *Пример. Обновление статуса проверки по черным спискам.*

Исходное состояние:

- Контрагент HID = 1024 (AL:1)
- Статус проверки по ЧС: На актуализацию

Событие:

- Статус проверки по ЧС изменился: Найден

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE_FIELD">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="blacklistCheckStatus">FOUND</field>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "changes": null,
      "relation": null,
      "party": [
        {
          "field": [
            {
              "value": "FOUND",
              "name": "blacklistCheckStatus"
            }
          ],
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "type": "PHYSICAL"
        }
      ],
      "type": "UPDATE_FIELD"
    }
  ],
}
```



```
"stamp": 1695313143893  
}
```

# JMS. РАЗЛИЯНИЕ КОНТРАГЕНТОВ

- Частичное разлияние контрагентов
- Полное разлияние контрагентов

## Частичное разлияние контрагентов

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович
- Контрагент HID = 1026 (BT:2); ФИО: Вяхорев Александр Александрович
- Все три контрагента слились в объединенного контрагента HID = 1027.

Событие: контрагента AL:1 разъединили с объединенным контрагентом HID = 1027.

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UNMERGE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="number">2171522</field>
        <field name="qualityCode">GOOD</field>
        <field name="rawSource">8-495-217-15-22</field>
      </attribute>
    </party>
  </jms:action>
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1027">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">BT:2</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <source hid="1025" sourceSystem="BT" rawId="1"/>
      <source hid="1026" sourceSystem="BT" rawId="2"/>
    </party>
  </jms:action>
</jms:event>
```

```
</party>
</jms:action>
</jms:event>
```

#### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ]
        },
        {
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {
              "field": [
                {
                  "value": "AL:1",
                  "name": "author"
                },
                {
                  "value": "HOME",
                  "name": "type"
                },
                {
                  "value": "7",
                  "name": "countryCode"
                },
                {
                  "value": "495",
                  "name": "cityCode"
                },
                {
                  "value": "2171522",
```

```

        "name": "number"
    },
    {
        "value": "GOOD",
        "name": "qualityCode"
    },
    {
        "value": "8-495-217-15-22",
        "name": "rawSource"
    }
],
"type": "PHONE",
"hid": 1024,
"rawId": "AL.PH.1",
"deleted": null
}
],
"type": "PHYSICAL"
}
],
"changes": null,
"relation": null,
"type": "UNMERGE"
},
{
    "attribute": null,
    "party": [
        {
            "field": [
                {
                    "value": "ВЯХОРЕВ",
                    "name": "surname"
                },
                {
                    "value": "АЛЕКСАНДР",
                    "name": "name"
                },
                {
                    "value": "АЛЕКСАНДРОВИЧ",
                    "name": "patronymic"
                },
                {
                    "value": "EDITED",
                    "name": "fullNameQC"
                },
                {
                    "value": "BT:2",
                    "name": "fullNameAuthor"
                },
                {
                    "value": "Вяхорев Александр Александрович",
                    "name": "fullNameRawSource"
                }
            ]
        },
        {
            "hid": 1027,
            "sourceSystem": null,
            "rawId": null,
            "attribute": null,
            "source": [
                {
                    "field": [

```

```

        "value": "false",
        "name": "archive"
    }
    ],
    "hid": 1025,
    "sourceSystem": "BT",
    "rawId": "1"
},
{
    "field": [
        {
            "value": "false",
            "name": "archive"
        }
    ],
    "hid": 1026,
    "sourceSystem": "BT",
    "rawId": "2"
}
],
"type": "PHYSICAL"
}
],
"changes": null,
"relation": null,
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

## Полное разлияние контрагентов

Исходное состояние:

- Контрагент HID = 1024 (AL:1); ФИО: Вяхорев Александр Александрович; Телефон: 7-495-2171522
- Контрагент HID = 1025 (BT:1); ФИО: Вяхорев Александр Александрович
- Оба контрагента слились в объединенного контрагента HID = 1026.

Событие: контрагента AL:1 разъединили с объединенным контрагентом HID = 1026.

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UNMERGE">
    <party type="PHYSICAL" hid="1024" sourceSystem="AL" rawId="1">
      <field name="surname">ВЯХОРЕВ</field>
      <field name="name">АЛЕКСАНДР</field>
      <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
      <attribute type="PHONE" hid="1024" rawId="AL.PH.1">
        <field name="author">AL:1</field>
        <field name="type">HOME</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>

```

```

    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="number">2171522</field>
    <field name="qualityCode">GOOD</field>
    <field name="rawSource">8-495-217-15-22</field>
  </attribute>
</party>
</jms:action>
<jms:action type="UNMERGE">
  <party type="PHYSICAL" hid="1025" sourceSystem="BT" rawId="1">
    <field name="surname">ВЯХОРЕВ</field>
    <field name="name">АЛЕКСАНДР</field>
    <field name="patronymic">АЛЕКСАНДРОВИЧ</field>
    <field name="fullNameQC">EDITED</field>
    <field name="fullNameAuthor">BT:1</field>
    <field name="fullNameRawSource">Вяхорев Александр Александрович</field>
  </party>
</jms:action>
<jms:action type="CLOSE">
  <party type="PHYSICAL" hid="1026"/>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "ВЯХОРЕВ",
              "name": "surname"
            },
            {
              "value": "АЛЕКСАНДР",
              "name": "name"
            },
            {
              "value": "АЛЕКСАНДРОВИЧ",
              "name": "patronymic"
            },
            {
              "value": "EDITED",
              "name": "fullNameQC"
            },
            {
              "value": "AL:1",
              "name": "fullNameAuthor"
            },
            {
              "value": "Вяхорев Александр Александрович",
              "name": "fullNameRawSource"
            }
          ],
          "hid": 1024,
          "sourceSystem": "AL",
          "rawId": "1",
          "attribute": [
            {

```

```

"field": [
  {
    "value": "AL:1",
    "name": "author"
  },
  {
    "value": "HOME",
    "name": "type"
  },
  {
    "value": "7",
    "name": "countryCode"
  },
  {
    "value": "495",
    "name": "cityCode"
  },
  {
    "value": "2171522",
    "name": "number"
  },
  {
    "value": "GOOD",
    "name": "qualityCode"
  },
  {
    "value": "8-495-217-15-22",
    "name": "rawSource"
  }
],
"type": "PHONE",
"hid": 1024,
"rawId": "AL.PH.1",
"deleted": null
}
],
"changes": null,
"relation": null,
"type": "UNMERGE"
},
{
  "attribute": null,
  "party": [
    {
      "field": [
        {
          "value": "ВЯХОРЕВ",
          "name": "surname"
        },
        {
          "value": "АЛЕКСАНДР",
          "name": "name"
        },
        {
          "value": "АЛЕКСАНДРОВИЧ",
          "name": "patronymic"
        },
        {
          "value": "EDITED",
          "name": "fullNameQC"
        }
      ]
    }
  ]
}

```

```
    },
    {
      "value": "BT:1",
      "name": "fullNameAuthor"
    },
    {
      "value": "Вяхорев Александр Александрович",
      "name": "fullNameRawSource"
    }
  ],
  "hid": 1025,
  "sourceSystem": "BT",
  "rawId": "1",
  "attribute": null,
  "type": "PHYSICAL"
}
],
"changes": null,
"relation": null,
"type": "UNMERGE"
},
{
  "attribute": null,
  "party": [
    {
      "hid": 1026,
      "sourceSystem": null,
      "rawId": null,
      "type": "PHYSICAL"
    }
  ]
},
"changes": null,
"relation": null,
"type": "CLOSE"
}
],
"stamp": 1695282318938
}
```



# JMS. СОЗДАНИЕ СВЯЗЕЙ

- Создана связь для контрагента
- Создано несколько связей

## Создана связь для контрагента

Единый клиент отправляет два вида сообщений о создании связей:

1. стандартное, которое содержит информацию о создании связи (используется по умолчанию);
2. расширенное, в котором содержатся идентификатор системы источника и код системы источника контрагента (доступно, начиная с релиза 16.7, подключается по запросу).

Пример сообщений о создании связи между исходной карточкой (hid=1) и золотой карточкой (hid=2):

### XML

```
<jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CREATE_RELATION">
    <relation type="1" hashcode="ac1af1a4cd07cee8f06fdbbee1ab8b820">
      <first type="PHYSICAL" hid="1"/>
      <second type="PHYSICAL" hid="2"/>
    </relation>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "changes": null,
      "relation": {
        "first": {
          "type": "PHYSICAL",
          "hid": 1,
          "rawId": null,
          "sourceSystem": null
        },
        "second": {
          "type": "PHYSICAL",
          "hid": 2,
          "rawId": null,
          "sourceSystem": null
        },
        "type": 1,
        "deleted": null,
        "hashcode": "ac1af1a4cd07cee8f06fdbbee1ab8b820"
      }
    }
  ],
}
```

```

    "type": "CREATE_RELATION"
  }
],
"stamp": 1695313143893
}

```

#### XML

```

<jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CREATE_RELATION">
    <relation type="1" hashcode="ac1af1a4cd07cee8f06fdbee1ab8b820">
      <first type="PHYSICAL" hid="1" rawld="test1" sourceSystem="AL"/>
      <second type="PHYSICAL" hid="2" rawld="test2" sourceSystem="BI"/>
    </relation>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": null,
      "relation": {
        "first": {
          "type": "PHYSICAL",
          "hid": 1,
          "rawld": "test1",
          "sourceSystem": "AL"
        },
        "second": {
          "type": "PHYSICAL",
          "hid": 2,
          "rawld": "test2",
          "sourceSystem": "BI"
        },
        "type": 1,
        "deleted": null,
        "hashcode": "ac1af1a4cd07cee8f06fdbee1ab8b820"
      },
      "type": "CREATE_RELATION"
    }
  ],
  "stamp": 1695313143893
}

```

## Создано несколько связей

Пример сообщений о создании связи между:

1. Карточкой с hid=1 и карточкой с hid=2. Тип связи = 1. Тип КА = МОСК
2. Карточкой с hid=1 и карточкой с hid=2. Тип связи = 2. Тип первого КА = МОСК. Тип второго КА = MOCK\_REQUISITE

### XML

```
<jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CREATE_RELATION">
    <relation type="1" hashCode="17838ae91875438725b881f6ded9bd38">
      <first type="MOCK" hid="1"/>
      <second type="MOCK" hid="2"/>
    </relation>
  </jms:action>
  <jms:action type="CREATE_RELATION">
    <relation type="2" hashCode="6a273354b42695f006be8439fe70250f">
      <first type="MOCK" hid="1"/>
      <second type="MOCK_REQUISITE" hid="1"/>
    </relation>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "changes": null,
      "relation": {
        "first": {
          "type": "MOCK",
          "hid": 1,
          "rawId": null,
          "sourceSystem": null
        },
        "second": {
          "type": "MOCK",
          "hid": 2,
          "rawId": null,
          "sourceSystem": null
        },
        "type": 1,
        "deleted": null,
        "hashCode": "17838ae91875438725b881f6ded9bd38"
      },
      "type": "CREATE_RELATION"
    },
    {
      "attribute": null,
      "changes": null,
      "relation": {
        "first": {
          "type": "MOCK",
          "hid": 1,
          "rawId": null,
          "sourceSystem": null
        },
        "second": {
          "type": "MOCK_REQUISITE",
          "hid": 1,
          "rawId": null,
          "sourceSystem": null
        },
        "type": 2,
        "deleted": null,

```

```
"hashcode": "6a273354b42695f006be8439fe70250f"  
},  
"type": "CREATE_RELATION"  
}  
},  
"stamp": 1695313143893  
}
```

Примечание: В одном сообщении — не более 100 связей.

# JMS. ЗАКРЫТИЕ СВЯЗЕЙ

## *Сообщение о закрытии связи*

Сообщение о закрытии связи между исходной карточкой (hid=1) и золотой карточкой (hid=2):

### XML

```
<jms:event stamp="2013-03-21T14:23:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="CLOSE_RELATION">
    <relation type="1">
      <first type="PHYSICAL" hid="1"/>
      <second type="PHYSICAL" hid="2"/>
    </relation>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "changes": null,
      "relation": {
        "first": {
          "type": "PHYSICAL",
          "hid": 1
        },
        "second": {
          "type": "PHYSICAL",
          "hid": 2
        },
        "type": 1,
        "deleted": null
      },
      "type": "CLOSE_RELATION"
    }
  ],
  "stamp": 1695313143893
}
```

# JMS. ВОССТАНОВЛЕНИЕ НА ВЕРСИЮ

- Восстановился только реквизит Исходное состояние:
- Восстановился только атрибут
- Восстановились и реквизиты, и атрибуты
- Отмена закрытия атрибута
- Отмена создания нового атрибута

Единый клиент отправляет два вида сообщений:

1. стандартное, которое содержит только актуальное состояние карточки (используется по умолчанию);
2. **расширенное**, в котором содержатся детали изменения (доступно с релиза 16.7, подключается по запросу).

Заголовок сообщений — UPDATE, так как при восстановлении на версию клиент обновляется, просто "старыми данными".

## ***Восстановился только реквизит Исходное состояние:***

- Контрагент HID = 1 (AL:1)
- ФИО: Сережа менялся на Колю
- Есть адрес и пара телефонов

Событие:

- Клиента откатили на предыдущую версию. Но изменилось при этом только имя.

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname">МОРОЗОВ</field>
      <field name="name">СЕРЕЖА</field>
      <field name="patronymic">ВИТАЛЬЕВИЧ</field>
      <field name="gender">MALE</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Морозов Сережа Витальевич</field>
      <field name="birthdate">18.03.1975</field>
      <field name="birthdateAuthor">AL:1</field>
      <field name="birthdateQC">EDITED</field>
      <field name="birthdateRawSource">18.03.1975</field>
      <attribute type="ADDRESS" hid="31" rawId="1.ADR.1">
        <field name="qualityCode">GOOD</field>
        <field name="validationCode">VALIDATED</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>
```

```

<field name="type">REGISTRATION</field>
<field name="kladrActualityDate"/>
<field name="postalCode">630124</field>
<field name="kladrPostalCode">630124</field>
<field name="country">РОССИЯ</field>
<field name="district"/>
<field name="regionType">ОБЛ</field>
<field name="region">НОВОСИБИРСКАЯ</field>
<field name="rayonType"/>
<field name="rayon"/>
<field name="cityType">Г</field>
<field name="city">НОВОСИБИРСК</field>
<field name="settlementType"/>
<field name="settlement"/>
<field name="streetType">УЛ</field>
<field name="street">ЕСЕНИНА</field>
<field name="houseNumber">59</field>
<field name="militaryNumber"/>
<field name="postalBox"/>
<field name="korporus"/>
<field name="stroenie"/>
<field name="vldenie"/>
<field name="hostel"/>
<field name="floor"/>
<field name="entrance"/>
<field name="section"/>
<field name="flat">6</field>
<field name="flat2"/>
<field name="office"/>
<field name="office2"/>
<field name="room"/>
<field name="room2"/>
<field name="kladrCode"/>
<field name="okatoCode"/>
<field name="validationExtension"/>
<field name="rawSource">RU,630124,Новосибирская область,,Новосибирск г,,
Улица Есенина,59,,4</field>
<field name="primary">>true</field>
<field name="additionalInfo"/>
<field name="author">performer</field>
</attribute>
<attribute type="PHONE" hid="11" rawId="1.PH.1">
<field name="qualityCode">GOOD</field>
<field name="type">HOME</field>
<field name="timezone">MISSED</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="extension"/>
<field name="rawSource">4952171522</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="author">performer</field>
</attribute>
<attribute type="PHONE" hid="12" rawId="1.PH.2">
<field name="qualityCode">GOOD</field>
<field name="type">WORK</field>
<field name="timezone">MISSED</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="extension"/>
<field name="rawSource">4952171533</field>
<field name="primary">>false</field>

```

```

    <field name="additionalInfo"/>
    <field name="author">AL:1</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "МОПО3ОБ",
              "name": "surname"
            },
            ...
            {
              "value": "18.03.1975",
              "name": "birthdateRawSource"
            }
          ],
          "attribute": [
            {
              "field": [
                {
                  "value": "GOOD",
                  "name": "qualityCode"
                },
                ...
                {
                  "value": "performer",
                  "name": "author"
                }
              ]
            },
            "type": "ADDRESS",
            "hid": 31,
            "rawId": "1.ADR.1"
          ]
        },
        ...
        {
          "field": [
            {
              "value": "GOOD",
              "name": "qualityCode"
            },
            ...
            {
              "value": "AL:1",
              "name": "author"
            }
          ],
          "type": "PHONE",
          "hid": 12,
          "rawId": "1.PH.2"
        }
      ],
      "hid": 1,

```



```

    "sourceSystem": "AL",
    "rawId": "1",
    "type": "PHYSICAL"
  }
],
"changes": null,
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change essence="PHYSICAL" hid="1" rawId="1" sourceSystem="AL" type="
UPDATE">
        <jms:detail fieldName="fullNameQC">
          <jms:oldValue>CONFIRMED_MANUALLY</jms:oldValue>
          <jms:newValue>EDITED</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="name">
          <jms:oldValue>КОЛЯ</jms:oldValue>
          <jms:newValue>СЕРЕЖА</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
    <party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname">МОРОЗОВ</field>
      <field name="name">СЕРЕЖА</field>
      <field name="patronymic">ВИТАЛЬЕВИЧ</field>
      <field name="gender">MALE</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Морозов Сережа Витальевич</field>
      <field name="birthdate">18.03.1975</field>
      <field name="birthdateAuthor">AL:1</field>
      <field name="birthdateQC">EDITED</field>
      <field name="birthdateRawSource">18.03.1975</field>
      <attribute type="ADDRESS" hid="31" rawId="1.ADR.1">
        <field name="qualityCode">GOOD</field>
        <field name="validationCode">VALIDATED</field>
        <field name="type">REGISTRATION</field>
        <field name="kladrActualityDate"/>
        <field name="postalCode">630124</field>
        <field name="kladrPostalCode">630124</field>
        <field name="country">РОССИЯ</field>
        <field name="district"/>
        <field name="regionType">ОБЛ</field>
        <field name="region">НОВОСИБИРСКАЯ</field>
        <field name="rayonType"/>
        <field name="rayon"/>
        <field name="cityType">Г</field>
        <field name="city">НОВОСИБИРСК</field>
        <field name="settlementType"/>
        <field name="settlement"/>
        <field name="streetType">УЛ</field>
        <field name="street">ЕСЕНИНА</field>
        <field name="houseNumber">59</field>
      </attribute>
    </party>
  </jms:event>

```

```

<field name="militaryNumber"/>
<field name="postalBox"/>
<field name="korpus"/>
<field name="stroenie"/>
<field name="vladenie"/>
<field name="hostel"/>
<field name="floor"/>
<field name="entrance"/>
<field name="section"/>
<field name="flat">6</field>
<field name="flat2"/>
<field name="office"/>
<field name="office2"/>
<field name="room"/>
<field name="room2"/>
<field name="kladrCode"/>
<field name="okatoCode"/>
<field name="validationExtension"/>
<field name="rawSource">RU,630124,Новосибирская область,,Новосибирск г,,
Улица Есенина,59,,,4</field>
<field name="primary">true</field>
<field name="additionalInfo"/>
<field name="author">performer</field>
</attribute>
<attribute type="PHONE" hid="11" rawId="1.PH.1">
<field name="qualityCode">GOOD</field>
<field name="type">HOME</field>
<field name="timezone">MISSED</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="extension"/>
<field name="rawSource">4952171522</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="author">performer</field>
</attribute>
<attribute type="PHONE" hid="12" rawId="1.PH.2">
<field name="qualityCode">GOOD</field>
<field name="type">WORK</field>
<field name="timezone">MISSED</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="extension"/>
<field name="rawSource">4952171533</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="author">AL:1</field>
</attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": {
        "change": [
          {
            "detail": [

```

```

    {
      "oldValue": "CONFIRMED_MANUALLY",
      "newValue": "EDITED",
      "fieldName": "fullNameQC"
    },
    {
      "oldValue": "КОЛЯ",
      "newValue": "СЕРЕЖА",
      "fieldName": "name"
    }
  ],
  "type": "UPDATE",
  "essence": "PHYSICAL",
  "hid": 1,
  "rawId": "1",
  "sourceSystem": "AL"
}
]
},
"party": [
  {
    "field": [
      {
        "value": "МОРОЗОВ",
        "name": "surname"
      },
      ...
      {
        "value": "18.03.1975",
        "name": "birthdateRawSource"
      }
    ],
    "attribute": [
      {
        "field": [
          {
            "value": "GOOD",
            "name": "qualityCode"
          },
          ...
          {
            "value": "performer",
            "name": "author"
          }
        ]
      },
      "type": "ADDRESS",
      "hid": 31,
      "rawId": "1.ADR.1"
    ]
  },
  ...
  {
    "field": [
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      ...
      {
        "value": "AL:1",
        "name": "author"
      }
    ]
  },
  ],

```

```

    "type": "PHONE",
    "hid": 12,
    "rawId": "1.PH.2"
  }
],
"hid": 1,
"sourceSystem": "AL",
"rawId": "1",
"type": "PHYSICAL"
}
],
"changes": null,
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

## Восстановился только атрибут

Исходное состояние:

- Контрагент HID = 1 (AL:1)
- Есть пара телефонов

Событие:

- Клиента восстановили на предыдущую версию. Изменился только тип одного из телефонов
- Восстановили не всего клиента, а только телефон. Событие будет точно такое же

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname">МОРОЗОВ</field>
      <field name="name">СЕРЕЖА</field>
      <field name="patronymic">ВИТАЛЬЕВИЧ</field>
      <field name="gender">MALE</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Морозов Сережа Витальевич</field>
      <field name="birthdate">18.03.1975</field>
      <field name="birthdateAuthor">AL:1</field>
      <field name="birthdateQC">EDITED</field>
      <field name="birthdateRawSource">18.03.1975</field>
      <attribute type="PHONE" hid="11" rawId="1.PH.1">
        <field name="qualityCode">GOOD</field>
        <field name="type">HOME</field>
        <field name="timezone">MISSED</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="extension"/>
        <field name="rawSource">4952171522</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>

```

```

    <field name="primary">false</field>
    <field name="additionalInfo"/>
    <field name="author">performer</field>
  </attribute>
  <attribute type="PHONE" hid="12" rawId="1.PH.2">
    <field name="qualityCode">GOOD</field>
    <field name="type">WORK</field>
    <field name="timezone">MISSED</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="extension"/>
    <field name="rawSource">4952171533</field>
    <field name="primary">false</field>
    <field name="additionalInfo"/>
    <field name="author">AL:1</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "МОПО30В",
              "name": "surname"
            },
            ...
            {
              "value": "18.03.1975",
              "name": "birthdateRawSource"
            }
          ],
          "attribute": [
            {
              "field": [
                {
                  "value": "GOOD",
                  "name": "qualityCode"
                },
                ...
                {
                  "value": "performer",
                  "name": "author"
                }
              ]
            },
            "type": "PHONE",
            "hid": 11,
            "rawId": "1.PH.1"
          ]
        },
        {
          "field": [
            {
              "value": "GOOD",
              "name": "qualityCode"
            },
            ...
          ]
        }
      ]
    }
  ]
}

```

```

...
    {
      "value": "AL:1",
      "name": "author"
    }
  ],
  "type": "PHONE",
  "hid": 12,
  "rawId": "1.PH.2"
}
],
"hid": 1,
"sourceSystem": "AL",
"rawId": "1",
"type": "PHYSICAL"
}
],
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change essence="PHONE" hid="11" rawId="1.PH.1" type="UPDATE">
        <jms:detail fieldName="type">
          <jms:oldValue>OTHER</jms:oldValue>
          <jms:newValue>WORK</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
    <party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname">МОРОЗОВ</field>
      <field name="name">СЕРЕЖА</field>
      <field name="patronymic">ВИТАЛЬЕВИЧ</field>
      <field name="gender">MALE</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Морозов Сережа Витальевич</field>
      <field name="birthdate">18.03.1975</field>
      <field name="birthdateAuthor">AL:1</field>
      <field name="birthdateQC">EDITED</field>
      <field name="birthdateRawSource">18.03.1975</field>
      <attribute type="PHONE" hid="11" rawId="1.PH.1">
        <field name="qualityCode">GOOD</field>
        <field name="type">HOME</field>
        <field name="timezone">MISSED</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="extension"/>
        <field name="rawSource">4952171522</field>
        <field name="primary">>false</field>
        <field name="additionalInfo"/>
        <field name="author">performer</field>
      </attribute>
      <attribute type="PHONE" hid="12" rawId="1.PH.2">
        <field name="qualityCode">GOOD</field>

```

```

    <field name="type">WORK</field>
    <field name="timezone">MISSED</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="extension"/>
    <field name="rawSource">4952171533</field>
    <field name="primary">>false</field>
    <field name="additionalInfo"/>
    <field name="author">AL:1</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

## JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": {
        "change": [
          {
            "detail": [
              {
                "oldValue": "OTHER",
                "newValue": "WORK",
                "fieldName": "type"
              }
            ],
            "type": "UPDATE",
            "essence": "PHONE",
            "hid": 11,
            "rawId": "1.PH.",
            "sourceSystem": null
          }
        ]
      },
      "party": [
        {
          "field": [
            {
              "value": "МОПОЗОВ",
              "name": "surname"
            },
            ...
            {
              "value": "18.03.1975",
              "name": "birthdateRawSource"
            }
          ],
          "attribute": [
            {
              "field": [
                {
                  "value": "GOOD",
                  "name": "qualityCode"
                },
                ...
                {
                  "value": "performer",
                  "name": "author"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}

```

```

    }
  ],
  "type": "PHONE",
  "hid": 11,
  "rawId": "1.PH.1"
}
{
  "field": [
    {
      "value": "GOOD",
      "name": "qualityCode"
    },
    ...
    {
      "value": "AL:1",
      "name": "author"
    }
  ],
  "type": "PHONE",
  "hid": 12,
  "rawId": "1.PH.2"
}
],
"hid": 1,
"sourceSystem": "AL",
"rawId": "1",
"type": "PHYSICAL"
}
],
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

## Восстановились и реквизиты, и атрибуты

Событие:

- Клиента откатили на предыдущую версию. Изменились и реквизиты, и атрибуты

### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname">МОРОЗОВ</field>
      <field name="name">СЕРЕЖА</field>
      <field name="patronymic">ВИТАЛЬЕВИЧ</field>
      <field name="gender">MALE</field>
      <field name="fullNameQC">EDITED</field>
      <field name="fullNameAuthor">AL:1</field>
      <field name="fullNameRawSource">Морозов Сережа Витальевич</field>
      <field name="birthdate">18.03.1975</field>
      <field name="birthdateAuthor">AL:1</field>
      <field name="birthdateQC">EDITED</field>
      <field name="birthdateRawSource">18.03.1975</field>
    </party>
  </jms:action>
</jms:event>

```



```

<attribute type="DOCUMENT_PASSPORT" hid="21" rawId="1.DOC.1">
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="issueDate"/>
  <field name="issueAuthority"/>
  <field name="departmentCode"/>
  <field name="expiryDate"/>
  <field name="qualityCode">NOT_SURE</field>
  <field name="type">PASSPORT</field>
  <field name="documentSeries">45 07</field>
  <field name="documentNumber">631945</field>
  <field name="rawSource">Паспорт РФ: 4507 631945</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
<attribute type="DOCUMENT_PASSPORT" hid="22" rawId="1.DOC.2">
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="issueDate">13.08.2007</field>
  <field name="issueAuthority">ОВД Бибирево гор. Москвы</field>
  <field name="departmentCode">771-819</field>
  <field name="expiryDate"/>
  <field name="qualityCode">EDITED</field>
  <field name="type">PASSPORT</field>
  <field name="documentSeries">45 07</field>
  <field name="documentNumber">631945</field>
  <field name="rawSource">Паспорт РФ: 4507 631945, выдан 13.08.2007 ОВД
Бибирево гор. Москвы</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
<attribute type="ADDRESS" hid="31" rawId="1.ADR.1">
  <field name="qualityCode">GOOD</field>
  <field name="validationCode">VALIDATED</field>
  <field name="type">REGISTRATION</field>
  <field name="kladrActualityDate"/>
  <field name="postalCode">630124</field>
  <field name="kladrPostalCode">630124</field>
  <field name="country">РОССИЯ</field>
  <field name="district"/>
  <field name="regionType">ОБЛ</field>
  <field name="region">НОВОСИБИРСКАЯ</field>
  <field name="rayonType"/>
  <field name="rayon"/>
  <field name="cityType">Г</field>
  <field name="city">НОВОСИБИРСК</field>
  <field name="settlementType"/>
  <field name="settlement"/>
  <field name="streetType">УЛ</field>
  <field name="street">ЕСЕНИНА</field>
  <field name="houseNumber">59</field>
  <field name="militaryNumber"/>
  <field name="postalBox"/>
  <field name="korporus"/>
  <field name="stroenie"/>
  <field name="vladenie"/>
  <field name="hostel"/>
  <field name="floor"/>
  <field name="entrance"/>
  <field name="section"/>
  <field name="flat">4</field>
  <field name="flat2"/>

```

```

<field name="office"/>
<field name="office2"/>
<field name="room"/>
<field name="room2"/>
<field name="kladrCode"/>
<field name="okatoCode"/>
<field name="validationExtension"/>
<field name="rawSource">RU,630124,Новосибирская область,,Новосибирск г,,
Улица Есенина,59,,,4</field>
<field name="primary">true</field>
<field name="additionalInfo"/>
<field name="author">AL:1</field>
</attribute>
<attribute type="ADDRESS" hid="32" rawId="1.ADR.2">
<field name="qualityCode">GOOD</field>
<field name="validationCode">VALIDATED</field>
<field name="type">HOME</field>
<field name="kladrActualityDate"/>
<field name="postalCode">630124</field>
<field name="kladrPostalCode">630124</field>
<field name="country">РОССИЯ</field>
<field name="district"/>
<field name="regionType">ОБЛ</field>
<field name="region">НОВОСИБИРСКАЯ</field>
<field name="rayonType"/>
<field name="rayon"/>
<field name="cityType">Г</field>
<field name="city">НОВОСИБИРСК</field>
<field name="settlementType"/>
<field name="settlement"/>
<field name="streetType">УЛ</field>
<field name="street">ЕСЕНИНА</field>
<field name="houseNumber">59</field>
<field name="militaryNumber"/>
<field name="postalBox"/>
<field name="korpus"/>
<field name="stroenie"/>
<field name="vladenie"/>
<field name="hostel"/>
<field name="floor"/>
<field name="entrance"/>
<field name="section"/>
<field name="flat">4</field>
<field name="flat2"/>
<field name="office"/>
<field name="office2"/>
<field name="room"/>
<field name="room2"/>
<field name="kladrCode"/>
<field name="okatoCode"/>
<field name="validationExtension"/>
<field name="rawSource">RU,630124,Новосибирская область,,Новосибирск г,,
Улица Есенина,59,,,4</field>
<field name="primary">true</field>
<field name="additionalInfo"/>
<field name="author">AL:1</field>
</attribute>
<attribute type="PHONE" hid="11" rawId="1.PH.1">
<field name="qualityCode">GARBAGE</field>
<field name="type">HOME</field>
<field name="operator"/>
<field name="operatorRegion"/>

```

```

<field name="operatorAddress"/>
<field name="timezone">MISSED</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="number">2171522</field>
<field name="extension"/>
<field name="numberProfile">UNKNOWN</field>
<field name="rawSource">4952171522 ---</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="author">AL:1</field>
</attribute>
<attribute type="PHONE" hid="12" rawId="1.PH.2">
  <field name="qualityCode">GOOD</field>
  <field name="type">WORK</field>
  <field name="operator"/>
  <field name="operatorRegion"/>
  <field name="operatorAddress"/>
  <field name="timezone">MISSED</field>
  <field name="countryCode">7</field>
  <field name="cityCode">495</field>
  <field name="number">2171533</field>
  <field name="extension"/>
  <field name="numberProfile">UNKNOWN</field>
  <field name="rawSource">4952171533</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "МОПО3ОБ",
              "name": "surname"
            },
            ...
            {
              "value": "18.03.1975",
              "name": "birthdateRawSource"
            }
          ],
          "attribute": [
            {
              "field": [
                {
                  "value": "PASSPORT_RU",
                  "name": "qualifierType"
                },
                ...
                {
                  "value": "AL:1",

```

```

        "name": "author"
      }
    ],
    "type": "DOCUMENT_PASSPORT",
    "hid": 21,
    "rawId": "1.DOC.1"
  }
  ...,
  {
    "field": [
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      ...
      {
        "value": "AL:1",
        "name": "author"
      }
    ],
    "type": "PHONE",
    "hid": 12,
    "rawId": "1.PH.2"
  }
],
"hid": 12,
"sourceSystem": null,
"rawId": "1",
"type": "PHYSICAL"
}
],
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change type="UPDATE" essence="PHYSICAL" hid="1" rawId="1"
sourceSystem="AL">
        <jms:detail fieldName="name">
          <jms:oldValue>КОЛЯ</jms:oldValue>
          <jms:newValue>СЕРЕЖА</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="fullNameQC">
          <jms:oldValue>CONFIRMED_MANUALLY</jms:oldValue>
          <jms:newValue>EDITED</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="birthdateQC">
          <jms:oldValue>CONFIRMED_MANUALLY</jms:oldValue>
          <jms:newValue>EDITED</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="marketingCategory">
          <jms:oldValue>UNKNOWN</jms:oldValue>
          <jms:newValue>VIP</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
  </jms:action>
</jms:event>

```

```

<jms:change type="UPDATE" essence="ADDRESS" hid="31" rawId="1.ADR.1">
  <jms:detail fieldName="flat">
    <jms:oldValue>6</jms:oldValue>
    <jms:newValue>4</jms:newValue>
  </jms:detail>
</jms:change>
<jms:change type="UPDATE" essence="PHONE" hid="11" rawId="1.PH.1">
  <jms:detail fieldName="rawSource">
    <jms:oldValue>4952171522</jms:oldValue>
    <jms:newValue>4952171522 ---</jms:newValue>
  </jms:detail>
  <jms:detail fieldName="qualityCode">
    <jms:oldValue>GOOD</jms:oldValue>
    <jms:newValue>GARBAGE</jms:newValue>
  </jms:detail>
</jms:change>
<jms:change type="UPDATE" essence="DOCUMENT_PASSPORT" hid="21" rawId="1.
DOC.1">
  <jms:detail fieldName="rawSource">
    <jms:oldValue>Паспорт РФ: 4507 631945, выдан 13.08.2007 ОБД Бибирево
гор. Москвы, код 771-819</jms:oldValue>
    <jms:newValue>Паспорт РФ: 4507 631945</jms:newValue>
  </jms:detail>
  <jms:detail fieldName="issueDate">
    <jms:oldValue>13.08.2007</jms:oldValue>
    <jms:newValue></jms:newValue>
  </jms:detail>
  <jms:detail fieldName="issueAuthority">
    <jms:oldValue>ОБД Бибирево гор. Москвы</jms:oldValue>
    <jms:newValue></jms:newValue>
  </jms:detail>
  <jms:detail fieldName="qualityCode">
    <jms:oldValue>EDITED</jms:oldValue>
    <jms:newValue>NOT_SURE</jms:newValue>
  </jms:detail>
</jms:change>
</jms:changes>
<party type="PHYSICAL" hid="1" sourceSystem="AL" rawId="1">
  <field name="surname">МОРОЗОВ</field>
  <field name="name">СЕРЕЖА</field>
  <field name="patronymic">ВИТАЛЬЕВИЧ</field>
  <field name="gender">MALE</field>
  <field name="fullNameQC">EDITED</field>
  <field name="fullNameAuthor">AL:1</field>
  <field name="fullNameRawSource">Морозов Сережа Витальевич</field>
  <field name="birthdate">18.03.1975</field>
  <field name="birthdateAuthor">AL:1</field>
  <field name="birthdateQC">EDITED</field>
  <field name="birthdateRawSource">18.03.1975</field>
  <attribute type="DOCUMENT_PASSPORT" hid="21" rawId="1.DOC.1">
    <field name="qualifierType">PASSPORT_RU</field>
    <field name="issueDate"/>
    <field name="issueAuthority"/>
    <field name="departmentCode"/>
    <field name="expiryDate"/>
    <field name="qualityCode">NOT_SURE</field>
    <field name="type">PASSPORT</field>
    <field name="documentSeries">45 07</field>
    <field name="documentNumber">631945</field>
    <field name="rawSource">Паспорт РФ: 4507 631945</field>
    <field name="primary">>false</field>
    <field name="additionalInfo"/>

```

```
<field name="author">AL:1</field>
</attribute>
<attribute type="DOCUMENT_PASSPORT" hid="22" rawId="1.DOC.2">
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="issueDate">13.08.2007</field>
  <field name="issueAuthority">ОВД Бибирево гор. Москвы</field>
  <field name="departmentCode">771-819</field>
  <field name="expiryDate"/>
  <field name="qualityCode">EDITED</field>
  <field name="type">PASSPORT</field>
  <field name="documentSeries">45 07</field>
  <field name="documentNumber">631945</field>
  <field name="rawSource">Паспорт РФ: 4507 631945, выдан 13.08.2007 ОВД
Бибирево гор. Москвы</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
<attribute type="ADDRESS" hid="31" rawId="1.ADR.1">
  <field name="qualityCode">GOOD</field>
  <field name="validationCode">VALIDATED</field>
  <field name="type">REGISTRATION</field>
  <field name="kladrActualityDate"/>
  <field name="postalCode">630124</field>
  <field name="kladrPostalCode">630124</field>
  <field name="country">РОССИЯ</field>
  <field name="district"/>
  <field name="regionType">ОБЛ</field>
  <field name="region">НОВОСИБИРСКАЯ</field>
  <field name="rayonType"/>
  <field name="rayon"/>
  <field name="cityType">Г</field>
  <field name="city">НОВОСИБИРСК</field>
  <field name="settlementType"/>
  <field name="settlement"/>
  <field name="streetType">УЛ</field>
  <field name="street">ЕСЕНИНА</field>
  <field name="houseNumber">59</field>
  <field name="militaryNumber"/>
  <field name="postalBox"/>
  <field name="korporus"/>
  <field name="stroenie"/>
  <field name="vladenie"/>
  <field name="hostel"/>
  <field name="floor"/>
  <field name="entrance"/>
  <field name="section"/>
  <field name="flat">4</field>
  <field name="flat2"/>
  <field name="office"/>
  <field name="office2"/>
  <field name="room"/>
  <field name="room2"/>
  <field name="kladrCode"/>
  <field name="okatoCode"/>
  <field name="validationExtension"/>
  <field name="rawSource">RU,630124,Новосибирская область,,Новосибирск г,,
Улица Есенина,59,,,4</field>
  <field name="primary">>true</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
```

```

<attribute type="ADDRESS" hid="32" rawId="1.ADR.2">
  <field name="qualityCode">GOOD</field>
  <field name="validationCode">VALIDATED</field>
  <field name="type">HOME</field>
  <field name="kladrActualityDate"/>
  <field name="postalCode">630124</field>
  <field name="kladrPostalCode">630124</field>
  <field name="country">РОССИЯ</field>
  <field name="district"/>
  <field name="regionType">ОБЛ</field>
  <field name="region">НОВОСИБИРСКАЯ</field>
  <field name="rayonType"/>
  <field name="rayon"/>
  <field name="cityType">Г</field>
  <field name="city">НОВОСИБИРСК</field>
  <field name="settlementType"/>
  <field name="settlement"/>
  <field name="streetType">УЛ</field>
  <field name="street">ЕСЕНИНА</field>
  <field name="houseNumber">59</field>
  <field name="militaryNumber"/>
  <field name="postalBox"/>
  <field name="korporus"/>
  <field name="stroenie"/>
  <field name="vladenie"/>
  <field name="hostel"/>
  <field name="floor"/>
  <field name="entrance"/>
  <field name="section"/>
  <field name="flat">4</field>
  <field name="flat2"/>
  <field name="office"/>
  <field name="office2"/>
  <field name="room"/>
  <field name="room2"/>
  <field name="kladrCode"/>
  <field name="okatoCode"/>
  <field name="validationExtension"/>
  <field name="rawSource">RU,630124,Новосибирская область,,Новосибирск г,,
Улица Есенина,59,,,4</field>
  <field name="primary">true</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
<attribute type="PHONE" hid="11" rawId="1.PH.1">
  <field name="qualityCode">GARBAGE</field>
  <field name="type">HOME</field>
  <field name="operator"/>
  <field name="operatorRegion"/>
  <field name="operatorAddress"/>
  <field name="timezone">MISSED</field>
  <field name="countryCode">7</field>
  <field name="cityCode">495</field>
  <field name="number">2171522</field>
  <field name="extension"/>
  <field name="numberProfile">UNKNOWN</field>
  <field name="rawSource">4952171522 ---</field>
  <field name="primary">false</field>
  <field name="additionalInfo"/>
  <field name="author">AL:1</field>
</attribute>
<attribute type="PHONE" hid="12" rawId="1.PH.2">

```

```

<field name="qualityCode">GOOD</field>
<field name="type">WORK</field>
<field name="operator"/>
<field name="operatorRegion"/>
<field name="operatorAddress"/>
<field name="timezone">MISSED</field>
<field name="countryCode">7</field>
<field name="cityCode">495</field>
<field name="number">2171533</field>
<field name="extension"/>
<field name="numberProfile">UNKNOWN</field>
<field name="rawSource">4952171533</field>
<field name="primary">>false</field>
<field name="additionalInfo"/>
<field name="author">AL:1</field>
</attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": {
        "change": [
          {
            "detail": [
              {
                "oldValue": "КОЛЯ",
                "newValue": "СЕРЕЖА",
                "fieldName": "name"
              },
              ...
            ],
            "type": "UPDATE",
            "essence": "PHYSICAL",
            "hid": 11,
            "rawId": "1",
            "sourceSystem": "AL"
          },
          ...
        ]
      },
      "detail": [
        {
          "oldValue": "Паспорт РФ: 4507 631945, выдан 13.08.2007 ОВД Бибирево гор.
Москвы, код 771-819",
          "newValue": "Паспорт РФ: 4507 631945",
          "fieldName": "rawSource"
        },
        ...
      ],
      "type": "UPDATE",
      "essence": "DOCUMENT_PASSPORT",
      "hid": 21,
      "rawId": "1.DOC.1",
      "sourceSystem": null
    }
  ]
},

```



```

"party": [
  {
    "field": [
      {
        "value": "МОПО30В",
        "name": "surname"
      },
      ...
      {
        "value": "18.03.1975",
        "name": "birthdateRawSource"
      }
    ],
    "attribute": [
      {
        "field": [
          {
            "value": "PASSPORT_RU",
            "name": "qualifierType"
          },
          ...
          {
            "value": "AL:1",
            "name": "author"
          }
        ],
        "type": "DOCUMENT_PASSPORT",
        "hid": 21,
        "rawId": "1.DOC.1"
      }
    ],
    ...
    {
      "field": [
        {
          "value": "GOOD",
          "name": "qualityCode"
        },
        ...
        {
          "value": "AL:1",
          "name": "author"
        }
      ],
      "type": "PHONE",
      "hid": 12,
      "rawId": "1.PH.2"
    }
  ],
  "hid": 1,
  "sourceSystem": "AL",
  "rawId": "1",
  "type": "PHYSICAL"
}
],
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

## Отмена закрытия атрибута

Исходное состояние:

- Контрагент HID = 1 (AL:1)
- Телефон удален

Событие:

- Клиента восстановили на предыдущую версию, где телефон еще существовал

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <party type="MOCK" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname"/>
      <field name="name"/>
      <field name="patronymic"/>
      <field name="gender">UNKNOWN</field>
      <field name="fullNameQC">MISSED</field>
      <field name="fullNameAuthor"/>
      <field name="fullNameRawSource"/>
      <field name="birthdate"/>
      <field name="birthdateAuthor"/>
      <field name="birthdateQC">MISSED</field>
      <field name="birthdateRawSource"/>
      <attribute type="PHONE" hid="1" rawId="AL.PH.1">
        <field name="qualityCode">GOOD</field>
        <field name="type">HOME</field>
        <field name="timezone">MISSED</field>
        <field name="countryCode">7</field>
        <field name="cityCode">495</field>
        <field name="extension"/>
        <field name="rawSource">4952171522</field>
        <field name="primary">>false</field>
        <field name="additionalInfo"/>
        <field name="author">AL:1</field>
      </attribute>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "",
              "name": "surname"
            },
            {
```

```

...
{
  "value": "",
  "name": "birthdateRawSource"
}
],
"attribute": [
  {
    "field": [
      {
        "value": "GOOD",
        "name": "qualityCode"
      },
      ...
      {
        "value": "AL:1",
        "name": "author"
      }
    ],
    "type": "PHONE",
    "hid": 1,
    "rawId": "AL.PH.1"
  },
  ...
],
"hid": 1,
"sourceSystem": "AL",
"rawId": "1",
"type": "MOCK"
}
],
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change essence="PHONE" hid="1" rawId="AL.PH.1" type="UPDATE">
        <jms:detail fieldName="cityCode">
          <jms:oldValue/>
          <jms:newValue>495</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="countryCode">
          <jms:oldValue/>
          <jms:newValue>7</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="mergeAuthor">
          <jms:oldValue/>
          <jms:newValue>AL:1</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="number">
          <jms:oldValue/>
          <jms:newValue>2171522</jms:newValue>
        </jms:detail>
        <jms:detail fieldName="numberProfile">
          <jms:oldValue/>
          <jms:newValue>UNKNOWN</jms:newValue>
        </jms:detail>
      </jms:change>
    </jms:changes>
  </jms:action>
</jms:event>

```

```

</jms:detail>
<jms:detail fieldName="primary">
  <jms:oldValue/>
  <jms:newValue>>false</jms:newValue>
</jms:detail>
<jms:detail fieldName="qualityCode">
  <jms:oldValue/>
  <jms:newValue>GOOD</jms:newValue>
</jms:detail>
<jms:detail fieldName="rawId">
  <jms:oldValue/>
  <jms:newValue>AL.PH.1</jms:newValue>
</jms:detail>
<jms:detail fieldName="rawSource">
  <jms:oldValue/>
  <jms:newValue>4952171522</jms:newValue>
</jms:detail>
<jms:detail fieldName="timezone">
  <jms:oldValue/>
  <jms:newValue>MISSED</jms:newValue>
</jms:detail>
<jms:detail fieldName="type">
  <jms:oldValue/>
  <jms:newValue>HOME</jms:newValue>
</jms:detail>
</jms:change>
</jms:changes>
<party type="MOCK" hid="1" sourceSystem="AL" rawId="1">
  <field name="surname"/>
  <field name="name"/>
  <field name="patronymic"/>
  <field name="gender">UNKNOWN</field>
  <field name="fullNameQC">MISSED</field>
  <field name="fullNameAuthor"/>
  <field name="fullNameRawSource"/>
  <field name="birthdate"/>
  <field name="birthdateAuthor"/>
  <field name="birthdateQC">MISSED</field>
  <field name="birthdateRawSource"/>
  <attribute type="PHONE" hid="1" rawId="AL.PH.1">
    <field name="qualityCode">GOOD</field>
    <field name="type">HOME</field>
    <field name="timezone">MISSED</field>
    <field name="countryCode">7</field>
    <field name="cityCode">495</field>
    <field name="extension"/>
    <field name="rawSource">4952171522</field>
    <field name="primary">>false</field>
    <field name="additionalInfo"/>
    <field name="author">AL:1</field>
  </attribute>
</party>
</jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": {

```

```

"change": [
  {
    "detail": [
      {
        "oldValue": "",
        "newValue": "495",
        "fieldName": "cityCode"
      },
      ...
    ],
    "type": "UPDATE",
    "essence": "PHONE",
    "hid": 1,
    "rawId": "AL.PH.1",
    "sourceSystem": null
  }
],
"party": [
  {
    "field": [
      {
        "value": "",
        "name": "surname"
      },
      ...
      {
        "value": "",
        "name": "birthdateRawSource"
      }
    ],
    "attribute": [
      {
        "field": [
          {
            "value": "GOOD",
            "name": "qualityCode"
          },
          ...
          {
            "value": "AL:1",
            "name": "author"
          }
        ],
        "type": "PHONE",
        "hid": 1,
        "rawId": "AL.PH.1"
      }
    ],
    "hid": 1,
    "sourceSystem": "AL",
    "rawId": "1",
    "type": "MOCK"
  }
],
"stamp": 1695282318938
}

```

## Отмена создания нового атрибута

Исходное состояние:

- Контрагент HID = 1 (AL:1)
- Создан телефон

Событие:

- Клиента восстановили на предыдущую версию, где телефона еще не было.

### XML

```
<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change essence="PHONE" hid="1" rawId="HFLabs-CDI.1" type="CLOSE"/>
    </jms:changes>
    <party type="MOCK" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname"/>
      <field name="name"/>
      <field name="patronymic"/>
      <field name="gender">UNKNOWN</field>
      <field name="fullNameQC">MISSED</field>
      <field name="fullNameAuthor"/>
      <field name="fullNameRawSource"/>
      <field name="birthdate"/>
      <field name="birthdateAuthor"/>
      <field name="birthdateQC">MISSED</field>
      <field name="birthdateRawSource"/>
    </party>
  </jms:action>
</jms:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "changes": {
        "change": [
          {
            "type": "CLOSE",
            "essence": "PHONE",
            "hid": 1,
            "rawId": "HFLabs-CDI.1",
            "sourceSystem": null
          }
        ]
      }
    }
  ],
  "party": [
    {
      "field": [
        {
          "value": "",

```

```

        "name": "surname"
    },
    ...
    {
        "value": "",
        "name": "birthdateRawSource"
    }
],
"hid": 1,
"sourceSystem": "AL",
"rawId": "1",
"type": "MOCK"
}
],
"type": "UPDATE"
}
],
"stamp": 1695282318938
}

```

#### XML

```

<jms:event stamp="2013-03-22T10:05:00" xmlns:jms="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <jms:action type="UPDATE">
    <jms:changes>
      <jms:change essence="PHONE" hid="1" rawId="HFLabs-CDI.1" type="CLOSE"/>
    </jms:changes>
    <party type="MOCK" hid="1" sourceSystem="AL" rawId="1">
      <field name="surname"/>
      <field name="name"/>
      <field name="patronymic"/>
      <field name="gender">UNKNOWN</field>
      <field name="fullNameQC">MISSED</field>
      <field name="fullNameAuthor"/>
      <field name="fullNameRawSource"/>
      <field name="birthdate"/>
      <field name="birthdateAuthor"/>
      <field name="birthdateQC">MISSED</field>
      <field name="birthdateRawSource"/>
    </party>
  </jms:action>
</jms:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "changes": {
        "change": [
          {
            "type": "CLOSE",
            "essence": "PHONE",
            "hid": 1,
            "rawId": "HFLabs-CDI.1",
            "sourceSystem": null
          }
        ]
      }
    }
  ],
}

```

```
"party": [
  {
    "field": [
      {
        "value": "",
        "name": "surname"
      },
      ...
      {
        "value": "",
        "name": "birthdateRawSource"
      }
    ],
    "hid": 1,
    "sourceSystem": "AL",
    "rawId": "1",
    "type": "MOCK"
  },
  ],
  "type": "UPDATE"
}
],
"stamp": 1695282318938
}
```



# JMS. ПРОВЕРКА ПО ЧЕРНЫМ СПИСКАМ

- Попадание в черный список
- Выход из черного списка

## Попадание в черный список

Исходное состояние:

- Контрагент не совпадает ни с одним черным списком.

Событие:

- Контрагент найден в черном списке экстремистов ЦБ и черном списке ФЛ.

```
XML
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:event stamp="2015-04-14T17:42:36" xmlns:ns2="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <ns2:action type="BLACKLIST_CHECK">
    <party type="PHYSICAL" hid="12">
      <field name="blacklistCheckStatus">FOUND</field>
      <source hid="10" sourceSystem="AL" rawId="11"/>
      <source hid="11" sourceSystem="BT" rawId="12"/>
      <relation type="40" hashCode="12345">
        <second type="BLACKLIST" hid="100" sourceSystem="TERRORIST_CB" rawId="1"
/>
      </relation>
      <relation type="40" hashCode="54321">
        <second type="BLACKLIST" hid="101" sourceSystem="BL_FL" rawId="2"/>
      </relation>
    </party>
    <relatedEssence type="BLACKLIST" hid="100" sourceSystem="TERRORIST_CB" rawId="
1">
      <field name="blacklistId">1</field>
      <field name="blacklistName">TERRORIST_CB</field>
      <field name="matchScope">98</field>
      <field name="rule">1</field>
      <field name="blacklistIdAuthor">TERRORIST_CB:1</field>
      <field name="surname">ТАБОЛАЕВ</field>
      <field name="patronymic">АБДУРАХМАНОВИЧ</field>
      <field name="name">ТОМАС</field>
      <field name="nameRawSource">ТАБОЛАЕВ ТОМАС АБДУРАХМАНОВИЧ</field>
      <field name="nameAuthor">TERRORIST_CB:1</field>
      <field name="birthdate">1975</field>
      <field name="birthdateAuthor">TERRORIST_CB:1</field>
      <field name="birthPlace">Г. УРУС-МАРТАН УРУС-МАРТАНОВСКОГО РАЙОНА
ЧИАССР</field>
      <field name="birthPlaceAuthor">TERRORIST_CB:1</field>
      <field name="inn">123123</field>
      <field name="innAuthor">TERRORIST_CB:1</field>
      <field name="documentNumber">44444444</field>
```

```

    <field name="documentNumberAuthor">TERRORIST_CB:1</field>
    <field name="actualityDate"/>
    <field name="actualityDateAuthor">TERRORIST_CB:1</field>
  </relatedEssence>
  <relatedEssence type="BLACKLIST" hid="101" sourceSystem="BL_FL" rawId="2">
    <field name="blacklistId">2</field>
    <field name="blacklistName">BL_FL</field>
    <field name="matchScope">98</field>
    <field name="rule">1</field>
    <field name="blacklistIdAuthor">BL_FL:2</field>
    <field name="surname">ТАБОЛАЕВ</field>
    <field name="patronymic">АБДУРАХМАНОВИЧ</field>
    <field name="name">ТОМАС</field>
    <field name="nameRawSource">ТАБОЛАЕВ ТОМАС АБДУРАХМАНОВИЧ</field>
    <field name="nameAuthor">BL_FL:2</field>
    <field name="birthdate">1975</field>
    <field name="birthdateAuthor">BL_FL:2</field>
    <field name="birthPlace">Г. УРУС-МАРТАН УРУС-МАРТАНОВСКОГО РАЙОНА
    ЧИАССР</field>
    <field name="birthPlaceAuthor">TERRORIST_CB:1</field>
    <field name="inn">123123</field>
    <field name="innAuthor">BL_FL:2</field>
    <field name="documentNumber">44444444</field>
    <field name="documentNumberAuthor">BL_FL:2</field>
    <field name="actualityDate"/>
    <field name="actualityDateAuthor">BL_FL:2</field>
  </relatedEssence>
</ns2:action>
</ns2:event>

```

#### JSON

```

{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "FOUND",
              "name": "blacklistCheckStatus"
            }
          ],
          "source": [
            {
              "field": [
                {
                  "value": "false",
                  "name": "archive"
                }
              ],
              "hid": 10,
              "sourceSystem": "AL",
              "rawId": "11"
            }
          ],
          {
            "field": [
              {
                "value": "false",
                "name": "archive"
              }
            ]
          }
        }
      ]
    }
  ]
}

```

```

    ],
    "hid": 11,
    "sourceSystem": "BT",
    "rawId": "12"
  }
],
"hid": 12,
"sourceSystem": null,
"rawId": null,
"attribute": null,
"type": "PHYSICAL"
}
],
"changes": null,
"relation": {
  "second": {
    "type": "BLACKLIST",
    "hid": 100,
    "rawId": "1",
    "sourceSystem": "TERRORIST_CB"
  },
  "type": 40,
  "deleted": null,
  "hashCode": "12345"
},
"relation": {
  "second": {
    "type": "BLACKLIST",
    "hid": 101,
    "rawId": "2",
    "sourceSystem": "BL_FL"
  },
  "type": 40,
  "deleted": null,
  "hashCode": "54321"
},
"relatedEssence": {
  "field": [
    {
      "value": "1",
      "name": "blacklistId"
    },
    {
      "value": "TERRORIST_CB",
      "name": "blacklistName"
    },
    ...
    {
      "value": "TERRORIST_CB:1",
      "name": "actualityDateAuthor"
    }
  ],
  "hid": 100,
  "sourceSystem": "TERRORIST_CB",
  "rawId": "1",
  "type": "BLACKLIST"
},
"relatedEssence": {
  "field": [
    {
      "value": "2",
      "name": "blacklistId"
    }
  ]
}

```

```

    },
    {
      "value": "BL_FL",
      "name": "blacklistName"
    },
    ...
    {
      "value": "BL_FL:2",
      "name": "actualityDateAuthor"
    }
  ],
  "hid": 101,
  "sourceSystem": "TERRORIST_CB",
  "rawId": "2",
  "type": "BLACKLIST"
},
"type": "BLACKLIST_CHECK"
}
],
"stamp": 1695282318938
}

```

Для объединенных клиентов идентификаторы исходных клиентов записываются в атрибут source.

#### XML

```

<ns2:action type="BLACKLIST_CHECK">
  <party type="PHYSICAL" hid="12">
    <field name="blacklistCheckStatus">FOUND</field>
    <source hid="10" sourceSystem="AL" rawId="11"/>
    <source hid="11" sourceSystem="BT" rawId="12"/>
  ...

```

#### JSON

```

...
"source": [
  {
    "hid": 10,
    "sourceSystem": "AL",
    "rawId": "11"
  },
  {
    "hid": 11,
    "sourceSystem": "BT",
    "rawId": "12"
  }
]
...

```

Для клиентов, которые ни с кем не объединялись — в party.

#### XML

```

<ns2:action type="BLACKLIST_CHECK">
  <party type="PHYSICAL" hid="12" sourceSystem="AL" rawId="11">
  ...

```

## Выход из черного списка

Исходное состояние: контрагент найден в черных списках.

Событие: контрагент больше не входит ни в один черный список.

### XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:event stamp="2015-04-14T17:42:36" xmlns:ns2="http://hflabs.ru/cdi/jms/2_27" xmlns="
http://hflabs.ru/cdi/soap/2_13">
  <ns2:action type="BLACKLIST_CHECK">
    <party type="PHYSICAL" hid="12">
      <field name="blacklistCheckStatus">NOT_FOUND</field>
      <source hid="10" sourceSystem="AL" rawId="11"/>
      <source hid="11" sourceSystem="BT" rawId="12"/>
    </party>
  </ns2:action>
</ns2:event>
```

### JSON

```
{
  "action": [
    {
      "attribute": null,
      "party": [
        {
          "field": [
            {
              "value": "NOT_FOUND",
              "name": "blacklistCheckStatus"
            }
          ],
          "source": [
            {
              "hid": 10,
              "sourceSystem": "AL",
              "rawId": "11"
            },
            {
              "hid": 11,
              "sourceSystem": "BT",
              "rawId": "12"
            }
          ],
          "hid": 12,
          "sourceSystem": null,
          "rawId": null,
          "attribute": null,
          "type": "PHYSICAL"
        }
      ],
      "changes": null,
      "type": "BLACKLIST_CHECK"
    }
  ],
  "stamp": 1695282318938
}
```



# JMS. НАСТРОЙКА ОТПРАВКИ ПРОШЛЫХ ЗНАЧЕНИЙ ФИО И ДУЛ

Единый клиент умеет отправлять в JMS сообщениях прошлые значения атрибутов ФИО и ДУЛ. В настройках системы необходимо для параметра `jms.events.attributes.pastValues` установить значение `true`. Рестарт CDI не требуется.

Прошлые значения атрибутов ФИО и ДУЛ добавляются в JMS сообщения при формировании событий `merge`, `unmerge`, `update` и `close`. Информация о прошлых значениях возвращается в сущности `pastAttribute`.

Для события `merge` прошлые атрибуты выгружаются в действии `create` или в действии `update`.

Если действие входит в перечень действий, для которых необходимо компактное представление, то для него прошлые атрибуты не выгружаются.

Если количество прошлых атрибутов более десяти, выгружаются десять последних по времени актуальности атрибутов суммарно для ФИО и ДУЛ.

## *Пример. Отправка прошлых значений атрибутов ФИО и ДУЛ для события update*

```
<ns2:event stamp="2023-10-02T14:06:26" xmlns="http://hflabs.ru/cdi/soap/2_13" xmlns:ns2="
http://hflabs.ru/cdi/jms/2_27">
  <ns2:action type="UPDATE">
    <party type="PHYSICAL" hid="1024" sourceSystem="QLF" rawId="1">
      <field name="surname">Виноградов</field>
      <field name="name">Вадим</field>
      <field name="patronymic">Петрович</field>
      <field name="gender">MALE</field>
      <field name="fullNameRawSource">Виноградов Егор Петрович</field>
      <field name="birthdate">04.04.1977</field>
      <field name="actualityDateAuthor">admin_performer</field>
      <field name="actualityDate">02.10.2023 14:06:26</field>
      <field name="lastChangeTimestamp">1696244786487</field>
      <attribute type="DOCUMENT_PASSPORT" hid="1024" rawId="DOCUMENT_PASSPORT.
1024">
        <field name="qualifierType">PASSPORT_RU</field>
        <field name="issueDate">01.04.1994</field>
        <field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
        <field name="departmentCode">111-437</field>
        <field name="expiryDate">01.04.2024</field>
        <field name="invalidStatus">ACTUAL</field>
        <field name="popularity">NONE</field>
        <field name="qualityCode">CONFIRMED_MANUALLY</field>
        <field name="type">PASSPORT</field>
        <field name="documentSeries">72 06</field>
        <field name="documentNumber">821258</field>
        <field name="rawSource">72 06•821255•01.04.1994•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-433•01.04.2024</field>
        <field name="primary">true</field>
        <field name="additionalInfo">нацпорт</field>
        <field name="actualityDate">02.10.2023 14:02:35</field>
        <field name="author">admin_performer</field>
      </attribute>
```

```

<pastAttribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1024">
  <field name="qualifierType">PASSPORT_RU</field>
  <field name="issueDate">01.04.1994</field>
  <field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
  <field name="departmentCode">111-437</field>
  <field name="expiryDate">01.04.2024</field>
  <field name="invalidStatus">ACTUAL</field>
  <field name="popularity">NONE</field>
  <field name="qualityCode">CONFIRMED_MANUALLY</field>
  <field name="type">PASSPORT</field>
  <field name="documentSeries">72 06</field>
  <field name="documentNumber">821255</field>
  <field name="rawSource">72 06•821255•01.04.1994•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-433•01.04.2024</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"></field>
  <field name="actualityDate">02.10.2023 14:02:35</field>
  <field name="author">admin_performer</field>
</pastAttribute>
<pastAttribute type="FIO">
  <field name="qualityCode">EDITED</field>
  <field name="surname">Виноградов</field>
  <field name="name">Егор</field>
  <field name="patronymic">Петрович</field>
  <field name="type">DEFAULT</field>
  <field name="rawSource">Виноградов Егор Петрович</field>
  <field name="primary">>false</field>
  <field name="additionalInfo"></field>
  <field name="actualityDate">02.10.2023 14:02:16</field>
  <field name="author">QLF:1</field>
</pastAttribute>
<source hid="1024" sourceSystem="QLF" rawId="1"/>
</party>
</ns2:action>
</ns2:event>

```

*Пример. Отправка прошлых значений атрибутов ФИО и ДУЛ для события merge (в действии create)*

```

<ns2:event stamp="2023-10-02T14:07:47" xmlns="http://hflabs.ru/cdi/soap/2_13" xmlns:ns2="
http://hflabs.ru/cdi/jms/2_27">
  <ns2:action type="MERGE">
    <party type="PHYSICAL" hid="1024" sourceSystem="QLF" rawId="1"/>
  </ns2:action>
  <ns2:action type="MERGE">
    <party type="PHYSICAL" hid="1027" sourceSystem="QLF" rawId="2"/>
  </ns2:action>
  <ns2:action type="CREATE">
    <party type="PHYSICAL" hid="1028">
      <field name="surname">Виноградов</field>
      <field name="name">Вадим</field>
      <field name="patronymic">Петрович</field>
      <field name="gender">MALE</field>
      <field name="fullNameRawSource">Виноградов Егор Петрович</field>
      <field name="actualityDateAuthor">admin_performer</field>
      <field name="actualityDate">02.10.2023 14:06:26</field>
      <field name="lastChangeTimestamp">1696244867969</field>
      <attribute type="DOCUMENT_PASSPORT" hid="1028" rawId="DOCUMENT_PASSPORT.
1024">
        <field name="qualifierType">PASSPORT_RU</field>

```



```

<field name="issueDate">01.04.1994</field>
<field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
<field name="departmentCode">111-437</field>
<field name="expiryDate">01.04.2024</field>
<field name="invalidStatus">ACTUAL</field>
<field name="popularity">NONE</field>
<field name="qualityCode">CONFIRMED_MANUALLY</field>
<field name="type">PASSPORT</field>
<field name="documentSeries">72 06</field>
<field name="documentNumber">821258</field>
<field name="rawSource">72 06•821255•01.04.1994•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-433•01.04.2024</field>
<field name="primary">true</field>
<field name="additionalInfo">нацпопт</field>
<field name="actualityDate">02.10.2023 14:02:35</field>
<field name="author">admin_performer</field>
</attribute>
<pastAttribute type="DOCUMENT_PASSPORT" rawId="DOCUMENT_PASSPORT.1024">
<field name="qualifierType">PASSPORT_RU</field>
<field name="issueDate">01.04.1994</field>
<field name="issueAuthority">ПВО ОВД СЕВЕРНОЕ БУТОВО</field>
<field name="departmentCode">111-437</field>
<field name="expiryDate">01.04.2024</field>
<field name="invalidStatus">ACTUAL</field>
<field name="popularity">NONE</field>
<field name="qualityCode">CONFIRMED_MANUALLY</field>
<field name="type">PASSPORT</field>
<field name="documentSeries">72 06</field>
<field name="documentNumber">821255</field>
<field name="rawSource">72 06•821255•01.04.1994•ПВО ОВД СЕВЕРНОЕ
БУТОВО•111-433•01.04.2024</field>
<field name="primary">false</field>
<field name="additionalInfo"></field>
<field name="actualityDate">02.10.2023 14:02:35</field>
<field name="author">admin_performer</field>
</pastAttribute>
<pastAttribute type="FIO">
<field name="qualityCode">EDITED</field>
<field name="surname">Виноградов</field>
<field name="name">Егор</field>
<field name="patronymic">Петрович</field>
<field name="type">DEFAULT</field>
<field name="rawSource">Виноградов Егор Петрович</field>
<field name="primary">false</field>
<field name="additionalInfo"></field>
<field name="actualityDate">02.10.2023 14:02:16</field>
<field name="author">QLF:1</field>
</pastAttribute>
<source hid="1024" sourceSystem="QLF" rawId="1"/>
<source hid="1027" sourceSystem="QLF" rawId="2"/>
</party>
</ns2:action>
</ns2:event>

```

# СТРУКТУРА JMS СООБЩЕНИЯ

- Заголовки выходных JMS-сообщений
- Тело выходных JMS-сообщений

# ЗАГОЛОВКИ ВЫХОДНЫХ JMS-СООБЩЕНИЙ

Единый клиент пишет в заголовок любого выходного JMS-сообщения два параметра:

- *cdiEventType*, может принимать значения:
  - UPDATE — обновление контрагента;
  - CREATE — создание контрагента;
  - CLOSE — закрытие контрагента;
  - DELETE — необратимое удаление контрагента;
  - MERGE — слияние контрагентов;
  - UNMERGE — разлияние контрагентов;
  - RELATION\_CHANGE — создание или закрытие связей между сущностями;
  - BLACKLIST\_CHECK — результат проверки по чёрным спискам;
  - DEPERSONALIZE — деперсонализация контрагента;
- *cdiPartyType* содержит тип контрагента или "RELATION" в случае создания/закрытия связей.

## Примеры

1. Создан новый атрибут у ФЛ

```
cdiEventType=UPDATE  
cdiPartyType=PHYSICAL
```

2. Изменено ФИО в договоре

```
cdiEventType=UPDATE  
cdiPartyType=AGREEMENT
```

3. Юридическое лицо закрыто

```
cdiEventType=CLOSE  
cdiPartyType=LEGAL
```

4. Создана новая связь

```
cdiEventType=RELATION_CHANGE  
cdiPartyType=RELATION
```

5. После проверки по черным спискам ФЛ меняет статус

```
cdiEventType=BLACKLIST_CHECK  
cdiPartyType=PHYSICAL
```

# ТЕЛО ВЫХОДНЫХ JMS-СООБЩЕНИЙ

- JMS.XSD-схема
- JMS.Объектная модель интерфейса
- JMS.Типы событий

# JMS.XSD-СХЕМА

## *Модель событий*

Файл	Описание
<a href="#">party-jms-event.xsd</a>	Модель JMS-событий

## *Модель вложенных объектов*

Файл	Описание
<a href="#">partyra.wadl</a>	WADL сервиса PartyRA
<a href="#">partyws.wsdl</a>	WSDL сервиса PartyWS
<a href="#">partyws-model.xsd</a>	Модель данных контрагентов в системе
<a href="#">partyws-msg.xsd</a>	Модель данных сервисов PartyWS и PartyRA
<a href="#">common-msg.xsd</a>	Модель данных базовых сообщений

# JMS. ОБЪЕКТНАЯ МОДЕЛЬ ИНТЕРФЕЙСА

Объектная модель JMS интерфейса состоит из двух элементов:

- **action** - Список произошедших событий.
- **stamp** - Дата изменения БД, вызванное событием

## Описание элемента Action

В данном списке указывается только одно событие и события, являющиеся дочерними к нему. Несколько примеров:

- Создание нескольких контрагентов - на каждого созданного контрагента будет отправлено отдельное сообщение. Пример сообщения: [JMS. Создание контрагента](#)
- Изменение нескольких контрагентов - на каждого измененного контрагента будет отправлено отдельное сообщение. Пример сообщения: [JMS. Изменение контрагента](#)
- Слияние контрагентов и создание новой карточки - будет отправлено одно сообщение с двумя событиями: [MERGE](#) (родительское) и [CREATE](#) (дочернее). Пример сообщения: [JMS. Слияние контрагентов](#)
- Создание нескольких связей - на каждую созданную связь отправляется отдельное сообщение. Пример сообщения: [JMS. Создание связей](#)

Элемент **action** включает в себя следующие элементы:

Имя элемента	Тип	Обязательный	Описание
attribute	WAttribute	Нет	Изменение псевдо-атрибута
party	WParty	Нет	Изменяемый контрагент. Содержит в себе те же сведения, что и ответы SOAP запросов. Состав полей для каждого заказчика свой, можно посмотреть в разделе "Автоматическая модель данных"
changes	Change (см. ниже)	Нет	Изменения, произошедшие с контрагентом
relation	WRelation	Нет	Связи с другими объектами
relatedEssence	WParty	Нет	Связанные сущности с объектом
type	Action Type	Да	Тип события

## Описание элемента Changes

В данном элементе указываются изменения, произошедшие с контрагентом. Данный элемент включается в сообщение, в случае, если подключен [расширенный](#) формат сообщения. Несколько примеров:

- У контрагента изменилось имя - Changes будет заполнен в расширенном формате сообщения с type = UPDATE. Пример сообщения: [Обновление из исходной системы](#)
- Контрагенту добавили дублирующий телефон - Changes будет заполнен в расширенном формате сообщения с type = CLOSE. Пример сообщения: [Схлопывание атрибута](#)
- Контрагенту добавили телефон - Changes будет заполнен в расширенном формате сообщения с type = CREATE. Пример сообщения: [Новый атрибут](#)

Имя элемента	Тип	Обязательный	Описание
detail	Detail (см. ниже)	Нет	Описание изменяемых полей
type	EssenceChangeType	Да	Тип изменения. Включает в себя следующие значения: CREATE, UPDATE, CLOSE
essense	string	Да	Тип изменяемого атрибута из объектной модели - <a href="#">Элементы — объектная модель (онлайн)</a>
hid	long	Да	Уникальный идентификатор изменяемого атрибута в CDI
rawId	string	Да (может быть null)	Уникальный идентификатор изменяемого атрибута в системе источнике
sourceSystem	string	Да (может быть null)	Код системы источника

### Описание элемента *Detail*

В данном элементе указываются поля, которые изменились в следствии исполнения события

Имя элемента	Тип	Обязательный	Описание
oldValue	string	Да	Старое значение поля
newValue	string	Да	Новое значение поля
fieldName	string	Да	Название изменяемого поля из объектной модели - <a href="#">Объектная модель онлайн-интерфейсов</a>

# JMS.ТИПЫ СОБЫТИЙ

Типы возможных событий (ActionType), генерируемых в JMS

- UPDATE — обновление контрагента;
- CREATE — создание контрагента;
- CLOSE — закрытие контрагента;
- DELETE — **необратимое удаление** контрагента;
- MERGE — слияние контрагентов;
- UNMERGE — разлияние контрагентов;
- RELATION\_CHANGE — создание или закрытие связей между сущностями;
- BLACKLIST\_CHECK — результат проверки по чёрным спискам;
- DEPERSONALIZE — **деперсонализация** контрагента.



# JMS.FAQ

- Где можно найти полный перечень полей, который передается?

**Релевантный перечень полей, которые могут вернуться в ответе, можно посмотреть в модели данных: сущности, атрибуты.**

- В каких случаях информация о событии объединяется в одно сообщения? Не могли бы раскрыть это на примерах и как будут отличаться сообщения?

**Примеры таких сообщений можно посмотреть, например, на страницах ‘Закрытие контрагента’, ‘Слияние контрагента’, ‘Разъединение контрагента’ в документации по JMS. Интересующие вас примеры содержат несколько блоков <action> в ответе. Опишу один из вариантов — закрытие:**

**а. Есть гипотетическая объединенная карточка клиента, которая состоит из трех исходных карточек.**

**б. Одну из исходных карточек закрыли.**

**с. В JMS-сообщение попадет два события:**

**Закрытие одной из исходных карточек (CLOSE).**

**Изменение состояния объединенной карточки (UPDATE).**

**Т.е. иногда одно событие (в данном примере CLOSE) приводит к дополнительным (в данном примере UPDATE), и в JMS-сообщении в таком случае вернутся несколько связанных событий (CLOSE + UPDATE).**

- Может ли информация о двух слияниях прийти в одном JMS-сообщении?

1024 и 1025 объединились в 1026

1027 и 1028 объединились в 1029

Это будет 2 сообщения или одно?

**Это будет несколько сообщений:**

**а. Пример 1**

**а) Через метод save создали 4 карточки (1024, 1025, 1027, 1028).**

**б) В JMS отправилось 4 отдельных события о создании карточек.**

**в) Далее в ЕК отработал штатный процесс поиска и слияния дублей.**

**г) Карточки слились (1024+1025?1026, 1027+1028?1029).**

**г) В JMS уйдет два сообщения о слиянии.**

**б. Пример 2**

**а) Через метод saveAndMerge создали 4 карточки (1024, 1025, 1027, 1028).**

**б) Поиск дублей и слияние выполнены в онлайн во время обработки запроса.**

**в) В JMS отправилось 6 отдельных сообщений (4 о создании и 2 о слиянии).**

- Как на стороне ЕК обрабатывается следующая ситуация:

Было 2 карточки, которые объединились в одну

Потом произошла смена фамилии

Новый hid\_original? Или обновление одной из карточек и как следствие разъединение ранее слитых?

**Здесь может быть несколько вариантов, чтобы лучше их понять рекомендуется ознакомиться с правилами обнаружения конфликтов.**

**а. При обновлении ФИО на одной из исходных может произойти конфликт обновления, тогда эта исходная автоматически выльется из объединенной, и после этого на ней изменится ФИО. Примеры возможных причин конфликтов при обновлении ФИО с описанием есть по ссылке выше.**

**НID (идентификатор в ЕК) у исходной в таком случае не меняется, у объединенной карточки тоже. При этом если объединенная карточка состояла только из двух исходных, и одна из них вылилась, то из объединенной выливается и вторая карточка, потому что объединять больше нечего. НID у обоих исходных в таком случае все ещё не меняется.**

**б. При обновлении ФИО на одной из исходных карточек оно может измениться на объединенной, если выигрывает у ФИО другой/других исходных по правилам слияния.**

- Можно ли выяснить на стороне CDI дошло ли сообщения из ОП до исходной системы?

**Нет, на стороне CDI нельзя посмотреть, дошло ли сообщение до исходной системы**