

# **АРМ «Орион Икс»**

Р.АЦДР.00086

версия 1.2.0

## **Модуль «Сервис интеграции Орион Икс»**

---

### Инструкция



## Содержание

Общие сведения .....	8
Глава 1. Подключение модуля в АРМ «Орион Икс».....	10
Глава 2. Транспортный уровень модуля .....	17
2.1 Протокол HTTP/1.1.....	17
2.1.1 Содержание заголовков протокола HTTP/1.1 .....	18
2.1.2 Содержание тела протокола HTTP/1.1 .....	18
2.2 WebSocket-подключение и постоянные соединения.....	19
Глава 3. Взаимодействие с API-приложением.....	20
3.1 Процесс авторизации.....	20
3.2 Взаимодействие с WebSocket- подключением .....	21
3.3 Взаимодействие с запросами.....	21
3.3.1 Структура запроса JSON-RPC .....	21
3.3.1.1 Запросы Batch Requests.....	23
3.3.2 Структура запросов с примерами .....	24
start (начало сессии).....	24
login (авторизация).....	25
renew_connection (поддержание сессии) .....	26
ЗАПРОСЫ ДАННЫХ.....	27
get_computers (получение списка идентификаторов компьютеров).....	27
get_elements (получение информации об элементах указанного типа) .....	28
get_computer_elements (получение информации об элементах указанного типа для идентификатора компьютера) .....	29
get_element (получение информации об элементе) .....	30
get_elements_by_link (получение информации об элементах, связанных с передаваемым).....	32
get_incident_types (получение списка типов инцидентов) .....	33
get_message_groups (получение списка групп сообщений).....	34
get_messages_desc (получение описания сообщений) .....	35
find_elements_by_param_i (поиск элементов по целочисленному значению) .....	36
find_elements_by_param_s (поиск элементов по строковому значению).....	37
find_elements_by_any_param_s (поиск элементов по строковому значению любого параметра).....	39
find_elements_by_name (поиск элементов по имени) .....	41
find_elements_by_additional_param_s (поиск пользователя по ключу) .....	42
types_by_tag (получение списка типов для прототипа).....	44
get_events (получение истории событий).....	45
СОЗДАНИЕ, УДАЛЕНИЕ, РЕДАКТИРОВАНИЕ .....	46
create_element (создание элемента) .....	46
create_element_by_link (создание дочернего элемента).....	47
delete_element (удаление элемента).....	48
update_element (изменение элемента) .....	49
create_link (создание связи между элементами) .....	54

delete_link (удаление связи между элементами).....	55
УПРАВЛЕНИЕ .....	56
get_possible_commands_by_element (получение списка возможных команд) .....	56
send_command_query_for_element (отправка запроса на выполнение команды) .....	58
Приложение А. Описание полей типов.....	61
cars.....	61
car .....	61
company .....	61
department.....	61
user .....	61
Дополнительные таблицы (относятся только к user) .....	62
departments .....	64
logic_group_zones.....	64
logic_zone .....	64
s2000m.....	64
signal_20 .....	64
signal_20p .....	65
s2000_sp1 .....	65
s2000_4.....	65
s2000_4_v300 .....	66
s2000_k.....	66
s2000_it.....	66
s2000_kdl.....	67
s2000_kdl_v300 .....	67
s2000_bi.....	67
s2000_bki_v220 .....	67
s2000_bi_v223 .....	68
signal_20_02 .....	68
s2000_ks .....	68
s2000_aspt.....	69
s2000_aspt_v200 .....	69
s2000_aspt_v300 .....	69
s2000_kpb.....	70
s2000_2.....	70
s2000_2_v220 .....	70
s2000_2_v275 .....	70
rupor.....	71
rupor_v200.....	71
rupor_disp_01 .....	71
s2000_pt.....	72
uo_4s.....	72
potok_3n .....	72
potok_3n_v103.....	73
signal_20m.....	73

signal_20m_v200 .....	73
s2000_bi_01 .....	73
rupor_01 .....	74
rupor_01_v103 .....	74
s2000_adem .....	74
rip_12_50 .....	75
signal_10 .....	75
s2000_pp .....	75
rip_12_54 .....	76
rip_24_50 .....	76
s2000_kdl_s .....	76
s2000_kdl_2i .....	76
s2000_pge .....	77
s2000_bki_v223 .....	77
potok_bki .....	77
rupor_200 .....	78
s2000_perimeter .....	78
mip_12 .....	78
mip_24 .....	79
mip_24_s .....	79
rip_48_01 .....	79
rip_12_56 .....	79
rip_24_56 .....	80
sirius .....	80
sirius_v200 .....	80
rupor_02 .....	81
s2000_kdl_modbus .....	81
rupor_03 .....	82
rupor_300 .....	82
rupor_300_v103 .....	82
rupor_300_v105 .....	83
rupor_300_v200 .....	83
rupor_300_v205 .....	83
s2000_kpb_s .....	84
s2000_kpb_s_sirius_v200 .....	84
s2000_pge_01 .....	84
rip_24_57 .....	85
pkv_rip_12_56 .....	85
pkv_rip_24_56 .....	85
s2000_kdl_2i_01 .....	85
s2000_kdl_2i_01_v300 .....	86
shkp_rs .....	86
microphone_console_20 .....	86
rupor_disp_02 .....	87

mip_12_11 .....	87
mip_24_11 .....	87
s2000_rpi_02.....	87
rupor_75.....	88
unknown_device.....	88
user_device .....	88
orion_input.....	89
orion_reader .....	89
orion_output.....	89
orion_destination.....	90
orion_zone.....	90
orion_group_zone.....	90
position .....	91
positions .....	91
Приложение Б. Связи между типами .....	92
Car .....	92
Cars.....	92
Company.....	92
Department.....	92
Departments .....	92
Logic_group_zones .....	92
Logic_zone.....	93
Microphone_console_20.....	94
Mip_12.....	94
Mip_12_11 .....	94
Mip_24.....	95
Mip_24_11 .....	95
Mip_24_s .....	95
Orion_destination.....	95
Orion_group_zone.....	96
Orion_input.....	96
Orion_output .....	97
Orion_reader .....	98
Orion_zone.....	98
Pkv_rip_12_56.....	100
Pkv_rip_24_56.....	100
Position .....	100
Positions.....	101
Potok_3n.....	101
Potok_3n_v103 .....	101
Potok_bki.....	101
Rip_12_50.....	101
Rip_12_54.....	102
Rip_12_56.....	102

Rip_24_50.....	102
Rip_24_56.....	102
Rip_24_57.....	103
Rip_48_01.....	103
Rupor.....	103
Rupor_01.....	103
Rupor_01_v103.....	104
Rupor_02.....	104
Rupor_03.....	104
Rupor_200.....	104
Rupor_300.....	105
Rupor_300_v103.....	105
Rupor_300_v105.....	105
Rupor_300_v200.....	105
Rupor_300_v205.....	106
Rupor_75.....	106
Rupor_disp_01.....	106
Rupor_disp_02.....	106
Rupor_v200.....	106
S2000_2.....	107
S2000_2_v220.....	107
S2000_2_v275.....	107
S2000_4.....	107
S2000_4_v300.....	108
S2000_adem.....	108
S2000_aspt.....	108
S2000_aspt_v200.....	108
S2000_aspt_v300.....	109
S2000_bi.....	109
S2000_bi_01.....	109
S2000_bi_v223.....	109
S2000_bki_v220.....	110
S2000_bki_v223.....	110
S2000_it.....	110
S2000_k.....	110
S2000_kdl.....	110
S2000_kdl_2i.....	111
S2000_kdl_2i_01.....	111
S2000_kdl_2i_01_v300.....	111
S2000_kdl_300.....	111
S2000_kdl_modbus.....	112
S2000_kdl_s.....	112
S2000_kpb.....	112
S2000_kpb_s.....	112

S2000_kpb_s_sirius_v200 .....	113
S2000_ks.....	113
S2000_perimeter .....	113
S2000_pge.....	113
S2000_pge_01 .....	114
S2000_pp .....	114
S2000_pt.....	114
S2000_rpi_02 .....	114
S2000_sp1 .....	114
S2000m.....	115
Shkp_rs .....	116
Signal_10 .....	116
Signal_20 .....	116
Signal_20_02 .....	117
Signal_20m .....	117
Signal_20m_v200.....	117
Signal_20p .....	117
Sirius .....	118
Sirius_v200 .....	119
Unknown_device.....	120
Uo_4s.....	120
User.....	120
User_device.....	121

## Общие сведения

Модуль «Сервис интеграции Орион Икс» (далее – модуль) предназначен для работы с сервером через API-клиент и WebSocket-протокол. Для подключения модуля необходим установленный и настроенный АРМ «Орион Икс».

Задачи модуля:

- обрабатывать JSON-RPC-запросы;
- получать и управлять идентификаторами для аутентификации;
- инициализация сессии с сервером;
- обеспечивать непрерывности сессии через heart-beat;
- предоставлять доступ к данным определённых типов и их связям;
- отправлять через WebSocket-сервер уведомления о произошедших изменениях и событиях;
- шифрование SSL-сертификатами и методом DH;
- защита данных от MITM-атак механизмом handshake.

Данные, которые можно получить из JSON-RPC-протокола:

- Сотрудники:
  - Список сотрудников (все данные и ключи);
  - Данные и ключи сотрудника по id;
  - Добавление сотрудника (и ключи);
  - Обновление данных сотрудника (и ключи);
  - Удаление сотрудника (и ключи);
  - Список связанных с сотрудником автомобилей, организаций, отделов;
- Автомобили:
  - Список автомобилей (все данные);
  - Данные автомобиля по id;
  - Добавление автомобиля;
  - Редактирование автомобиля;
  - Удаление автомобиля;
  - Добавление связи сотрудника с автомобилем;
  - Удаление связи сотрудника с автомобилем;
  - Список автомобилей сотрудника;
  - Список сотрудников, связанных с автомобилем;
- Организации:
  - Список организаций (все данные);
  - Данные организации по id;
  - Добавление организации;
  - Редактирование организации;
  - Удаление организации;
  - Добавление связи сотрудника с организацией;
  - Удаление связи сотрудника с организацией;
  - Список организаций сотрудника;

- Список сотрудников, связанных с организацией;
- Должности:
  - Список должностей;
  - Данные должности по id;
  - Добавление должности;
  - Редактирование должности;
  - Удаление должности;
  - Добавление связи сотрудника с должностью;
  - Удаление связи сотрудника с должностью;
  - Должность сотрудника;
  - Список сотрудников, связанных с должностью;
- Устройства:
  - Список компьютеров;
  - Список всех приборов (все данные);
  - Список элементов определённого типа (тип прибора, вход, выход, считыватель и др.);
  - Список элементов определённого типа для определённого компьютера;
  - Список элементов, связанных с устройством (приборы, входы, выходы, зоны, группы зон и др.);
- Зоны:
  - Список зон;
  - Список групп зон;
  - Список элементов, входящих в состав указанной зоны;
  - Список зон, входящих в состав указанной группы зон;
- Управление:
  - Получение списка команд, доступных для элемента (прибор, вход, выход, зона);
  - Отправка запроса на выполнение команды;
- Поиск:
  - По строковому значению любого параметра для указанного типа;
  - По целочисленному параметру для указанного типа;
  - По строковому параметру для указанного типа;
  - Сотрудника по ключу;
  - Игнорирование регистра (да/нет);
  - Точное совпадение (да/нет);
  - Поиск элементов по имени;
- События:
  - Получение списка событий за указанную дату (указывается количество от 1 до 1 000, точка отсчёта и направление выборки);
  - Получение списка событий с описанием.

## Глава 1. Подключение модуля в АРМ «Орион Икс»



*Если Вы приобрели или обновили лицензию, сначала необходимо выгрузить систему, вставить ключ, а затем включить систему.*

**Шаг 1.** Создайте элемент «ТСР-подключение» в модуле «Менеджер конфигурации» → вкладка «Оборудование» → «Интерфейсы подключений». Здесь откройте окно «Создание элемента», вызвав его контекстное меню ПКМ «Добавить дочерний элемент». После нажатия кнопки «ОК» в дереве элементов появится «ТСР-подключение».

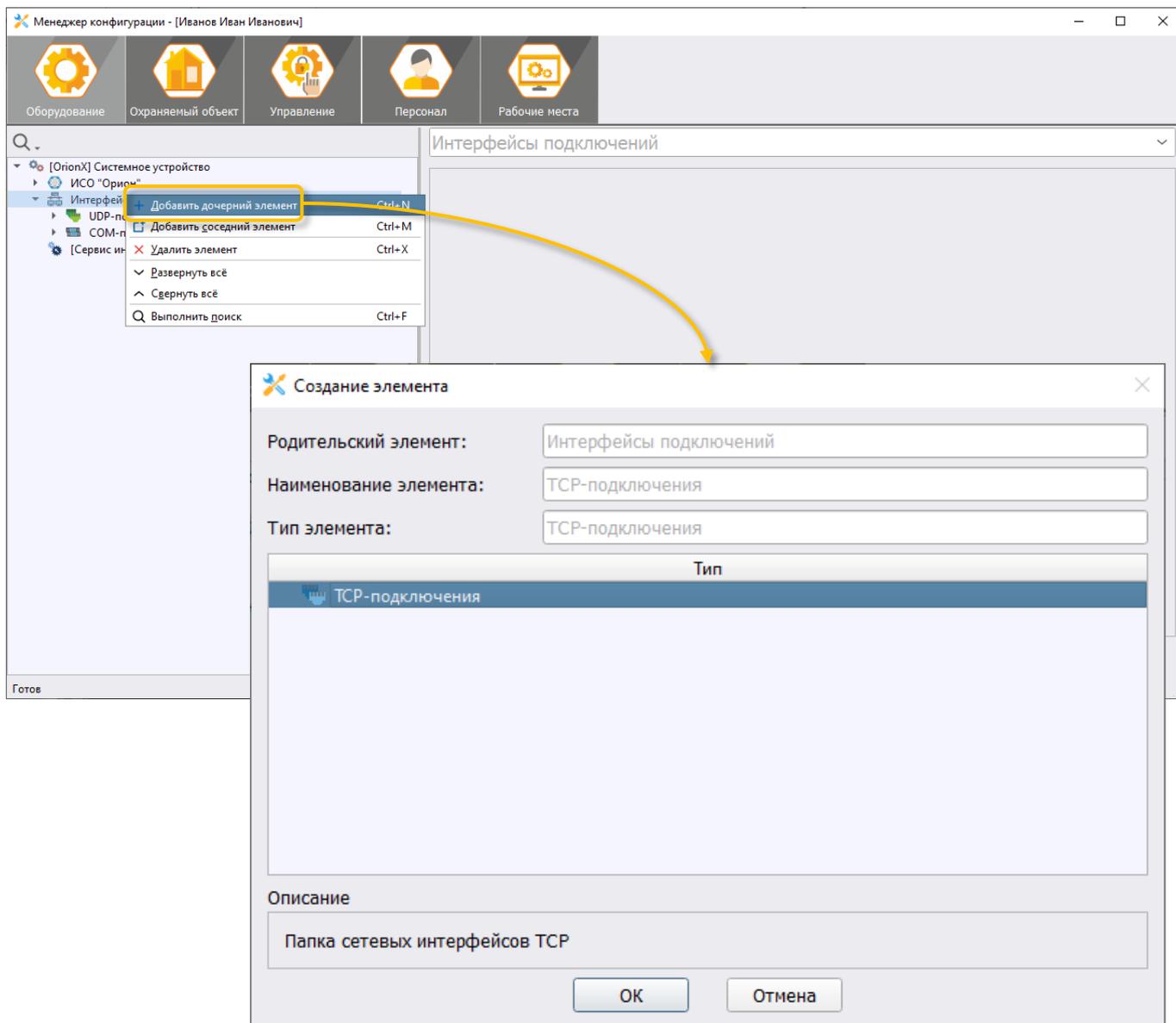


Рисунок 1 – Добавление ТСР-подключения

**Шаг 2.** Далее под TCP-подключением необходимо создать элемент «TCP-сервер». Для этого вызовите окно «Создание элемента» через контекстное меню «Добавить дочерний элемент», кликнув ПКМ по элементу «TCP-подключение». После нажатия кнопки «ОК» в дереве элементов появится «TCP-сервер» (при необходимости элемент можно переименовать). В настройках укажите для него свободный порт и сетевой интерфейс.

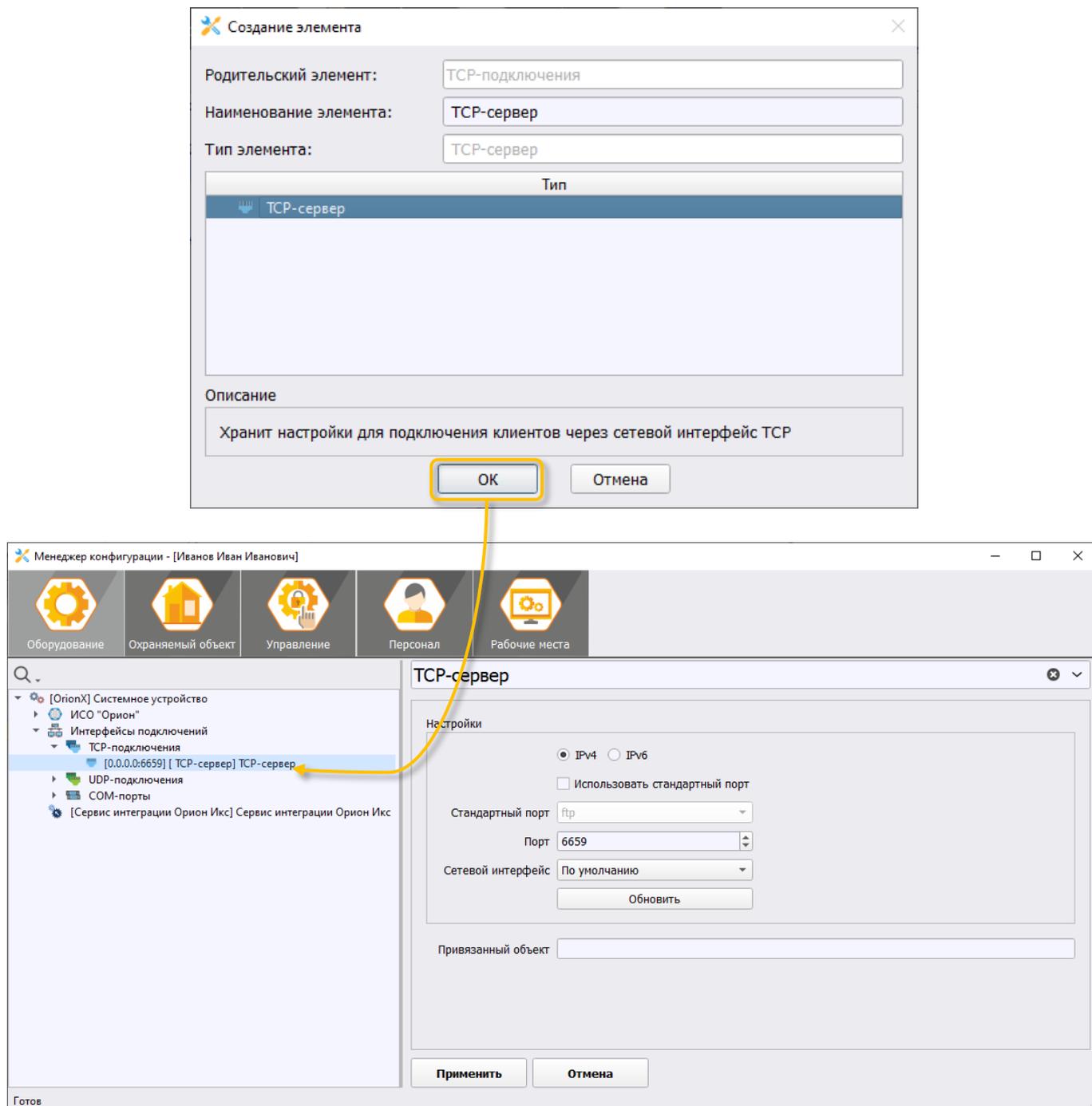


Рисунок 2 – Добавление TCP-сервера

**Шаг 3.** Добавить модуль «Сервис интеграции Орион Икс». Для этого вызовите окно «Создание элемента» ПКМ в дереве устройств по элементу «Системное устройство». Здесь выберите «Сервис интеграции Орион Икс». После нажатия кнопки «ОК» модуль будет добавлен в дерево устройств.

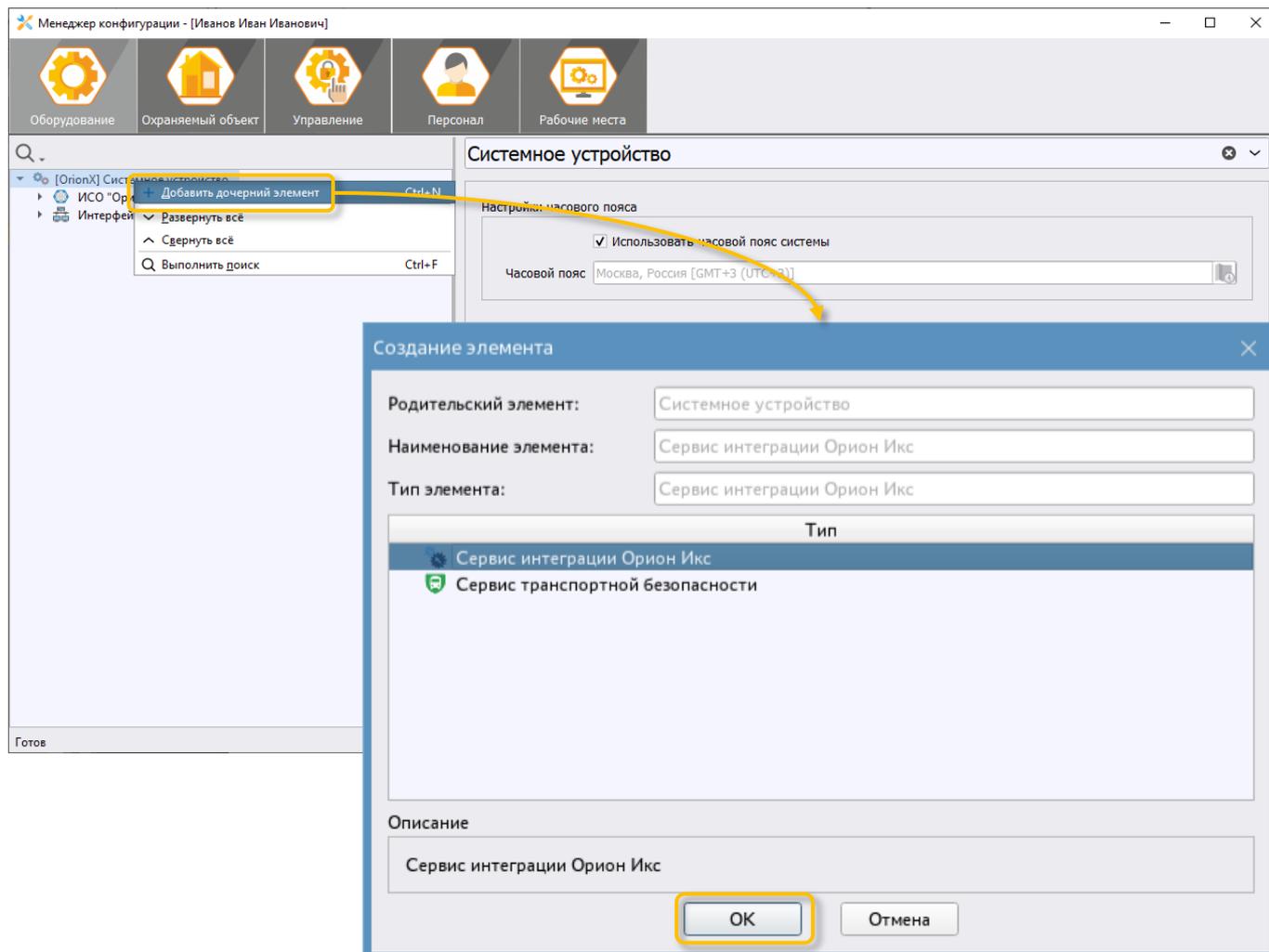


Рисунок 3 – Добавление модуля

**Шаг 4.** Далее необходимо добавить элемент «Канал связи JSON-RPC» через окно «Создание элемента» в элементе «Сервис интеграции Орион Икс».

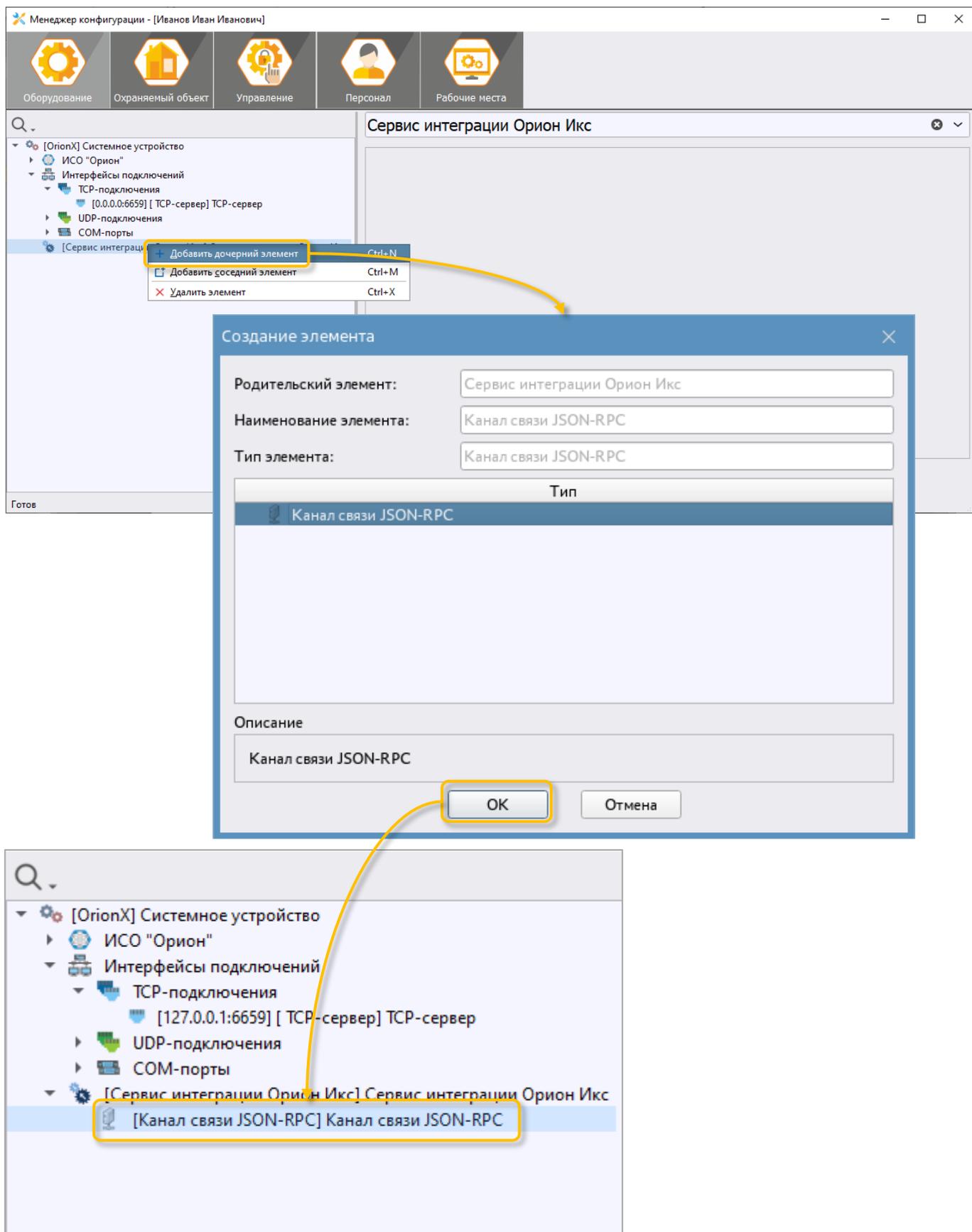


Рисунок 4 – Подключение канала связи JSON-RPC

## Шаг 5. Настроить элемент «Канал связи JSON-RPC».

Сервер поддерживает защищённое соединение HTTPS (HyperText Transfer Protocol Secure) и WSS (WebSocket Secure). Это гарантирует шифрование данных при передаче, защиту от перехвата и модификации сообщений.

На сервере и клиенте можно выбрать способ шифрования. Можно использовать метод SSL и при необходимости дополнительно можно использовать метод DH, алгоритм Диффи-Хеллмана. Метод DH используется только совместно с методом SSL.

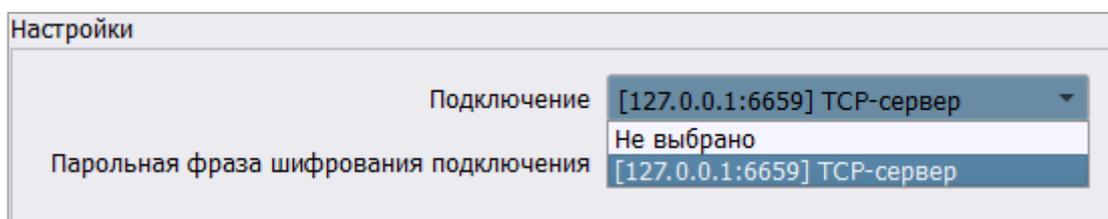
Помимо этого можно включить механизм CORS для контроля доступа к web-ресурсам с указанием конкретных доступных ресурсов, либо разрешить все источники.

По умолчанию всё шифрование отключено и контроль доступа к ресурсам также отключён.

В случае если обмен данными происходит внутри уже защищённой сети, то шифрование не обязательно, т.к. уже есть своя защита.

Для настройки безопасности передачи данных в модуле предусмотрены соответствующие настройки. Ниже приведено описание всех настроек модуля.

- **«Подключение»** – здесь выбирается созданный ранее TCP-сервер.



- **«Парольная фраза шифрования подключения»** – здесь можно задать пароль (ограничение длины в 255 символов)
- **«Использовать SSL»** – чекбокс включения/выключения настроек SSL-шифрования.
- **«Файл сертификата»** – здесь указывается месторасположение SSL-сертификата.
- **«Файл ключа»** – здесь указывается месторасположение SSL-ключа.
- **«Пароль ключа»** – здесь вводится пароль от SSL-ключа.
- **«DH-файл»** – здесь указывается месторасположение файла шифрования DH.
- **«Настройка совместного пользования веб-ресурсами (CORS)»** – включает/выключает контроль доступа к web-ресурсам.
- **«Разрешить все источники»** – отключение флажка позволяет включать только определённые источники. По кнопке **«Добавить источник»** в таблице **«Разрешенные источники»** добавляется строка, где необходимо указать разрешённый ресурс в формате <ip-адрес:порт>.

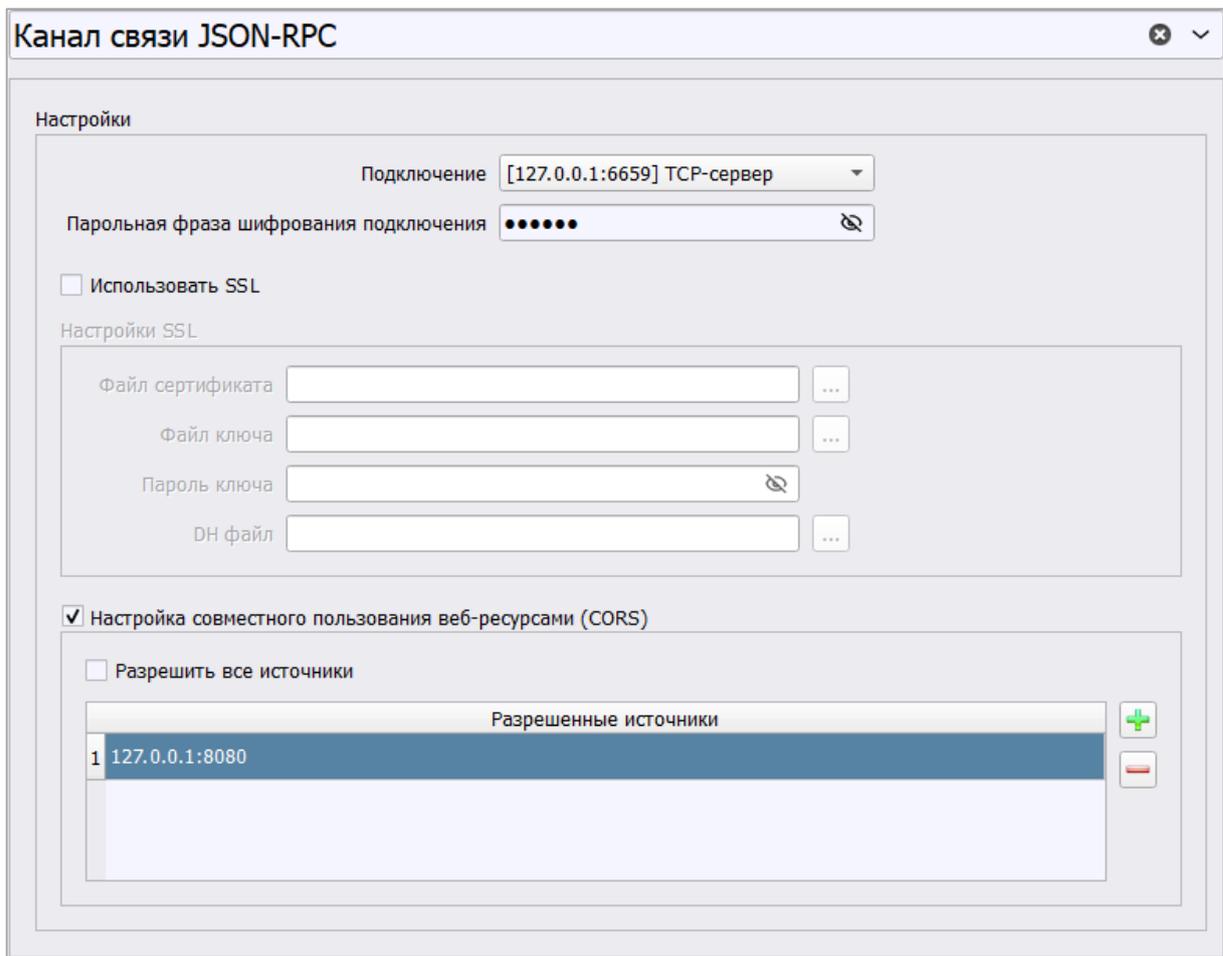


Рисунок 5 – Окно с настройками элемента «Канал связи JSON-RPC»

**Шаг 6.** Подключение пользователя к модулю. Для этого необходимо зайти на вкладку «Персонал» в Менеджере конфигурации и у выбранного пользователя перейти на вкладку «Полномочия». Здесь при нажатии на кнопку «Добавить»  в таблице появится строка с наименованием модуля. Задайте логин и пароль в соответствующих ячейках.

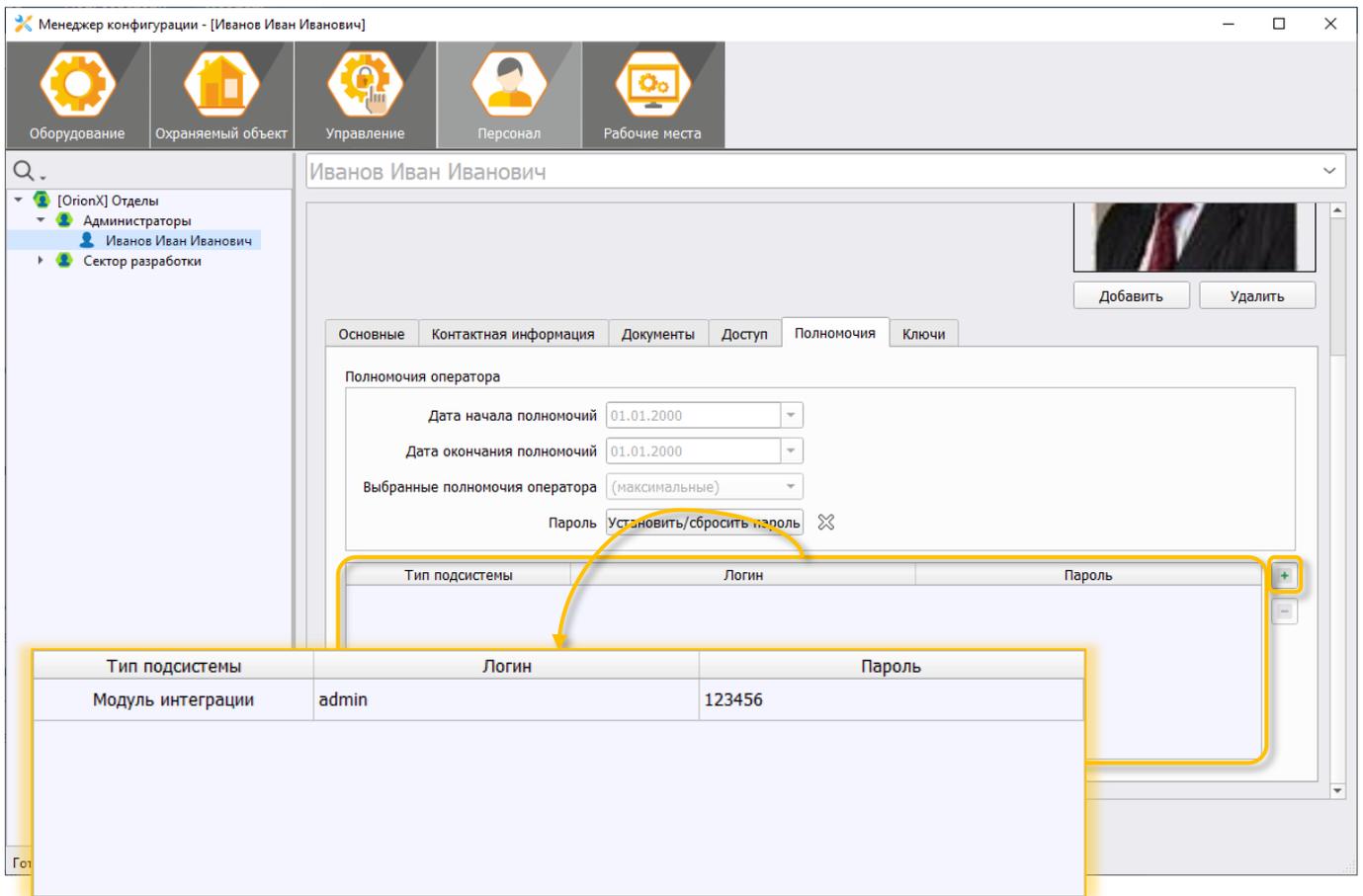


Рисунок 6 – Подключение пользователей к модулю

## Глава 2. Транспортный уровень модуля

Сервер интеграции использует JSON- RPC-протокол – лёгкий протокол удалённого вызова процедур RPC (Remote Procedure Call), передающий данные в формате JSON.

На низком уровне обмен сообщениями происходит по стандартным сетевым протоколам: HTTP/1.1 или WebSocket.

HTTP – это протокол прикладного уровня, который лежит в основе передачи данных во Всемирной паутине. Он работает по модели «клиент-сервер», где клиент (например, браузер или наше клиентское приложение) отправляет запрос (request) серверу, а сервер возвращает ответ (response). HTTPS – это расширение протокола HTTP, которое добавляет шифрование для защиты передаваемых данных. Он использует криптографические протоколы SSL (Secure Sockets Layer) или, что более современно и безопасно, TLS (Transport Layer Security).

Принципы шифрования HTTPS.

- **Конфиденциальность:** Шифрование гарантирует, что данные, передаваемые между клиентом и сервером, не могут быть прочитаны третьими лицами.
- **Целостность:** HTTPS обеспечивает проверку того, что данные не были изменены во время передачи.
- **Аутентификация:** Позволяет клиенту проверить подлинность сервера с помощью цифровых сертификатов, выданных доверенными центрами сертификации (CA). Это защищает от атак типа «человек посередине» (Man-in-the-Middle).

WebSocket – это сетевой протокол, который позволяет устанавливать постоянное, двунаправленное (full-duplex) соединение между клиентом и сервером поверх одного TCP-соединения. Он часто используется для приложений, требующих обмена данными в реальном времени.

В отличие от HTTP, где клиент инициирует каждый запрос, WebSocket позволяет серверу отправлять данные клиенту в любой момент после установления соединения, без необходимости явного запроса от клиента. Это значительно снижает задержки и накладные расходы по сравнению с традиционными техниками вроде HTTP-поллинга (регулярных запросов клиента к серверу).

Протокол WebSocket имеет две схемы URI, как и HTTP и HTTPS:

- `ws://`: Нешифрованное WebSocket соединение. Данные передаются в открытом виде.
- `wss://`: Шифрованное WebSocket соединение. Использует TLS (тот же механизм, что и HTTPS) для защиты передаваемых данных. Рукопожатие происходит через HTTPS, а последующий обмен данными шифруется.

Использование `wss://` и `https://` настоятельно рекомендуется, особенно при передаче чувствительной информации или при работе в недоверенных сетях, так как оно обеспечивает ту же конфиденциальность, целостность и аутентификацию сервера, что и HTTPS.

### 2.1 Протокол HTTP/1.1

На низком уровне запросы JSON-RPC основаны на протоколе HTTP 1.1. Это означает, что каждый запрос отправляется с использованием HTTP-метода **POST** на специальный адрес сервера, называемый точкой входа (endpoint). Для JSON-RPC всегда используется метод POST, т.к.

запросы содержат тело (в отличие от GET). Метод GET в JSON-RPC не используется, потому что параметры запроса не должны передаваться в URL.

Точка входа (endpoint) – это URL, на который клиент посылает запросы JSON-RPC.

Например: <http://127.0.0.1:6659> или, если используется SSL-шифрование: <https://127.0.0.1:6659>.

Протокол HTTP состоит из заголовков (контент) и тела (запрос или ответ).

### 2.1.1 Содержание заголовков протокола HTTP/1.1

Для успешной работы запроса клиент обязан передавать определённые HTTP-заголовки.

Обязательные заголовки	Пример	Описание
<метод> <путь до ресурса> <b>HTTP</b> / <b>&lt;версия&gt;</b>	POST /api/jsonrpc HTTP/1.1	строка запроса (метод и путь)
<b>Host</b>	Host: example.com	имя сервера
<b>Content-Type</b>	Content-Type: application/json	обязательно application/json, чтобы сервер понял, что тело запроса содержит JSON
<b>Content-Length</b>	Content-Length: <длина запроса в байтах>	количество байт в теле запроса
<b>Connection</b>	Connection: <keep-alive или close>	keep-alive: Соединение остаётся открытым для последующих запросов, пока не будет явно закрыто или не истечёт время ожидания close: закрывает соединение после завершения передачи данных

Примечание – После каждого заголовка идёт перевод строки (\n), а после последнего заголовка необходимо добавить две пустые строки (возврат каретки и перевод строки – \r\n), что отделит заголовки от тела.

### 2.1.2 Содержание тела протокола HTTP/1.1

Тело запроса должно включать корректный формат JSON. Необходимо строго соблюдать правила расстановки скобок, двоеточий и запятых. Ошибки в синтаксисе, такие как пропущенные запятые между элементами или двойные кавычки у ключей, приведут к некорректному формату.



**Формат не должен включать комментарии, лишние запятые и пробелов в ключевых местах.**

Пример корректного запроса (в сыром HTTP-виде):

Пример:

```
1: POST /api/jsonrpc HTTP/1.1
2: Host: example.com
3: Content-Type: application/json
4: Content-Length: 70
5:
6: {"jsonrpc": "2.0", "method": "getStatus", "params": {}, "id": 1}
```

Ответ так же передаётся в формате JSON и должен содержать одно из двух полей:

- **result** – если вызов прошёл успешно;  
или
- **error** – если произошла ошибка.

Пример успешного ответа:

```
1: {"jsonrpc": "2.0", "result": {"status": "ok"}, "id": 1}
```

Пример ошибки ответа:

```
1: {"jsonrpc": "2.0", "error": {"code": "-32601", "message": "Method not found"}, "id": 1}
```

## 2.2 WebSocket-подключение и постоянные соединения

Реализация сервера поддерживает WebSocket-подключение для отправки уведомлений клиенту без повторных HTTP-запросов.

В этом случае JSON-RPC-пакеты передаются в чистом виде (без HTTP-заголовков), но формат JSON остаётся тем же.

## Глава 3. Взаимодействие с API-приложением

### 3.1 Процесс авторизации

Процедура запуска и входа предназначена для организации защищённой сеансовой связи клиента с сервером. Здесь работает механизм `handshake` – процесс идентификации сервера и клиента при установлении безопасного соединения по HTTPS или по WSS. В результате генерируется общий секретный ключ, который используется для шифрования данных. Для дальнейшего взаимодействия клиент получает обязательные идентификаторы: одноразовую метку **nonce**, уникальный идентификатор пользователя **user\_id**, токен авторизации **token** и идентификатор канала событий **events\_channel**.

При старте происходит обмен RSA-ключами, используется парольная фраза для создания HMAC-подписи в формате Base64.

Процесс регистрации можно разбить на три этапа:

#### 1. Инициализация:

- Клиент генерирует RSA-ключи, из публичного ключа и парольной фразы создаёт подпись.
- Отправляет запрос с методом **start**, передавая:
  - **user**: имя пользователя.
  - **public\_key**: публичный ключ в формате pem.
  - **signature**: подпись в формате Base64.
- Сервер проводит проверку имени пользователя и подписи следующим образом: используя полученный открытый ключ и заранее известную серверу парольную фразу, вычисляется контрольная подпись, которая сравнивается с переданной клиентом подписью. Если проверка успешна, сервер отправляет клиенту объект результата, включающий в себя:
  - **nonce**: уникальный идентификатор для авторизации (действителен в течение 30 с).
  - **server\_public\_key\_b64**: публичный ключ сервера, зашифрованный публичным ключом клиента (в формате Base64).

#### 2. Авторизация:

- Клиент отправляет запрос с методом **login**, передавая:
  - **nonce**: полученный ранее идентификатор.
  - **encrypted\_password\_b64**: пароль пользователя, зашифрованный публичным ключом сервера (в формате Base64).
- Сервер проверяет данные и возвращает:
  - **token**: токен сессии для аутентификации последующих запросов (действителен в течение 20 мин).
  - **events\_channel**: канал для подключения к WebSocket.

#### 3. Поддержание сессии (heart-beat):

- Для продления времени действия **token** клиент отправляет запрос с методом **renew\_connection**, используя текущий **token**.

- Сервер подтверждает активность сессии, сбрасывая таймер (30 с), после чего клиент должен повторить отправку запроса.
- Если запрос не был отправлен в указанный промежуток времени, требуется повторное выполнение процедуры авторизации для получения нового действительного **token**.

### 3.2 Взаимодействие с WebSocket- подключением

WebSocket-подключение используется для получения уведомлений от сервера, таких как обновления данных и команды обработки запросов. Это позволяет клиенту своевременно реагировать на изменения, происходящие на серверной стороне.

Установление соединения.

Клиент подключается к WebSocket-подключению, используя **events\_channel**, полученный на этапе авторизации. В данном случае вместо {protocol} необходимо писать ws или wss в зависимости от того используется защищённый протокол или нет.

```
ws_url = f"{protocol}://{server}/{events_channel}"
```

Обработка сообщений.

Сервер отправляет JSON-объект, содержащий параметры **category, type, action**.

Пример сообщения:

```
1:  "jsonrpc": "2.0",
2:    "params": {
3:      "category": "control",
4:      "msg_id": "5114ad6e-b1cd-4c09-b8a0-3695c4522098",
5:      "date_time": "2025-07-30T15:02:11.4001300Z",
6:      "comp_id": "c849386f-4812-425e-be6d-b853b6d30f1c",
7:      "type": "company",
8:      "id": "23245ffa-50f0-4655-83ae-fed45cb09401",
9:      "action": "create",
10:     "link_type": "companies",
11:     "link_id": "34bbfcbc-5264-4bbb-b922-e2b121078cec"
12:  }
```

### 3.3 Взаимодействие с запросами

Клиент отправляет запросы к серверу для получения данных, внесения изменений и управления элементами.

#### 3.3.1 Структура запроса JSON-RPC

Каждый запрос клиента соответствует стандарту JSON-RPC 2.0 и состоит из следующих полей:

Структура запроса:

- jsonrpc: Версия протокола (всегда «2.0»).

- **method**: Название метода, выполняемого на сервере.
- **params**: Параметры метода, включающие:
  - Данные для выполнения запроса (например: **type**, **id**, **link\_type**).
  - Аутентификационные (например: **security** с **token**).
- **id**: Идентификатор запроса.

#### Пример запроса:

```

1:  "jsonrpc": "2.0",
2:    "method": "get_elements",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "car"
8:    },
9:    "id": 6

```

#### Типы запросов:

- Запросы управления
  - **start**, **login**, **renew\_connection**: Управление сессией и авторизацией.
- Запросы данных
  - **get\_elements**: Получение всех элементов указанного типа.
  - **get\_elements\_by**: Получение элементов, связанных с передаваемым.
  - Служебные (**get\_incident\_types**, **get\_message\_groups**): Получение специализированных данных.
- Создание, редактирование элементов
  - **create\_element**, **delete\_element**: Создание, удаление элемента указанного типа.
  - **update\_element**: Изменение данных элемента.
  - **create\_link**, **delete\_link**: Создание связи между элементами.
- Связанные с поиском
  - **find\_elements\_by\_param\_i**, **find\_elements\_by\_param\_s**, **find\_elements\_by\_any\_param\_s**: Поиск элементов по целочисленному или строковому значению указанного параметра.
  - **find\_elements\_by\_name**: Поиск элементов по имени.
  - **find\_elements\_by\_additional\_param\_s**: Поиск пользователя по ключу

#### Обработка ответа.

Сервер возвращает JSON-объект с полями:

- **result**: Результат выполнения метода (объект, массив или значение).
- **error**: Описание ошибки, если запрос неуспешен (содержит сообщение).
- **id**: идентификатор запроса.

#### Пример ответа:

```
1:  "jsonrpc": "2.0",
2:    "id": 6,
3:    "result": {
4:      "id": "3ae19ed4-51e6-40f2-8bd8-468498fd8f0e",
5:      "name": "Автомобиль",
6:      "params": {
7:        "color": "red",
8:        "comment": "",
9:        "model": "lada vesta",
10:       "registration_number": "777",
11:       "vin": ""
12:     },
13:     "additional_params": []
14: }
```

### 3.3.1.1 Запросы Batch Requests

Сервер поддерживает пакетные запросы (Batch Requests). Клиент может отправить массив из нескольких объектов.

#### Пример запроса:

```
1:  {
2:    "jsonrpc": "2.0",
3:    "method": "get_elements",
4:    "params": {
5:      "type": "potok_3n_v103",
6:      "security": {
7:        "token": "705e9137-7aa7-4d8f-9bc4-2197c45a0e49"
8:      }
9:    },
10:   "id": 54
11: },
12: {
13:   "jsonrpc": "2.0",
14:   "method": "get_elements",
15:   "params": {
16:     "type": "potok_3n",
17:     "security": {
18:       "token": "705e9137-7aa7-4d8f-9bc4-2197c45a0e49"
19:     }
20:   },
21:   "id": 55
22: },
```

```

23:
24:  {
25:    "jsonrpc": "2.0",
26:    "method": "get_elements",
27:    "params": {
28:      "type": "uo_4s",
29:      "security": {
30:        "token": "705e9137-7aa7-4d8f-9bc4-2197c45a0e49"
31:      }
32:    },
33:    "id": 56
34:  }

```

### 3.3.2 Структура запросов с примерами

Все описания типов полей представлены в приложении А.

**start**

(начало сессии)

Запросом передаётся логин пользователя **user**, публичный ключ клиента **public\_key** и HMAC-подпись **signature**. Метод возвращает объект, содержащий значение **nonce** и зашифрованный пароль **server\_public\_key\_b64**.

Структура запроса:

- method: start
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - user: <логин пользователя>
  - public\_key: <публичный ключ клиента (в формате pem)>
  - signature: <HMAC-подпись, сгенерированная из публичного ключа клиента и парольной фразы (в формате Base64)>
  - version: 1.0

**Пример:**

```

1:  "jsonrpc": "2.0",
2:    "method": "start",
3:    "params": {
4:      "user": "admin",
5:      "public_key": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAnl4s4GRShjsetAQIie1A\nw
T3ndFBKZ+hY8ChNEDqCNEYs+yvvqzM7jZaQTmBUCDV9kfp6+1d/RqLEUC3l18Rn\nneFn5
RmjYv8gHjuvJZ0v/CaSb1nLuH5PINZT/wQy/oZeQVKo9I3Rfw5qcW2XioGpX\nnd3I4qIh
uA7zNoVIgQ0iWxan2stbbxaHRgiOHbVHeHb5lzpZIm7FT0I+LI3HEX6JZ\nnQBkqL3NuMS
rA70mm2dFwC8v2eCE7BrJ3NGtFmW3t+TkUD06wlK+j0PaBwfa00Pv2\nnhre5bAhg8qclp
kC4V8d3LtlIRrRY+ggkmsEVsILhw2TURNPV2VEHiN/lz5nrt9v\nnyQIDAQAB\n-----
END PUBLIC KEY-----\n",

```

```
6:     "signature": "w5mqolfiWacpAjdc3uzh69iljLPfyAmK8YfWQy5eK+A=",
7:     "version": "1.0"
8:   },
9:   "id": 1
```

#### Ответ сервера:

```
1:   "jsonrpc": "2.0",
2:   "id": 1,
3:   "result": {
4:     "nonce":
"vT37RBKwISvox90RPEv6mhrEzY1IM6X0P9l0Ay50PDVX0idGmW4Nubndw6YRwNPF",
5:     "server_public_key_b64":
"oJduX7DrbCiSFLwQHo8R4kV5b6l4DyC/WOdHWVR5pTh4gf/JQBjy2z1sdrkJJ5sFiDV
BMs59UamZhm5ZNIx5L3vuJJx6VLIBM9t53mo2nEm/JTYvMdpY3LfMeEAF/faCoziwzzsH
KWFpqwDdSeq6a+5MXTz9Ll/TJ5nDseL6C5GWhRe/t/vY5uII8dmzDSAz+f4noVPaiLYaC
rtThRRbjzRqYnjlAonmFg2AHkSi/OacqNnuYH0miLKS24KHs0zDwITusy/e4pqmoaMcy
VGoMh6rAqUH9aH65b0q9C/kgKTD7sCNbXNIWv7dqsscE2sGUS8rrTFAOemQiEYysfAAB6
EfPTYOUPODFaNLz3lfYJeFmqM1ypua02Xr6vS+ATY/gWc7NkriwJl5eOB7rB4+U5OymvL
OSZjZyVUIR3wmXW0X83v+Vb5MEcbWb9rm53W6qh5qDaqfoxCBi/DdwUF5QI19v+pM3aPR
YTuK09ZvLOWs4v+V/JEwuu+d5iXDnt+uKjabc9UImpVM1lGVvu2OqQqWt8IBBZltJctgo
Rvps+Xoa068eJXcWaIutEu8Yj1/Wodpu0erfEcPOpu7Q5Z6M5StpBp+vibz55vkeia/vA
y1EoiI7IOgdjLhCO3Kx92YZMYGxBxt6x/sVXNLYZbmUP8/eclY280jJDC2Yedc/TRBSm
0pfmeBodJ7Ecgny+7jhaKyZOjRppULsovZXRxqgvT+cEdGTnwfENB/V3YOFvXHblE0Ja
r+VbNfZxpSY+c+M9uBXSUIo2PyhxUqE6OXGfy1TiqvP2z8NHNaxgVr6nRDtoLE7P+...
...xA6D1ZoMl0BAKU2PEb/DbMNUZ7bChKcdv7JHh5sHnFPo9DUeTFw/h4WoaxJunzoUM5fO
g3M1aAeONzLXG840wNDFMV/rAOB05uKOZzdH86vreejW395XxkAWst5WnKULX9wxMSAdf
GilBhY6m9WwpoCpPjppuSGDdY8X3EJkcTHEm4qj6YkdcEc9+EqP6bOiHrYd1qCp/wWlSU
Zg9ZdZtwFUm3LbFr/toMRi5wUth/oux1z1InNvzJuI=",
6:     "current_version": "1.0"
7:   }
```

### login

#### (авторизация)

Клиентом отправляются значения **nonce** и зашифрованный пароль **encoded\_password\_b64**. Ответ метода включает объект с полями **token** и **events\_channel**. Срок действия полученных значений **token** и **events\_channel** ограничен 20 минутами.

Структура запроса:

- method: login
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - nonce: <полученный при старте>

- encrypted\_password\_b64: <пароль пользователя, зашифрованный публичным ключом сервера>

#### Пример:

```
1: "jsonrpc": "2.0",
2:   "method": "login",
3:   "params": {
4:     "nonce":
5:       "vT37RBKwISvox90RPeV6mhrEzY1IM6X0P9l0Ay50PDVX0idGmW4Nubndw6YRwNPF",
6:     "encrypted_password_b64":
7:       "p9x2HYFcKb8af2LYaB28zs+NUlRnvKJWY/yBuXyOgPNVYxc2UH3Wn4oQIzE9YmHaUOuh
TWyA/c0VsNLMIHl+Nvq8he6OpuzDRVoBEEFnNfwoFiacPZ+QecWQhl4B6m0lr86KVDQft
V+ImX7GwUjM3yhUXUD3OjxNbm2GRM+ZYcPrY34byvygkhsLWMkhhF7YF1Ln7xGAZueHtQ
791PD+S3PZDDtt343h93DKfD4Z8HCcS/WffFAH4cFHx51BizDhJpwa2q8m3avEflnq4wT
nTO5xRYAiJ5smh/X9fYypD1BalsohRTHk0apK4BcJOTBZuOCXAoBp7mM0widluQRqbA==
"
8:   },
9:   "id": 2
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2:   "id": 3,
3:   "result": {
4:     "token": "ae067f28-bb67-46aa-847f-066ff634bb78",
5:     "events_channel": "87ce43e0-f03f-4986-b04d-bf809fa95e68"
6:   }
```

### renew\_connection

#### (поддержание сессии)

Чтобы продлить срок действия **token** и **events\_channel**, выполняется соответствующий запрос. В случае успешного завершения сервер возвращает значение **true** и увеличивает продолжительность сессии ещё на 30 с, после чего процедуру необходимо повторить заново. В случае результата **false** значит, что **token** более недействителен и необходимо заново подключиться.

Структура запроса:

- method: renew\_connection
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
    - user: <логин пользователя>

#### Пример:

```
1: "jsonrpc": "2.0",
2:   "method": "renew_connection",
```

```
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78",
6:       "type": "user"
7:     }
8:   },
9:   "id": 3
```

#### Ответ сервера:

```
1:   "jsonrpc": "2.0",
2:   "id": 3,
3:   "result": true
```

## ЗАПРОСЫ ДАННЫХ

### get\_computers

(получение списка идентификаторов компьютеров)

Возвращает список идентификаторов компьютеров **comp\_id**.

Структура запроса:

- method: get\_computers
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>

#### Пример:

```
1:   "jsonrpc": "2.0",
2:   "method": "get_computers",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     }
7:   },
8:   "id": 4
```

#### Ответ сервера:

```
1:   "jsonrpc": "2.0",
2:   "id": 4,
3:   "result": [
4:     {
5:       "comp_id": "c849386f-4812-425e-be6d-b853b6d30f1c",
6:       "name": "orion_x"
7:     }
8:   ]
```

## get\_elements

### (получение информации об элементах указанного типа)

Возвращает список объектов указанного типа со всеми свойствами и дополнительными таблицами. Здесь и далее под дополнительными таблицами понимаются таблицы с данными самого пользователя.

Структура запроса:

- method: get\_elements
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <запрашиваемый тип>

#### Пример:

```
1: "jsonrpc": "2.0",
2:   "method": "get_elements",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "car"
8:   },
9:   "id": 5
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2:   "id": 5,
3:   "result": [
4:     {
5:       "id": "3ae19ed4-51e6-40f2-8bd8-468498fd8f0e",
6:       "name": "Автомобиль",
7:       "params": {
8:         "color": "red",
9:         "comment": "",
10:        "model": "lada vesta",
11:        "registration_number": "777",
12:        "vin": ""
13:       },
14:       "additional_params": []
15:     }
16:   ]
```

## get\_computer\_elements

(получение информации об элементах указанного типа для идентификатора компьютера)

Возвращает список объектов указанного типа со всеми свойствами и дополнительными таблицами, связанными с передаваемым **comp\_id**.

Структура запроса:

- method: get\_computer\_elements
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <запрашиваемый тип>
  - comp\_id: <идентификатор компьютера>

### Пример:

```
1:  "jsonrpc": "2.0",
2:    "method": "get_computer_elements",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "car",
8:      "comp_id": "c849386f-4812-425e-be6d-b853b6d30f1c"
9:    },
10:   "id": 6
```

### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:    "id": 6,
3:    "result": [
4:      {
5:        "id": "3ae19ed4-51e6-40f2-8bd8-468498fd8f0e",
6:        "name": "Автомобиль",
7:        "params": {
8:          "color": "red",
9:          "comment": "",
10:         "model": "lada vesta",
11:         "registration_number": "777",
12:         "vin": ""
13:       },
14:       "additional_params": []
15:     }
16:   ]
```

## get\_element

### (получение информации об элементе)

Отвечает объектом нужного типа **type** и идентификатора **id**, содержащим все свойства и дополнительные таблицы, указанные в списке **additional\_params**. Эти таблицы служат хранилищем данных о ключах и личной документации пользователей, включая паспортные данные, заграничный паспорт, военный билет, водительское удостоверение, идентификационный номер налогоплательщика (ИНН) и страховой номер индивидуального лицевого счёта (СНИЛС).

Структура запроса:

- method: get\_element
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип запрашиваемого элемента>
  - id: <идентификатор запрашиваемого элемента>

#### Пример:

```
1: "jsonrpc": "2.0",
2:   "method": "get_element",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "car",
8:     "id": "3ae19ed4-51e6-40f2-8bd8-468498fd8f0e"
9:   },
10:  "id": 7
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2:   "id": 7,
3:   "result": {
4:     "id": "3ae19ed4-51e6-40f2-8bd8-468498fd8f0e",
5:     "name": "Автомобиль",
6:     "params": {
7:       "color": "red",
8:       "comment": "",
9:       "model": "lada vesta",
10:      "registration_number": "777",
11:      "vin": ""
12:     },
13:     "additional_params": []
14:   }
```

Ответ, содержащий дополнительные таблицы:

```
1:  "jsonrpc": "2.0",
2:    "id": 19,
3:    "result": {
4:      "id": "1731ce64-5e04-4193-a0f4-db55fbb84390",
5:      "name": "Поль Зователь ",
6:      "params": {
7:        "additional_work_phone": "",
8:        ...
9:        "user_name": "Зователь",
10:       "work_phone": ""
11:     },
12:     "additional_params": [
13:       {
14:         "name": "passport",
15:         "params": [
16:           {
17:             "authority": "Полномочия",
18:             "date_of_issue": "2020-01-01T00:00:00.0000000Z",
19:             "passport_number": "123456",
20:             "passport_series": "1234",
21:             "place_of_birth": "Королёв",
22:             "sex": 1,
23:             "subdivision_code": "123-456"
24:           }
25:         ]
26:       },
27:       {
28:         "name": "military_id",
29:         "params": []
30:       },
31:       {
32:         "name": "international_passport",
33:         "params": [
34:           {
35:             "authority": "Полномочия",
36:             "date_of_expiry": "2031-01-01T00:00:00.0000000Z",
37:             "date_of_issue": "2021-01-01T00:00:00.0000000Z",
38:             "passport_number": "2345",
39:             "passport_series": "01",
40:             "sex": 1,
41:             "subdivision_code": "123"
42:           }
43:         ]

```

```

44:     },
45:     {
46:         "name": "keys",
47:         "params": []
48:     },
49:     {
50:         "name": "snils",
51:         "params": []
52:     },
1:     {
2:         "name": "inn",
3:         "params": []
4:     },
5:     {
6:         "name": "driver_license",
7:         "params": [
8:             {
9:                 "authority": "",
10:                "date_of_expiry": "2032-01-01T00:00:00.0000000Z",
11:                "date_of_issue": "2022-01-01T00:00:00.0000000Z",
12:                "document_number": "",
13:                "document_series": "",
14:                "place_of_issue": "",
15:                "sex": 1,
16:                "subdivision_code": "",
17:                "vehicle_category": "Б"
18:            }
19:        ]
20:     }
21: ]
22: }

```

### **get\_elements\_by\_link**

**(получение информации об элементах, связанных с передаваемым)**

Возвращает список объектов указанного типа со всеми свойствами и дополнительными таблицами.

Структура запроса:

- method: get\_elements\_by\_link
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элемента, связь с которым запрашивается>

- id: <элемент, связь с которым запрашивается>
- link\_type: <тип запрашиваемых элементов>
- left: true/false (тип связи между элементами)

Возможные связи см. в Приложении Б «Связи между типами».

#### Пример:

```

1:  "jsonrpc": "2.0",
2:    "method": "get_elements_by_link",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "logic_zone",
8:      "id": "alf9ed32-bef5-4721-841d-ed5c4bd25a94",
9:      "link_type": "orion_reader",
10:     "left": true
11:   },
12:   "id": 8

```

#### Ответ сервера:

```

1:  "jsonrpc": "2.0",
2:    "id": 8,
3:    "result": [
4:      {
5:        "id": "73db82c3-bfa1-4ccc-b371-6e7b9546b5fe",
6:        "name": "СЧИТЫВАТЕЛЬ",
7:        "params": {
8:          "contact_id": 0,
9:          "number": 1,
10:         "type_id": 1
11:        },
12:        "additional_params": []
13:      }
14:    ]

```

### **get\_incident\_types**

**(получение списка типов инцидентов)**

Запрос возвращает список объектов.

Структура запроса:

- method: get\_incident\_types
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>

#### Пример:

```
1:  "jsonrpc": "2.0",
2:    "method": "get_incident_types",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      }
7:    },
8:    "id": 9
```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:    "id": 9,
3:    "result": [
4:      {
5:        "number": 0,
6:        "id": "start",
7:        "name": "Пуски",
8:        "text": "Инцидент в категории \"Пуски\""
9:      },
10:     ...
11:  ]
```

### **get\_message\_groups**

#### **(получение списка групп сообщений)**

Запрос возвращает список объектов.

Структура запроса:

- method: get\_message\_groups
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>

#### Пример:

```
1:  "jsonrpc": "2.0",
2:    "method": "get_message_groups",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      }
7:    },
8:    "id": 10
```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:    "id": 10,
3:    "result": [
4:      {
5:        "number": 1,
6:        "id": "blink_mask",
7:        "text": "Маски мигания реле",
8:        "color": "#fff8a15a",
9:        "text_color": "#ffffffff"
10:     },
11:     ...
12:  ]
```

#### **get\_messages\_desc**

#### **(получение описания сообщений)**

Возвращает список объектов с описанием сообщений для указанного типа.

Структура запроса:

- method: get\_messages\_desc
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элементов, для которых запрашивается описание событий>

#### Пример:

```
1:  "jsonrpc": "2.0",
2:    "method": "get_messages_desc",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "user"
8:    },
9:    "id": 11
```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:    "id": 11,
3:    "result": [
4:      {
5:        "action": 449,
6:        "text": "Изменение комментария по инциденту",
```

```
7:         "hide": true,
8:         "log": true,
9:         "color": "#00ffffff",
10:        "text_color": "#ff000000",
11:        "group_number": 51,
12:        "incident_type_number": 4294967295
13:    },
14:    ...
15: ]
```

### **find\_elements\_by\_param\_i**

(поиск элементов по целочисленному значению)

Запрос осуществляет поиск элементов по числовому значению заданного атрибута внутри определённого типа элементов. Результатом является перечень обнаруженных объектов с указанием их характеристик.

Структура запроса:

- method: find\_elements\_by\_param\_i
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элементов, среди которых выполняется поиск>
  - param\_name: <параметр для поиска> (допускаются только параметры, хранящие целочисленный тип)
  - param\_value: <значение для поиска> (int)
  - additional: true/false (поиск в дополнительных таблицах)

#### **Пример:**

```
1: "jsonrpc": "2.0",
2:   "method": "find_elements_by_param_i",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "user",
8:     "param_name": "fired",
9:     "param_value": 1,
10:    "additional": false,
11:  },
12:  "id": 12
```

#### **Ответ сервера:**

```
1: "jsonrpc": "2.0",
```

```

2:   "id": 12,
3:   "result": [
4:     {
5:       "id": "bb26eef8-7b60-4cf2-9cd2-d6edeb736193",
6:       "name": "Поль Зователь ",
7:       "params": {
8:         "additional_work_phone": "300",
9:         ...
10:        "black_list": 1,
11:        "black_list_cause": "",
12:        ...
13:        "user_name": "Зователь",
14:        "work_phone": "+74957757155"
15:      },
16:      "additional_params": [
17:        {
18:          "name": "keys",
19:          "params": [
20:            {
21:              "key_code": "1234",
22:              "locked": 0,
23:              "type_key": 0
24:            }
25:          ]
26:        }
27:      ]
28:    },
29:    ...
30:  ]

```

### **find\_elements\_by\_param\_s**

#### **(поиск элементов по строковому значению)**

Выполняет поиск элементов по строке заданного параметра среди объектов указанного типа. Результат представляет собой список найденных объектов с отображением их атрибутов.

Структура запроса:

- method: find\_elements\_by\_param\_s
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элементов, среди которых выполняется поиск>
  - param\_name: <параметр для поиска> (допускаются только параметры, хранящие строки)
  - param\_value: <значение для поиска> (str)

- `additional: true/false` (поиск в дополнительных таблицах)
- `case_insensitive: true/false` (игнорировать регистр)
- `exact: true/false` (искать точное совпадение)

#### Пример:

```
1:  "jsonrpc": "2.0",
2:    "method": "find_elements_by_param_s",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "user",
8:      "param_name": "email",
9:      "param_value": "pochta",
10:     "additional": false,
11:     "case_insensitive": true,
12:     "exact": false
13:   },
14:   "id": 13
```

#### Ответ сервера:

```
31: "jsonrpc": "2.0",
32:   "id": 13,
33:   "result": [
34:     {
35:       "id": "a65eaeb1-3241-41f3-86d2-13035e07c9c6",
36:       "name": "Иванов Иван Иванович",
37:       "params": {
38:         "additional_work_phone": "437",
39:         ...
40:         "email": "pochta@pochta.ru",
41:         ...
42:         "work_phone": "+74957757155"
43:       },
44:       "additional_params": [
45:         {
46:           "name": "keys",
47:           "params": [
48:             {
49:               "key_code": "2345",
50:               "locked": 0,
51:               "type_key": 0
52:             },
53:             {
54:               "key_code": "3456",
55:               "locked": 0,
56:               "type_key": 0
57:             }
58:           ]
59:         }
60:       ]
61:     },
62:     ...
63:   ]
```

#### **find\_elements\_by\_any\_param\_s**

**(поиск элементов по строковому значению любого параметра)**

Выполняет поиск элементов по строке заданного параметра среди объектов указанного типа.

Результат представляет собой список найденных объектов с отображением всех их атрибутов.

Структура запроса:

- method: find\_elements\_by\_any\_param\_s
- jsonrpc: 2.0
- id: <номер запроса>
- params:

- security:
  - token: <полученный при логине>
- type: <тип элементов, среди которых выполняется поиск>
- param\_value: <значение для поиска> (str)
- additional: true/false (поиск в дополнительных таблицах)
- case\_insensitive: true/false (игнорировать регистр)
- exact: true/false (искать точное совпадение)

#### Пример:

```

1:  "jsonrpc": "2.0",
2:    "method": "find_elements_by_any_param_s",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "user",
8:      "param_value": "pochta",
9:      "additional": false,
10:     "case_insensitive": true,
11:     "exact": false
12:   },
13:   "id": 14

```

#### Ответ сервера:

```

1:  "jsonrpc": "2.0",
2:    "id": 14,
3:    "result": [
4:      {
5:        "id": "a65eaeb1-3241-41f3-86d2-13035e07c9c6",
6:        "name": "Иванов Иван Иванович",
7:        "params": {
8:          "additional_work_phone": "437",
9:          ...
10:         "email": "pochta@pochta.ru",
11:         ...
12:         "work_phone": "+74957757155"
13:       },
14:       "additional_params": [
15:         {
16:           "name": "keys",
17:           "params": [
18:             {
19:               "key_code": "2345",
20:               "locked": 0,
21:               "type_key": 0

```

```

22:         },
23:         {
24:             "key_code": "3456",
25:             "locked": 0,
26:             "type_key": 0
27:         }
28:     ]
29: }
30: ]
31: },
32: ...
33: ]

```

### **find\_elements\_by\_name** (поиск элементов по имени)

Поиск элементов по имени среди указанного типа элементов. Возвращает список найденных объектов со всеми свойствами.

Структура запроса:

- method: find\_elements\_by\_name
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элементов, среди которых выполняется поиск>
  - name: <значение для поиска> (str)
  - case\_insensitive: true/false (игнорировать регистр)
  - exact: true/false (искать точное совпадение)

**Пример:**

```

1: "jsonrpc": "2.0",
2:   "method": "find_elements_by_name",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "user",
8:     "name": "Иван",
9:     "case_insensitive": false,
10:    "exact": false
11:  },
12:  "id": 15

```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:    "id": 15,
3:    "result": [
4:      {
5:        "id": "a65eaeb1-3241-41f3-86d2-13035e07c9c6",
6:        "name": "Иванов Иван Иванович",
7:        "params": {
8:          "additional_work_phone": "437",
9:          ...
10:         "work_phone": "+74957757155"
11:       },
12:       "additional_params": [
13:         {
14:           "name": "keys",
15:           "params": [
16:             {
17:               "key_code": "2345",
18:               "locked": 0,
19:               "type_key": 0
20:             },
21:             {
22:               "key_code": "3456",
23:               "locked": 0,
24:               "type_key": 0
25:             }
26:           ]
27:         }
28:       ]
29:     },
30:     ...
31:   ]
```

### **find\_elements\_by\_additional\_param\_s**

(поиск пользователя по ключу)

Поиск элементов по строковому значению (ключа) в дополнительной таблице. Возвращает список найденных объектов со всеми свойствами.

Структура запроса:

- method: find\_elements\_by\_additional\_param\_s
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>

- type: <тип элементов, среди которых выполняется поиск>
- additional\_params\_name: "keys" (название доп.таблицы)
- additional\_param\_name: "key\_code" (название поля)
- additional\_param\_value: <значение для поиска> (str)
- case\_insensitive: true/false (игнорировать регистр)
- exact: true/false (искать точное совпадение)

#### Пример:

```

1:  "jsonrpc": "2.0",
2:    "method": "find_elements_by_additional_param_s",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "user",
8:      "additional_params_name": "keys",
9:      "additional_param_name": "key_code",
10:     "additional_param_value": "9",
11:     "case_insensitive": false,
12:     "exact": false
13:   },
14:   "id": 16

```

#### Ответ сервера:

```

1:  "jsonrpc": "2.0",
2:    "id": 16,
3:    "result": [
4:      {
5:        "id": "a65eaeb1-3241-41f3-86d2-13035e07c9c6",
6:        "name": "Иванов Иван Иванович",
7:        "params": {
8:          "additional_work_phone": "437",
9:          ...
10:         "work_phone": "+74957757155"
11:       },
12:       "additional_params": [
13:         {
14:           "name": "keys",
15:           "params": [
16:             {
17:               "key_code": "5678",
18:               "locked": 0,
19:               "type_key": 0
20:             },
21:             {

```

```

22:         "key_code": "9999",
23:         "locked": 0,
24:         "type_key": 0
25:     }
26: ]
27: }
28: ]
29: },
30: ...
31: ]

```

### types\_by\_tag

#### (получение списка типов для прототипа)

Возвращает список типов, соответствующих указанному **tag** (прототипу).

Структура запроса:

- method: types\_by\_tag
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - tag: <прототип> (orion\_device)

#### Пример:

```

1: "jsonrpc": "2.0",
2:   "method": "types_by_tag",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "tag": "orion_device"
8:   },
9:   "id": 17

```

#### Ответ сервера:

```

1: "jsonrpc": "2.0",
2:   "id": 17,
3:   "result": [
4:     "user_device",
5:     "unknown_device",
6:     "rupor_75",
7:     ...
8:     "signal_20",
9:     "s2000m"

```

## **get\_events** (получение истории событий)

Возвращает список объектов событий.

Структура запроса:

- method: get\_events
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - date\_time: <дата и время начала выборки> (в формате «2025-05-19T00:00:00.0000000Z»)
  - count: <количество событий в выборке> (int, максимальное количество в одном запросе – 1000)
  - direction: «+/-» (направление выборки относительно date\_time: если «+», то выбираются события после указанного времени, если «-», то выборка идёт в обратном направлении)

**Пример:**

```

1:  "jsonrpc": "2.0",
2:    "method": "get_events",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "date_time": "2025-08-13T04:00:00.0000000Z",
8:      "count": 2,
9:      "direction": "+"
10:   },
11:   "id": 18

```

**Ответ сервера:**

```

1:  "jsonrpc": "2.0",
2:    "id": 18,
3:    "result": [
4:      {
5:        "category": "regular",
6:        "msg_id": "31d8c965-a33b-4adc-ac92-72ffdb65857a",
7:        "date_time": "2025-08-13T05:19:25.7734710Z",
8:        "comp_id": "c849386f-4812-425e-be6d-b853b6d30f1c",
9:        "type": "shell",
10:       "id": "00000000-0000-0000-0000-000000000000",
11:       "action": 1043
12:     },

```

```

13:     {
14:         "category": "regular",
15:         "msg_id": "12f47fe6-cf00-478d-856d-813c0b845922",
16:         "date_time": "2025-08-13T05:19:36.9078690Z",
17:         "comp_id": "c849386f-4812-425e-be6d-b853b6d30f1c",
18:         "type": "core",
19:         "id": "00000000-0000-0000-0000-000000000000",
20:         "action": 450
21:     }
22: ]

```

## СОЗДАНИЕ, УДАЛЕНИЕ, РЕДАКТИРОВАНИЕ

### create\_element

#### (создание элемента)

Создаёт элемент указанного типа с указанным именем (создаётся в корне). Возвращает созданный объект.

Примечание – Доступно для определённых типов: car, company и position.

Структура запроса:

- method: create\_element
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип создаваемого элемента>
  - name: <имя создаваемого элемента>
  - params: { <список параметров со значениями> } (может быть пустой)
  - additional params: [ <объекты с дополнительными параметрами> ] (может быть пустой)

#### Пример:

```

1:  "jsonrpc": "2.0",
2:  "method": "create_element",
3:  "params": {
4:    "security": {
5:      "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:    },
7:    "type": "car",
8:    "name": "Новый автомобиль",
9:    "params": {},
10:   "additional_params": []
11: },
12:  "id": 19

```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:  "id": 19,
3:  "result": {
4:    "id": "43f9eecc-da5c-4529-b7f3-b385289643ab",
5:    "name": "Новый автомобиль",
6:    "params": {},
7:    "additional_params": []
8:  }
```

#### **create\_element\_by\_link**

##### **(создание дочернего элемента)**

Создаёт элемент указанного типа с указанным именем под выбранным элементом. Возвращает созданный объект.

Примечание – Доступно только для типов user и department.

Структура запроса:

- method: create\_element\_by\_link
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип создаваемого элемента>
  - name: <имя создаваемого элемента>
  - params: { <список параметров со значениями> } (может быть пустой)
  - additional params: [ <объекты с дополнительными параметрами> ] (может быть пустой)
  - link\_type: <тип родительского элемента>
  - link\_id: <id родительского элемента>

#### Пример:

```
1:  "jsonrpc": "2.0",
2:  "method": "create_element_by_link",
3:  "params": {
4:    "security": {
5:      "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:    },
7:    "type": "department",
8:    "name": "Отдел под Новым отделом",
9:    "link_type": "department",
10:   "link_id": "43f9eecc-da5c-4529-b7f3-b385289643ab",
11:   "params": {},
12:   "additional_params": []
13: },
```

```
14: "id": 20
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2: "id": 20,
3: "result": {
4:   "id": "09aae008-3b52-42b5-8e08-7411aaead563",
5:   "name": "Отдел под Новым отделом",
6:   "params": {},
7:   "additional_params": []
8: }
```

### **delete\_element**

#### **(удаление элемента)**

Данный запрос удаляет выбранный элемент. В случае успешной операции возвращается значение **true**.

#### Примечания:

1. Доступно для определённого списка типов: car, company, department, user и position. Недоступно для приборов и их элементов.
2. В текущей реализации недоступно удаление суперпользователя и текущего пользователя системы.

#### Структура запроса:

- method: delete\_element
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элемента>
  - id: <id элемента>

#### Пример:

```
1: "jsonrpc": "2.0",
2: "method": "delete_element",
3: "params": {
4:   "security": {
5:     "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:   },
7:   "type": "department",
8:   "id": "aec4e4b3-4b95-47f0-a146-355719a52899"
9: },
10: "id": 21
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2:   "id": 21,
3:   "result": true
```

### **update\_element** (изменение элемента)

Этот запрос обновляет заданные характеристики конкретного элемента и возвращает изменённый объект.

Структура запроса:

- method: update\_element
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элемента>
  - id: <идентификатор запрашиваемого элемента>
  - name: <имя (или новое имя) элемента>
  - params: { <список изменяемых параметров со значениями> } (может быть пустой)
  - additional params: [ <изменяемые объекты с дополнительными параметрами> ] (может быть пустой)

Примечание – При изменении дополнительных параметров перезаписываются все поля объекта из additional\_params. Если требуется изменить поле "passport\_number" в объекте "passport", то нужно заполнить все поля, иначе они будут перезаписаны пустыми значениями. При создании нового ключа, либо изменении или удалении существующего, необходимо отправить полный обновлённый перечень ключей, поскольку система обновит весь список ключей целиком.

#### Пример (изменение значения поля "name"):

```
1: "jsonrpc": "2.0",
2:   "method": "update_element",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "department",
8:     "id": "43f9eecc-da5c-4529-b7f3-b385289643ab",
9:     "name": "Первый сектор",
10:    "params": {},
11:    "additional_params": []
12:  },
13:   "id": 22
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2:   "id": 22,
```

```
3:   "result": {
4:     "id": "43f9eecc-da5c-4529-b7f3-b385289643ab",
5:     "name": "Первый сектор",
6:     "params": {},
7:     "additional_params": []
8:   }
```

#### Пример (изменения в additional\_params):

```
1:   "jsonrpc": "2.0",
2:   "method": "update_element",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "user",
8:     "id": "35882fd7-6d4f-4660-968d-145d3ac01e92",
9:     "name": "Иванов Иван Иванович",
10:    "params": {},
11:    "additional_params": [
12:      {
13:        "name": "driver_license",
14:        "params": [
15:          {
16:            "authority": "OVD",
17:            "date_of_expiry": "2060-01-17T00:00:00.0000000Z",
18:            "date_of_issue": "1999-01-01T00:00:00.0000000Z",
19:            "document_number": "222222",
20:            "document_series": "1111",
21:            "place_of_birth": "city",
22:            "place_of_issue": "city",
23:            "vehicle_category": "B, C "
24:          }
25:        ]
26:      },
27:      {
28:        "name": "inn",
29:        "params": [
30:          {
31:            "inn": "111111111122"
32:          }
33:        ]
34:      },
35:      {
36:        "name": "international_passport",
```

```
37:         "params": [
38:             {
39:                 "authority": "OVD",
40:                 "date_of_expiry": "2030-01-07T00:00:00.0000000Z",
41:                 "date_of_issue": "2004-01-13T00:00:00.0000000Z",
42:                 "passport_number": "2222222",
43:                 "passport_series": "11",
44:                 "place_of_birth": "city"
45:             }
46:         ]
47:     },
48:     {
49:         "name": "keys",
50:         "params": [
51:             {
52:                 "key_code": "4444",
53:                 "locked": 0,
54:                 "type_key": 0
55:             }
56:         ]
57:     },
58:     {
59:         "name": "military_id",
60:         "params": [
61:             {
62:                 "authority": "ovd",
63:                 "date_of_issue": "1980-01-01T00:00:00.0000000Z",
64:                 "document_number": "2222222",
65:                 "document_series": "11",
66:                 "military_rank": "пианист",
67:                 "mos": "1",
68:                 "place_of_birth": "city"
69:             }
70:         ]
71:     },
72:     {
73:         "name": "passport",
74:         "params": [
75:             {
76:                 "authority": "OVD",
77:                 "date_of_issue": "2003-01-16T00:00:00.0000000Z",
78:                 "passport_number": "222222",
79:                 "passport_series": "1111",
80:                 "place_of_birth": "city",
```

```
81:         "subdivision_code": "111-222"
82:     }
83: ]
84: },
85: {
86:     "name": "snils",
87:     "params": [
88:         {
89:             "snils": "11111111333"
90:         }
91:     ]
92: }
93: ]
94: },
95: "id": 22
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",
2:   "id": 22,
3:   "result": {
4:     "id": "35882fd7-6d4f-4660-968d-145d3ac01e92",
5:     "name": "Иванов Иван Иванович",
6:     "params": {
7:       "additional_work_phone": "444",
8:       ...
9:     },
10:    "additional_params": [
11:      {
12:        "name": "passport",
13:        "params": [
14:          {
15:            "authority": "OVD",
16:            "date_of_issue": "2003-01-16T00:00:00.0000000Z",
17:            "passport_number": "222222",
18:            "passport_series": "1111",
19:            "place_of_birth": "city",
20:            "subdivision_code": "111-222"
21:          }
22:        ]
23:      },
24:      {
25:        "name": "military_id",
26:        "params": [
27:          {
```

```
28:         "authority": "ovd",
29:         "date_of_issue": "1980-01-01T00:00:00.0000000Z",
30:         "document_number": "2222222",
31:         "document_series": "11",
32:         "military_rank": "пианист",
33:         "mos": "1",
34:         "place_of_birth": "city"
35:     }
36: ]
37: },
38: {
39:     "name": "international_passport",
40:     "params": [
41:         {
42:             "authority": "OVD",
43:             "date_of_expiry": "2030-01-07T00:00:00.0000000Z",
44:             "date_of_issue": "2004-01-13T00:00:00.0000000Z",
45:             "passport_number": "2222222",
46:             "passport_series": "11",
47:             "place_of_birth": "city"
48:         }
49:     ]
50: },
51: {
52:     "name": "keys",
53:     "params": [
54:         {
55:             "key_code": "4444",
56:             "locked": 0,
57:             "type_key": 0
58:         }
59:     ]
60: },
61: {
62:     "name": "snils",
63:     "params": [
64:         {
65:             "snils": "11111111333"
66:         }
67:     ]
68: },
69: {
70:     "name": "inn",
71:     "params": [
```

```

72:         {
73:             "inn": "111111111122"
74:         }
75:     ]
76: },
77: {
78:     "name": "driver_license",
79:     "params": [
80:         {
81:             "authority": "OVD",
82:             "date_of_expiry": "2060-01-17T00:00:00.0000000Z",
83:             "date_of_issue": "1999-01-01T00:00:00.0000000Z",
84:             "document_number": "222222",
85:             "document_series": "1111",
86:             "place_of_birth": "city",
87:             "place_of_issue": "city",
88:             "vehicle_category": "B, C "
89:         }
90:     ]
91: }
92: ]
93: }

```

### create\_link

#### (создание связи между элементами)

Запрос устанавливает взаимосвязь между заданными элементами и возвращает значение **true** в случае успешного выполнения. Запрос работает с типом `user`.

Структура запроса:

- `method`: `create_link`
- `jsonrpc`: 2.0
- `id`: <номер запроса>
- `params`:
  - `security`:
    - `token`: <полученный при логине>
  - `left_type`: <тип левого связываемого элемента> (например, `user`)
  - `left_id`: <id левого связываемого элемента>
  - `right_type`: <тип правого связываемого элемента> (например, `car`)
  - `right_id`: <id правого связываемого элемента>

#### Пример:

```

1: "jsonrpc": "2.0",
2:   "method": "create_link",

```

```
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "left_type": "user",
8:     "left_id": "bb26eef8-7b60-4cf2-9cd2-d6edeb736193",
9:     "right_type": "company",
10:    "right_id": "23245ffa-50f0-4655-83ae-fed45cb09401"
11:  },
12:  "id": 23
```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:    "id": 23,
3:    "result": true
```

### delete\_link

(удаление связи между элементами)

Запрос удаляет существующую связь между указанными элементами и возвращает значение **true** в случае успешного удаления. Запрос работает с типом **user**.

Структура запроса:

- method: delete\_link
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - left\_type: <тип левого связанного элемента> (например, user)
  - left\_id: <id левого связанного элемента>
  - right\_type: <тип правого связанного элемента> (например, car)
  - right\_id: <id правого связанного элемента>

#### Пример:

```
1:  "jsonrpc": "2.0",
2:    "method": "delete_link",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "left_type": "user",
8:      "left_id": "bb26eef8-7b60-4cf2-9cd2-d6edeb736193",
9:      "right_type": "company",
10:     "right_id": "23245ffa-50f0-4655-83ae-fed45cb09401"
11:   },
```

```
12: "id": 24
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",  
2: "id": 24,  
3: "result": true
```

## УПРАВЛЕНИЕ

### **get\_possible\_commands\_by\_element** (получение списка возможных команд)

Запрос возвращает перечень допустимых команд (наименование, номер и возможные параметры) применительно к указанному элементу с учётом его текущего состояния.

Структура запроса:

- method: get\_possible\_commands\_by\_element
- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элемента>
  - id: <идентификатор элемента>

#### Пример:

```
1: "jsonrpc": "2.0",  
2: "method": "get_possible_commands_by_element",  
3: "params": {  
4: "security": {  
5: "token": "ae067f28-bb67-46aa-847f-066ff634bb78"  
6: },  
7: "type": "orion_output",  
8: "id": "1238743b-c588-46df-9e5e-0cd47c93c0cd"  
9: },  
10: "id": 25
```

#### Ответ сервера:

```
1: "jsonrpc": "2.0",  
2: "id": 25,  
3: "result": {  
4: "commands": [  
5: {  
6: "name": "Выключение контроля",  
7: "command": 1104,  
8: "params": [  
9: {
```

```
10:         "optional": true,
11:         "name": "time",
12:         "description": "Время выключения",
13:         "type": "numeric",
14:         "internal_type": 0,
15:         "default_value": 60,
16:         "max_value": 16777215,
17:         "min_value": 1,
18:         "unit": "секунды"
19:     }
20: ]
21: },
22: {
23:     "name": "Включение теста индикации",
24:     "command": 8404,
25:     "params": []
26: },
27: {
28:     "name": "Выключение теста индикации",
29:     "command": 8413,
30:     "params": []
31: },
32: {
33:     "name": "Установить яркость",
34:     "command": 8418,
35:     "params": [
36:         {
37:             "optional": false,
38:             "name": "brightness",
39:             "description": "Яркость",
40:             "type": "enumerable",
41:             "internal_type": 1,
42:             "values": [
43:                 {
44:                     "value": 1,
45:                     "text": "10%"
46:                 },
47:                 {
48:                     "value": 2,
49:                     "text": "20%"
50:                 },
51:                 {
52:                     "value": 3,
53:                     "text": "30%"
```

```

54:         },
55:         {
56:             "value": 4,
57:             "text": "40%"
58:         },
59:         {
60:             "value": 5,
61:             "text": "50%"
62:         },
63:         {
64:             "value": 6,
65:             "text": "60%"
66:         },
67:         {
68:             "value": 7,
69:             "text": "70%"
70:         },
71:         {
72:             "value": 8,
73:             "text": "80%"
74:         },
75:         {
76:             "value": 9,
77:             "text": "90%"
78:         },
79:         {
80:             "value": 10,
81:             "text": "100%"
82:         }
83:     ]
84: }
85: ]
86: }
87: ]
88: }

```

### **send\_command\_query\_for\_element**

#### **(отправка запроса на выполнение команды)**

Запрос инициирует исполнение указанной команды для выбранного элемента. В случае успешной отправки запроса возвращает значение **true**. Это означает, что сервер получил запрос на выполнение команды, но не гарантирует её фактическое выполнение. Для получения результата выполнения команды требуется получение соответствующего сообщения по каналу WebSocket.

Структура запроса:

- method: send\_command\_query\_for\_element

- jsonrpc: 2.0
- id: <номер запроса>
- params:
  - security:
    - token: <полученный при логине>
  - type: <тип элемента>
  - id: <идентификатор элемента>
  - command: <номер команды>
  - params: { <название параметра>: <значение параметра> } (может быть пустой)
    - если у параметра есть поле «optional» и оно равно «true», то этот параметр в запросе является необязательным
    - <название параметра> берётся из поля «name»
    - тип параметра указан в поле «type»: numeric, enumerable или string

Пример отправки команды без параметра (здесь параметр не обязателен):

```

1:  "jsonrpc": "2.0",
2:    "method": "send_command_query_for_element",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "orion_output",
8:      "id": "1238743b-c588-46df-9e5e-0cd47c93c0cd",
9:      "command": 1104,
10:     "params": {}
11:   },
12:   "id": 26

```

Пример отправки команды с параметром типа «numeric»:

```

1:  "jsonrpc": "2.0",
2:    "method": "send_command_query_for_element",
3:    "params": {
4:      "security": {
5:        "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:      },
7:      "type": "orion_output",
8:      "id": "1238743b-c588-46df-9e5e-0cd47c93c0cd",
9:      "command": 1104,
10:     "params": {
11:       "time": 60
12:     }
13:   },
14:   "id": 26

```

Пример отправки команды с параметром типа «enumerable»:

```

1:  "jsonrpc": "2.0",

```

```
2:   "method": "send_command_query_for_element",
3:   "params": {
4:     "security": {
5:       "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:     },
7:     "type": "orion_output",
8:     "id": "1238743b-c588-46df-9e5e-0cd47c93c0cd",
9:     "command": 8418,
10:    "params": {
11:      "brightness": 2
12:    }
13:  },
14:  "id": 26
```

#### Пример отправки команды с параметром типа «string»:

```
1:  "jsonrpc": "2.0",
2:  "method": "send_command_query_for_element",
3:  "params": {
4:    "security": {
5:      "token": "ae067f28-bb67-46aa-847f-066ff634bb78"
6:    },
7:    "type": "s2000_k",
8:    "id": "8e56e5ce-a06c-4914-a217-81174bec05d4",
9:    "command": 380,
10:   "params": {
11:     "message": "some text"
12:   }
13: },
14: "id": 26
```

#### Ответ сервера:

```
1:  "jsonrpc": "2.0",
2:  "id": 26,
3:  "result": true
```

## Приложение А. Описание полей типов

### **cars**

- Описание: Автомобили
- Поля: отсутствуют

### **car**

- Описание: Автомобиль
- Поля:
  - **model**: Модель автомобиля
    - type: string
    - значение по умолчанию:
  - **color**: Цвет автомобиля
    - type: string
    - значение по умолчанию:
  - **vin**: VIN номер
    - type: string
    - значение по умолчанию:
  - **registration\_number**: Регистрационный номер
    - type: string
    - значение по умолчанию:
  - **comment**: Комментарии
    - type: string
    - значение по умолчанию:

### **company**

- Описание: Организация
- Поля:
  - **address**: Адрес
    - type: string
    - значение по умолчанию:
  - **phone**: Телефон
    - type: string
    - значение по умолчанию:
  - **comment**: Комментарии
    - type: string
    - значение по умолчанию:

### **department**

- Описание: Отдел
- Поля: отсутствуют

### **user**

- Описание: Пользователь
- Поля:
  - **user\_name**: Имя
    - type: string
    - значение по умолчанию:
  - **surname**: Фамилия
    - type: string
    - значение по умолчанию:
  - **patronymic**: Отчество
    - type: string
    - значение по умолчанию:
  - **street**: Улица
    - type: string
    - значение по умолчанию:
  - **home\_address**: Домашний адрес

- type: string
  - значение по умолчанию:
- office\_address: Офис
  - type: string
  - значение по умолчанию:
- tab\_number: Табельный номер
  - type: string
  - значение по умолчанию:
- work\_phone: Рабочий телефон
  - type: string
  - значение по умолчанию:
- status:
  - type: integer
  - значение по умолчанию:
- date\_birth: Дата рождения
  - type: datetime
  - значение по умолчанию:
- black\_list: Черный список
  - type: integer
  - значение по умолчанию:
- black\_list\_cause: Причина попадания в черный список
  - type: string
  - значение по умолчанию:
- fired: Уволен
  - type: integer
  - значение по умолчанию:
- fired\_cause: Причина увольнения
  - type: string
  - значение по умолчанию:
- free\_schedule: Свободный график
  - type: integer
  - значение по умолчанию:
- prohibited\_transition\_in\_a\_day: Запрет перехода через сутки
  - type: integer
  - значение по умолчанию:
- home\_phone: Домашний телефон
  - type: string
  - значение по умолчанию:
- email: Email
  - type: string
  - значение по умолчанию:
- sex: Пол
  - type: integer
  - значение по умолчанию: 1
- photo: Фотография
  - type: binary
  - значение по умолчанию:
- additional\_work\_phone: Добавочный номер
  - type: string
  - значение по умолчанию:

**Дополнительные таблицы (относятся только к user)**

- **keys**
  - key\_code:
    - type: string
    - значение по умолчанию:
  - type\_key:
    - type: integer
    - значение по умолчанию:
  - locked:
    - type: integer
    - значение по умолчанию:
- **passport**
  - passport\_series: Серия
    - type: string
    - значение по умолчанию:
  - passport\_number: Номер
    - type: string
    - значение по умолчанию:

- **authority**: Кем выдан
  - type: string
  - значение по умолчанию:
- **date\_of\_issue**: Дата выдачи
  - type: datetime
  - значение по умолчанию:
- **place\_of\_birth**: Место рождения
  - type: string
  - значение по умолчанию:
- **subdivision\_code**: Код подразделения
  - type: string
  - значение по умолчанию:
- **international passport**
  - **passport\_series**: Серия
    - type: string
    - значение по умолчанию:
  - **passport\_number**: Номер
    - type: string
    - значение по умолчанию:
  - **authority**: Кем выдан
    - type: string
    - значение по умолчанию:
  - **date\_of\_issue**: Дата выдачи
    - type: datetime
    - значение по умолчанию:
  - **date\_of\_expiry**: Действителен до
    - type: datetime
    - значение по умолчанию:
  - **place\_of\_birth**: Место рождения
    - type: string
    - значение по умолчанию:
- **military\_id**
  - **document\_series**: Серия
    - type: string
    - значение по умолчанию:
  - **document\_number**: Номер
    - type: string
    - значение по умолчанию:
  - **authority**: Кем выдан
    - type: string
    - значение по умолчанию:
  - **date\_of\_issue**: Дата выдачи
    - type: datetime
    - значение по умолчанию:
  - **military\_rank**: Воинское звание
    - type: string
    - значение по умолчанию:
  - **mos**: ВУС
    - type: string
    - значение по умолчанию:
  - **place\_of\_birth**: Место рождения
    - type: string
    - значение по умолчанию:
- **driver\_license**
  - **document\_series**: Серия
    - type: string
    - значение по умолчанию:
  - **document\_number**: Номер
    - type: string
    - значение по умолчанию:
  - **authority**: Кем выдан
    - type: string
    - значение по умолчанию:
  - **date\_of\_issue**: Дата выдачи
    - type: datetime
    - значение по умолчанию:
  - **date\_of\_expiry**: Действителен до
    - type: datetime
    - значение по умолчанию:
  - **place\_of\_issue**: Регион выдачи

- type: string
    - значение по умолчанию:
  - vehicle\_category: Категория транспорта
    - type: string
    - значение по умолчанию:
  - place\_of\_birth: Место рождения
    - type: string
    - значение по умолчанию:
- inn
  - inn: ИНН
    - type: string
    - значение по умолчанию:
- snils
  - snils: СНИЛС
    - type: string
    - значение по умолчанию:

#### departments

- Описание: Организации и Отделы
- Поля: отсутствуют

#### logic\_group\_zones

- Описание: Системная группа зон
- Поля:
  - number: номер
    - type: integer
    - значение по умолчанию:
  - comment: Комментарии
    - type: string
    - значение по умолчанию:

#### logic\_zone

- Описание: Системная зона
- Поля:
  - number: номер
    - type: integer
    - значение по умолчанию:
  - comment: Комментарии
    - type: string
    - значение по умолчанию:

#### s2000m

- Описание: С2000/С2000М
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:

#### signal\_20

- Описание: Сигнал-20
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID

- type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### signal\_20p

- Описание: Сигнал-20П
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_sp1

- Описание: С2000-СП1
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_4

- Описание: С2000-4
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:

- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### s2000\_4\_v300

- Описание: С2000-4 (вер. 3.00 и выше)
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_k

- Описание: С2000-К
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_it

- Описание: С2000-ИТ
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_kdl

- Описание: С2000-КДЛ
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_kdl\_v300

- Описание: С2000-КДЛ (вер. 3.00 и выше)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_bi

- Описание: С2000-БИ
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_bki\_v220

- Описание: С2000-БКИ (вер. 2.20 - 2.22)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID

- type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### s2000\_bi\_v223

- Описание: С2000-БИ (вер. 2.23 и выше)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### signal\_20\_02

- Описание: Сигнал-20 сер. 02
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_ks

- Описание: С2000-КС
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:

- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### s2000\_aspt

- Описание: С2000-АСПТ
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_aspt\_v200

- Описание: С2000-АСПТ (вер. 2.xx)
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_aspt\_v300

- Описание: С2000-АСПТ (вер. 3.00 и выше)
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_kpb

- Описание: С2000-КПБ
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_2

- Описание: С2000-2
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_2\_v220

- Описание: С2000-2 (вер. 2.20 - 2.74)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_2\_v275

- Описание: С2000-2 (вер. 2.75 и выше)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID

- type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
- значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

### rupor

- Описание: Рупор
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - voice\_message\_number: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

### rupor\_v200

- Описание: Рупор (вер. 2.00 и выше)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - voice\_message\_number: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

### rupor\_disp\_01

- Описание: Рупор-Диспетчер исп. 01
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:

- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### s2000\_pt

- Описание: С2000-ПТ
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### uo\_4s

- Описание: УО-4С
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### potok\_3n

- Описание: Поток-3Н
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer

- значение по умолчанию: 10

### potok\_3n\_v103

- Описание: Поток-3Н (вер. 1.03 и выше)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

### signal\_20m

- Описание: Сигнал-20М
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

### signal\_20m\_v200

- Описание: Сигнал-20М (вер. 2.00 и выше)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

### s2000\_bi\_01

- Описание: С2000-БИ исп. 01
- Поля:
  - version: Версия
    - type: integer

- значение по умолчанию:
- **contact\_id**: Contact ID
  - type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

### **rupor\_01**

- Описание: Рупор исп. 01
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - **voice\_message\_number**: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

### **rupor\_01\_v103**

- Описание: Рупор исп. 01 (вер. 1.03 и выше)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - **voice\_message\_number**: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

### **s2000\_adem**

- Описание: С2000-Adem
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID

- type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

### rip\_12\_50

- Описание: РИП-12 исп. 50/51
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

### signal\_10

- Описание: Сигнал-10
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

### s2000\_pp

- Описание: С2000-ПП
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:

- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### **rip\_12\_54**

- Описание: РИП-12 исп. 54
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **rip\_24\_50**

- Описание: РИП-24 исп. 50/51
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **s2000\_kdl\_s**

- Описание: С2000-КДЛ-С
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **s2000\_kdl\_2i**

- Описание: С2000-КДЛ-2И
- Поля:

- version: Версия
  - type: integer
  - значение по умолчанию:
- contact\_id: Contact ID
  - type: integer
  - значение по умолчанию:
- number: Номер
  - type: integer
  - значение по умолчанию:
- adc\_request: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- adc\_request\_period: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### s2000\_pge

- Описание: С2000-PGE
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_bki\_v223

- Описание: С2000-БКИ (вер. 2.23 и выше)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### potok\_bki

- Описание: Поток-БКИ
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer

- значение по умолчанию:
- `adc_request`: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### **rupor\_200**

- Описание: Рупор-200
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - `voice_message_number`: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

#### **s2000\_perimeter**

- Описание: С2000-Периметр
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **mip\_12**

- Описание: МИП-12
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП

- type: integer
- значение по умолчанию: 10

#### mip\_24

- Описание: МИП-24
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### mip\_24\_s

- Описание: МИП-24-С
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### rip\_48\_01

- Описание: РИП-48 исп. 01
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### rip\_12\_56

- Описание: РИП-12 исп. 56
- Поля:
  - version: Версия

- type: integer
  - значение по умолчанию:
- **contact\_id**: Contact ID
  - type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### rip\_24\_56

- Описание: РИП-24 исп. 56
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### sirius

- Описание: ППКУП «Сириус»
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **ip\_port**: IP-адрес + порт
    - type: string
    - значение по умолчанию:
  - **mac**: MAC-адрес
    - type: string
    - значение по умолчанию:
  - **time\_zone**: Часовой пояс
    - type: string
    - значение по умолчанию:
  - **interpanel\_mode**: Межпанельный режим
    - type: integer
    - значение по умолчанию: 5
  - **login**: Логин
    - type: string
    - значение по умолчанию:
  - **password**: Пароль
    - type: string
    - значение по умолчанию:

#### sirius\_v200

- Описание: ППКУП “Сириус” (вер. 2.00)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - ip\_port: IP-адрес + порт
    - type: string
    - значение по умолчанию:
  - mac: MAC-адрес
    - type: string
    - значение по умолчанию:
  - time\_zone: Часовой пояс
    - type: string
    - значение по умолчанию:
  - interpanel\_mode: Межпанельный режим
    - type: integer
    - значение по умолчанию: 5
  - login: Логин
    - type: string
    - значение по умолчанию:
  - password: Пароль
    - type: string
    - значение по умолчанию:

#### **rupor\_02**

- Описание: Рупор исп. 02
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - voice\_message\_number: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

#### **s2000\_kdl\_modbus**

- Описание: С2000-КДЛ-Modbus
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:

- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

### **rupor\_03**

- Описание: Рупор исп. 03
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - `voice_message_number`: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

### **rupor\_300**

- Описание: Рупор-300
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - `voice_message_number`: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

### **rupor\_300\_v103**

- Описание: Рупор-300 (вер. 1.03 - 1.04)
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer

- значение по умолчанию: 10
- voice\_message\_number: Номер голосового сообщения
  - type: integer
  - значение по умолчанию: 1

#### rupor\_300\_v105

- Описание: Рупор-300 (вер. 1.05 - 1.99)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - voice\_message\_number: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

#### rupor\_300\_v200

- Описание: Рупор-300 (вер. 2.00 - 2.04)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - voice\_message\_number: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

#### rupor\_300\_v205

- Описание: Рупор-300 (вер. 2.05 и выше)
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП

- type: integer
  - значение по умолчанию: 10
- voice\_message\_number: Номер голосового сообщения
  - type: integer
  - значение по умолчанию: 1

#### s2000\_kpb\_s

- Описание: С2000-КПБ-С
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_kpb\_s\_sirius\_v200

- Описание: С2000-КПБ-С
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### s2000\_pge\_01

- Описание: С2000-PGE исп. 01
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## rip\_24\_57

- Описание: РИП-24 исп. 57
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## pkv\_rip\_12\_56

- Описание: ПКВ-РИП-12 исп. 56
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## pkv\_rip\_24\_56

- Описание: ПКВ-РИП-24 исп. 56
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

## s2000\_kdl\_2i\_01

- Описание: С2000-КДЛ-2И исп. 01
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID

- type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### **s2000\_kdl\_2i\_01\_v300**

- Описание: С2000-КДЛ-2И исп. 01 (вер. 3.00 и выше)
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **shkp\_rs**

- Описание: ШКП-RS
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **microphone\_console\_20**

- Описание: Микрофонная консоль - 20
- Поля:
  - **version**: Версия
    - type: integer
    - значение по умолчанию:
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:

- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### **rupor\_disp\_02**

- Описание: Рупор-Диспетчер исп. 02
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **mip\_12\_11**

- Описание: МИП-12 исп. 11
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **mip\_24\_11**

- Описание: МИП-24 исп. 11
- Поля:
  - `version`: Версия
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `adc_request`: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - `adc_request_period`: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10

#### **s2000\_rpi\_02**

- Описание: С2000-РПИ исп. 02
- Поля:

- version: Версия
  - type: integer
  - значение по умолчанию:
- contact\_id: Contact ID
  - type: integer
  - значение по умолчанию:
- number: Номер
  - type: integer
  - значение по умолчанию:
- adc\_request: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- adc\_request\_period: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### rupor\_75

- Описание: Рупор-75
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - voice\_message\_number: Номер голосового сообщения
    - type: integer
    - значение по умолчанию: 1

#### unknown\_device

- Описание: Неопознанный прибор
- Поля:
  - version: Версия
    - type: integer
    - значение по умолчанию:
  - contact\_id: Contact ID
    - type: integer
    - значение по умолчанию:
  - number: Номер
    - type: integer
    - значение по умолчанию:
  - adc\_request: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - adc\_request\_period: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - type: Тип прибора по протоколу Орион
    - type: integer
    - значение по умолчанию:

#### user\_device

- Описание: Пользовательский прибор
- Поля:
  - version: Версия
    - type: integer

- значение по умолчанию:
- **contact\_id**: Contact ID
  - type: integer
  - значение по умолчанию:
- **number**: Номер
  - type: integer
  - значение по умолчанию:
- **base\_type**: Базовый тип
  - type: string
  - значение по умолчанию:
- **adc\_request**: Запрос АЦП
  - type: integer
  - значение по умолчанию:
- **adc\_request\_period**: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10

#### orion\_input

- Описание: Вход
- Поля:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **type\_id**: Индекс типа
    - type: integer
    - значение по умолчанию: -10
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП
    - type: integer
    - значение по умолчанию:
  - **adc\_request\_period**: Периодичность запроса АЦП
    - type: integer
    - значение по умолчанию: 10
  - **subtype\_id**: Идентификатор подтипа
    - type: integer
    - значение по умолчанию: -1

#### orion\_reader

- Описание: Считыватель
- Поля:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **type\_id**: Индекс типа
    - type: integer
    - значение по умолчанию: -10
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:

#### orion\_output

- Описание: Выход
- Поля:
  - **number**: Номер
    - type: integer
    - значение по умолчанию:
  - **type\_id**: Индекс типа
    - type: integer
    - значение по умолчанию: -10
  - **contact\_id**: Contact ID
    - type: integer
    - значение по умолчанию:
  - **adc\_request**: Запрос АЦП

- type: integer
  - значение по умолчанию:
- `adc_request_period`: Периодичность запроса АЦП
  - type: integer
  - значение по умолчанию: 10
- `control_delay_1`: Control delay 1
  - type: double
  - значение по умолчанию:
- `control_delay_2`: Control delay 2
  - type: double
  - значение по умолчанию:
- `control_active_time`: Control active time
  - type: double
  - значение по умолчанию:
- `infinite_control_active_time`: Infinite control active time
  - type: integer
  - значение по умолчанию:
- `control_program_number`: Control program number
  - type: integer
  - значение по умолчанию:

### **orion\_destination**

- Описание: Адресат
- Поля:
  - `number`: Номер
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:

### **orion\_zone**

- Описание: Аппаратная зона
- Поля:
  - `number`: Номер раздела
    - type: integer
    - значение по умолчанию:
  - `comment`: Комментарий
    - type: string
    - значение по умолчанию:
  - `device_number`: Номер в приборе
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:
  - `pt`: Зона пожаротушения
    - type: integer
    - значение по умолчанию:

### **orion\_group\_zone**

- Описание: Аппаратная группа зон
- Поля:
  - `number`: Номер группы
    - type: integer
    - значение по умолчанию:
  - `comment`: Комментарий
    - type: string
    - значение по умолчанию:
  - `device_number`: Номер в приборе
    - type: integer
    - значение по умолчанию:
  - `contact_id`: Contact ID
    - type: integer
    - значение по умолчанию:

## **position**

- Описание: Должность
- Поля:
  - comment:
    - type: string
    - значение по умолчанию:

## **positions**

- Описание: Должности
- Поля: отсутствуют

## Приложение Б. Связи между типами

### Car

- Описание: Автомобиль
- Связи слева: Отсутствуют
- Связи справа:
  - cars
  - user
- Возможность удаления: Да
- Возможность создания: Только под корневым объектом
- Родительский (корневой) тип: cars

### Cars

- Описание: Автомобили
- Связи слева:
  - car
- Связи справа: Отсутствуют
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Company

- Описание: Организация
- Связи слева:
  - department
- Связи справа:
  - departments
- Возможность удаления: Да
- Возможность создания: Только под корневым объектом
- Родительский (корневой) тип: departments

### Department

- Описание: Отдел
- Связи слева:
  - department
  - user
- Связи справа:
  - company
  - department
- Возможность удаления: Да
- Возможность создания: Только под родителем
- Родительский (корневой) тип:

### Departments

- Описание: Организации и Отделы
- Связи слева:
  - company
- Связи справа: Отсутствуют
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Logic\_group\_zones

- Описание: Системная группа зон
- Связи слева:

- logic\_zone
- orion\_zone
- Связи справа: Отсутствуют
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Logic\_zone

- Описание: Системная зона
- Связи слева:
  - microphone\_console\_20
  - mip\_12
  - mip\_12\_11
  - mip\_24
  - mip\_24\_11
  - mip\_24\_s
  - orion\_destination
  - orion\_input
  - orion\_output
  - orion\_reader
  - pkv\_rip\_12\_56
  - pkv\_rip\_24\_56
  - potok\_3n
  - potok\_3n\_v103
  - potok\_bki
  - rip\_12\_50
  - rip\_12\_54
  - rip\_12\_56
  - rip\_24\_50
  - rip\_24\_56
  - rip\_24\_57
  - rip\_48\_01
  - rupor
  - rupor\_01
  - rupor\_01\_v103
  - rupor\_02
  - rupor\_03
  - rupor\_200
  - rupor\_300
  - rupor\_300\_v103
  - rupor\_300\_v105
  - rupor\_300\_v200
  - rupor\_300\_v205
  - rupor\_75
  - rupor\_disp\_01
  - rupor\_disp\_02
  - rupor\_v200
  - s2000\_2
  - s2000\_2\_v220
  - S2000\_2\_v275
  - s2000\_4
  - s2000\_4\_v300
  - s2000\_adem
  - s2000\_aspt
  - s2000\_aspt\_v200
  - s2000\_aspt\_v300
  - s2000\_bi
  - s2000\_bi\_01
  - s2000\_bi\_v223
  - s2000\_bki\_v220
  - s2000\_bki\_v223
  - s2000\_it
  - s2000\_k
  - s2000\_kdl
  - s2000\_kdl\_2i
  - s2000\_kdl\_2i\_01

- s2000\_kdl\_2i\_01\_v300
- s2000\_kdl\_v300
- s2000\_kdl\_modbus
- s2000\_kdl\_s
- s2000\_kpb
- s2000\_kpb\_s
- s2000\_kpb\_s\_sirius\_v200
- s2000\_ks
- s2000\_perimeter
- s2000\_pge
- s2000\_pge\_01
- s2000\_pp
- s2000\_pt
- s2000\_rpi\_02
- s2000\_sp1
- s2000m
- shkp\_rs
- signal\_10
- signal\_20
- signal\_20\_02
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- sirius
- sirius\_v200
- unknown\_device
- uo\_4s
- user\_device
- Связи справа:
  - logic\_group\_zones
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Microphone\_console\_20

- Описание: Микрофонная консоль - 20
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Mip\_12

- Описание: МИП-12
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Mip\_12\_11

- Описание: МИП-12 исп. 11
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Mip\_24**

- Описание: МИП-24
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Mip\_24\_11**

- Описание: МИП-24 исп. 11
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Mip\_24\_s**

- Описание: МИП-24-С
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Orion\_destination**

- Описание: Адресат
- Связи слева: Отсутствуют
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000\_it

- s2000\_kpb\_s
- s2000\_pge
- s2000\_pge\_01
- s2000\_pp
- sirius\_v200
- unknown\_device
- uo\_4s
- user\_device
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Orion\_group\_zone

- Описание: Аппаратная группа зон
- Связи слева:
  - orion\_zone
- Связи справа:
  - orion\_output
  - orion\_reader
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Orion\_input

- Описание: Вход
- Связи слева: Отсутствуют
- Связи справа:
  - logic\_zone
  - microphone\_console\_20
  - mip\_12
  - mip\_12\_11
  - mip\_24
  - mip\_24\_11
  - mip\_24\_s
  - orion\_zone
  - pkv\_rip\_12\_56
  - pkv\_rip\_24\_56
  - potok\_3n
  - potok\_3n\_v103
  - rip\_12\_50
  - rip\_12\_54
  - rip\_12\_56
  - rip\_24\_50
  - rip\_24\_56
  - rip\_24\_57
  - rip\_48\_01
  - rurog
  - rurog\_01\_v103
  - rurog\_02
  - rurog\_03
  - rurog\_200
  - rurog\_300
  - rurog\_300\_v103
  - rurog\_300\_v105
  - rurog\_300\_v200
  - rurog\_300\_v205
  - rurog\_75
  - rurog\_disp\_01
  - rurog\_disp\_02
  - rurog\_v200
  - s2000\_2
  - s2000\_2\_v220

- S2000\_2\_v275
- s2000\_4
- s2000\_4\_v300
- s2000\_adem
- s2000\_aspt
- s2000\_aspt\_v200
- s2000\_aspt\_v300
- s2000\_kdl
- s2000\_kdl\_2i
- s2000\_kdl\_2i\_01
- s2000\_kdl\_2i\_01\_v300
- s2000\_kdl\_300
- s2000\_kdl\_modbus
- s2000\_kdl\_s
- s2000\_kpb
- s2000\_kpb\_s
- s2000\_kpb\_s\_sirius\_v200
- s2000\_perimeter
- shkp\_rs
- signal\_10
- signal\_20
- signal\_20\_02
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- unknown\_device
- uo\_4s
- user\_device
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Orion\_output

- Описание: Выход
- Связи слева:
  - orion\_group\_zone
  - orion\_zone
- Связи справа:
  - logic\_zone
  - orion\_zone
  - potok\_3n
  - potok\_3n\_v103
  - rupor
  - rupor\_01
  - rupor\_01\_v103
  - rupor\_02
  - rupor\_03
  - rupor\_200
  - rupor\_300
  - rupor\_300\_v103
  - rupor\_300\_v105
  - rupor\_300\_v200
  - rupor\_300\_v205
  - rupor\_75
  - rupor\_v200
  - s2000\_2
  - s2000\_2\_v220
  - S2000\_2\_v275
  - s2000\_4
  - s2000\_4\_v300
  - s2000\_aspt\_v200
  - s2000\_aspt\_v300
  - s2000\_kdl
  - s2000\_kdl\_2i
  - s2000\_kdl\_2i\_01
  - s2000\_kdl\_2i\_01\_v300

- s2000\_kdl\_v300
- s2000\_kdl\_modbus
- s2000\_kdl\_s
- s2000\_kpb
- s2000\_kpb\_s
- s2000\_kpb\_s\_sirius\_v200
- s2000\_perimeter
- s2000\_sp1
- shkp\_rs
- signal\_10
- signal\_20\_02
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- unknown\_device
- uo\_4s
- user\_device
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Orion\_reader

- Описание: Считыватель
- Связи слева:
  - orion\_group\_zone
  - orion\_zone
- Связи справа:
  - logic\_zone
  - orion\_zone
  - potok\_bki
  - s2000\_2
  - s2000\_2\_v220
  - S2000\_2\_v275
  - s2000\_4
  - s2000\_4\_v300
  - s2000\_bi\_v223
  - s2000\_bki\_v220
  - s2000\_bki\_v223
  - s2000\_k
  - s2000\_kdl
  - s2000\_kdl\_2i
  - s2000\_kdl\_2i\_01
  - s2000\_kdl\_modbus
  - s2000\_kpb\_s
  - s2000\_ks
  - s2000\_pge
  - s2000\_pge\_01
  - s2000\_pt
  - s2000m
  - signal\_10
  - signal\_20m
  - signal\_20m\_v200
  - signal\_20p
  - sirius\_v200
  - unknown\_device
  - uo\_4s
  - user\_device
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Orion\_zone

- Описание: Аппаратная зона
- Связи слева:
  - microphone\_console\_20

- mip\_12
- mip\_12\_11
- mip\_24
- mip\_24\_11
- mip\_24\_s
- orion\_destination
- orion\_input
- orion\_output
- orion\_reader
- pkv\_rip\_12\_56
- pkv\_rip\_24\_56
- potok\_3n
- potok\_3n\_v103
- potok\_bki
- rip\_12\_50
- rip\_12\_54
- rip\_12\_56
- rip\_24\_50
- rip\_24\_56
- rip\_24\_57
- rip\_48\_01
- rupor
- rupor\_01
- rupor\_01\_v103
- rupor\_02
- rupor\_03
- rupor\_200
- rupor\_300
- rupor\_300\_v103
- rupor\_300\_v105
- rupor\_300\_v200
- rupor\_300\_v205
- rupor\_75
- rupor\_disp\_01
- rupor\_disp\_02
- rupor\_v200
- s2000\_2
- s2000\_2\_v220
- S2000\_2\_v275
- s2000\_4
- s2000\_4\_v300
- s2000\_adem
- s2000\_aspt
- s2000\_aspt\_v200
- s2000\_aspt\_v300
- s2000\_bi
- s2000\_bi\_01
- s2000\_bi\_v223
- s2000\_bki\_v220
- s2000\_bki\_v223
- s2000\_it
- s2000\_k
- s2000\_kdl
- s2000\_kdl\_2i
- s2000\_kdl\_2i\_01
- s2000\_kdl\_2i\_01\_v300
- s2000\_kdl\_v300
- s2000\_kdl\_modbus
- s2000\_kdl\_s
- s2000\_kpb
- s2000\_kpb\_s
- s2000\_kpb\_s\_sirius\_v200
- s2000\_ks
- s2000\_perimeter
- s2000\_pge
- s2000\_pge\_01
- s2000\_pp
- s2000\_pt

- s2000\_rpi\_02
- s2000\_sp1
- s2000m
- shkp\_rs
- signal\_10
- signal\_20
- signal\_20\_02
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- sirius
- sirius\_v200
- unknown\_device
- uo\_4s
- user\_device
- Связи справа:
  - logic\_group\_zones
  - orion\_group\_zone
  - orion\_output
  - orion\_reader
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Pkv\_rip\_12\_56**

- Описание: ПКВ-РИП-12 исп. 56
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Pkv\_rip\_24\_56**

- Описание: ПКВ-РИП-24 исп. 56
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **Position**

- Описание: Должность
- Связи слева: Отсутствуют
- Связи справа:
  - positions
  - user
- Возможность удаления: Да
- Возможность создания: Только под корневым объектом

- Родительский (корневой) тип: positions

### Positions

- Описание: Должности
- Связи слева:
  - position
- Связи справа: Отсутствуют
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Potok\_3n

- Описание: Поток-3Н
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Potok\_3n\_v103

- Описание: Поток-3Н (вер. 1.03 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Potok\_bki

- Описание: Поток-БКИ
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Rip\_12\_50

- Описание: РИП-12 исп. 50/51
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone

- s2000m
- sirius
- sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rip\_12\_54

- Описание: РИП-12 исп. 54
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rip\_12\_56

- Описание: РИП-12 исп. 56
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rip\_24\_50

- Описание: РИП-24 исп. 50/51
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rip\_24\_56

- Описание: РИП-24 исп. 56
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет

- Родительский (корневой) тип:

#### Rip\_24\_57

- Описание: РИП-24 исп. 57
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rip\_48\_01

- Описание: РИП-48 исп. 01
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rupor

- Описание: Рупор
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Rupor\_01

- Описание: Рупор исп. 01
- Связи слева:
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Rupor\_01\_v103

- Описание: Рупор исп. 01 (вер. 1.03 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Rupor\_02

- Описание: Рупор исп. 02
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Rupor\_03

- Описание: Рупор исп. 03
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Rupor\_200

- Описание: Рупор-200
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_300

- Описание: Рупор-300
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_300\_v103

- Описание: Рупор-300 (вер. 1.03 - 1.04)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_300\_v105

- Описание: Рупор-300 (вер. 1.05 - 1.99)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_300\_v200

- Описание: Рупор-300 (вер. 2.00 - 2.04)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_300\_v205

- Описание: Рупор-300 (вер. 2.05 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_75

- Описание: Рупор-75
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_disp\_01

- Описание: Рупор-Диспетчер исп. 01
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_disp\_02

- Описание: Рупор-Диспетчер исп. 02
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Rupor\_v200

- Описание: Рупор (вер. 2.00 и выше)
- Связи слева:

- orion\_input
- orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_2

- Описание: C2000-2
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_2\_v220

- Описание: C2000-2 (вер. 2.20 – 2.74)
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_2\_v275

- Описание: C2000-2 (вер. 2.75 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_4

- Описание: C2000-4
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:

- logic\_zone
- orion\_zone
- s2000m
- sirius
- sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_4\_v300

- Описание: С2000-4 (вер. 3.00 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_adem

- Описание: С2000-Adem
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_aspt

- Описание: С2000-АСПТ
- Связи слева:
  - orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_aspt\_v200

- Описание: С2000-АСПТ (вер. 2.xx)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius

- sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_aspt\_v300

- Описание: С2000-АСПТ (вер. 3.00 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_bi

- Описание: С2000-БИ
- Связи слева: Отсутствуют
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_bi\_01

- Описание: С2000-БИ исп. 01
- Связи слева: Отсутствуют
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_bi\_v223

- Описание: С2000-БИ (вер. 2.23 и выше)
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_bki\_v220

- Описание: С2000-БКИ (вер. 2.20 - 2.22)
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_bki\_v223

- Описание: С2000-БКИ (вер. 2.23 и выше)
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_it

- Описание: С2000-ИТ
- Связи слева:
  - orion\_destination
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_k

- Описание: С2000-К
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_kdl

- Описание: С2000-КДЛ
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader

- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_kdl\_2i

- Описание: С2000-КДЛ-2И
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_kdl\_2i\_01

- Описание: С2000-КДЛ-2И исп. 01
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_kdl\_2i\_01\_v300

- Описание: С2000-КДЛ-2И исп. 01 (вер. 3.00 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_kdl\_v300

- Описание: С2000-КДЛ (вер. 3.00 и выше)
- Связи слева:

- orion\_input
- orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **S2000\_kdl\_modbus**

- Описание: С2000-КДЛ-Modbus
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **S2000\_kdl\_s**

- Описание: С2000-КДЛ-С
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **S2000\_kpb**

- Описание: С2000-КПБ
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### **S2000\_kpb\_s**

- Описание: С2000-КПБ-С
- Связи слева:
  - orion\_destination
  - orion\_input
  - orion\_output

- orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - sirius
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_kpb\_s\_sirius\_v200

- Описание: С2000-КПБ-С
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_ks

- Описание: С2000-КС
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_perimeter

- Описание: С2000-Периметр
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### S2000\_pge

- Описание: С2000-PGE
- Связи слева:
  - orion\_destination
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_pge\_01

- Описание: С2000-PGE исп. 01
- Связи слева:
  - orion\_destination
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_pp

- Описание: С2000-ПП
- Связи слева:
  - orion\_destination
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_pt

- Описание: С2000-ПТ
- Связи слева:
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_rpi\_02

- Описание: С2000-РПИ исп. 02
- Связи слева: Отсутствуют
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000\_sp1

- Описание: С2000-СП1
- Связи слева:
  - orion\_output
- Связи справа:

- logic\_zone
- orion\_zone
- s2000m
- sirius
- sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## S2000m

- Описание: C2000/C2000M
- Связи слева:
  - microphone\_console\_20
  - mip\_12
  - mip\_12\_11
  - mip\_24
  - mip\_24\_11
  - orion\_group\_zone
  - orion\_reader
  - orion\_zone
  - pkv\_rip\_12\_56
  - pkv\_rip\_24\_56
  - potok\_3n
  - potok\_3n\_v103
  - potok\_bki
  - rip\_12\_50
  - rip\_12\_54
  - rip\_12\_56
  - rip\_24\_50
  - rip\_24\_56
  - rip\_24\_57
  - rip\_48\_01
  - rupor
  - rupor\_01
  - rupor\_01\_v103
  - rupor\_02
  - rupor\_03
  - rupor\_200
  - rupor\_300
  - rupor\_300\_v103
  - rupor\_300\_v105
  - rupor\_300\_v200
  - rupor\_300\_v205
  - rupor\_75
  - rupor\_disp\_01
  - rupor\_disp\_02
  - rupor\_v200
  - s2000\_2
  - s2000\_2\_v220
  - s2000\_2\_v275
  - s2000\_4
  - s2000\_4\_v300
  - s2000\_adem
  - s2000\_aspt
  - s2000\_aspt\_v200
  - s2000\_aspt\_v300
  - s2000\_bi
  - s2000\_bi\_01
  - s2000\_bi\_v223
  - s2000\_bki\_v220
  - s2000\_bki\_v223
  - s2000\_it
  - s2000\_k
  - s2000\_kdl
  - s2000\_kdl\_2i
  - s2000\_kdl\_2i\_01

- s2000\_kdl\_2i\_01\_v300
- s2000\_kdl\_modbus
- s2000\_kdl\_v300
- s2000\_kpb
- s2000\_ks
- s2000\_perimeter
- s2000\_pge
- s2000\_pge\_01
- s2000\_pp
- s2000\_pt
- s2000\_rpi\_02
- s2000\_sp1
- shkp\_rs
- signal\_10
- signal\_20
- signal\_20\_02
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- unknown\_device
- uo\_4s
- user\_device
- Связи справа:
  - logic\_zone
  - orion\_zone
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Shkp\_rs

- Описание: ШКП-RS
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Signal\_10

- Описание: Сигнал-10
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Signal\_20

- Описание: Сигнал-20
- Связи слева:

- orion\_input
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Signal\_20\_02

- Описание: Сигнал-20 сер. 02
- Связи слева:
  - orion\_input
  - orion\_output
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Signal\_20m

- Описание: Сигнал-20М
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Signal\_20m\_v200

- Описание: Сигнал-20М (вер. 2.00 и выше)
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Signal\_20p

- Описание: Сигнал-20П
- Связи слева:
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:

- logic\_zone
- orion\_zone
- s2000m
- sirius
- sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

## Sirius

- Описание: ППКУП "Сириус"
- Связи слева:
  - microphone\_console\_20
  - mip\_12
  - mip\_12\_11
  - mip\_24
  - mip\_24\_11
  - mip\_24\_s
  - orion\_group\_zone
  - orion\_zone
  - pkv\_rip\_12\_56
  - pkv\_rip\_24\_56
  - potok\_3n\_v103
  - potok\_bki
  - rip\_12\_50
  - rip\_12\_54
  - rip\_12\_56
  - rip\_24\_50
  - rip\_24\_56
  - rip\_24\_57
  - rip\_48\_01
  - rupor
  - rupor\_01
  - rupor\_01\_v103
  - rupor\_02
  - rupor\_03
  - rupor\_200
  - rupor\_300
  - rupor\_300\_v103
  - rupor\_300\_v105
  - rupor\_300\_v200
  - rupor\_300\_v205
  - rupor\_disp\_01
  - rupor\_disp\_02
  - rupor\_v200
  - s2000\_4
  - s2000\_4\_v300
  - s2000\_aspt
  - s2000\_aspt\_v200
  - s2000\_aspt\_v300
  - s2000\_bi
  - s2000\_bi\_01
  - s2000\_bi\_v223
  - s2000\_bki\_v220
  - s2000\_bki\_v223
  - s2000\_k
  - s2000\_kdl
  - s2000\_kdl\_2i
  - s2000\_kdl\_2i\_01
  - s2000\_kdl\_2i\_01\_v300
  - s2000\_kdl\_s
  - s2000\_kdl\_v300
  - s2000\_kpb
  - s2000\_kpb\_s
  - s2000\_pge
  - s2000\_pge\_01

- s2000\_pp
- s2000\_pt
- s2000\_sp1
- shkp\_rs
- signal\_10
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- unknown\_device
- user\_device
- Связи справа:
  - logic\_zone
  - orion\_zone
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

### Sirius\_v200

- Описание: ППКУП "Сириус" (вер. 2.00)
- Связи слева:
  - microphone\_console\_20
  - mip\_12
  - mip\_12\_11
  - mip\_24
  - mip\_24\_11
  - mip\_24\_s
  - orion\_destination
  - orion\_group\_zone
  - orion\_reader
  - orion\_zone
  - pkv\_rip\_12\_56
  - pkv\_rip\_24\_56
  - potok\_3n\_v103
  - potok\_bki
  - rip\_12\_50
  - rip\_12\_54
  - rip\_12\_56
  - rip\_24\_50
  - rip\_24\_56
  - rip\_24\_57
  - rip\_48\_01
  - rupor
  - rupor\_01
  - rupor\_01\_v103
  - rupor\_02
  - rupor\_03
  - rupor\_200
  - rupor\_300
  - rupor\_300\_v103
  - rupor\_300\_v105
  - rupor\_300\_v200
  - rupor\_300\_v205
  - rupor\_75
  - rupor\_disp\_01
  - rupor\_disp\_02
  - rupor\_v200
  - s2000\_4
  - s2000\_4\_v300
  - s2000\_aspt
  - s2000\_aspt\_v200
  - s2000\_aspt\_v300
  - s2000\_bi
  - s2000\_bi\_01
  - s2000\_bi\_v223
  - s2000\_bki\_v220
  - s2000\_bki\_v223

- s2000\_k
- s2000\_kdl
- s2000\_kdl\_2i
- s2000\_kdl\_2i\_01
- s2000\_kdl\_2i\_01\_v300
- s2000\_kdl\_s
- s2000\_kdl\_v300
- s2000\_kpb
- s2000\_kpb\_s\_sirius\_v200
- s2000\_pge
- s2000\_pge\_01
- s2000\_pp
- s2000\_pt
- s2000\_rpi\_02
- s2000\_sp1
- shkp\_rs
- signal\_10
- signal\_20m
- signal\_20m\_v200
- signal\_20p
- unknown\_device
- user\_device
- Связи справа:
  - logic\_zone
  - orion\_zone
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Unknown\_device

- Описание: Неопознанный прибор
- Связи слева:
  - orion\_destination
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### Uo\_4s

- Описание: УО-4С
- Связи слева:
  - orion\_destination
  - orion\_input
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип:

#### User

- Описание: Пользователь
- Связи слева:

- car
- position
- Связи справа:
  - department
- Возможность удаления: Да
- Возможность создания: Только под родителем
- Родительский (корневой) тип:

#### User\_device

- Описание: Пользовательский прибор
- Связи слева:
  - orion\_destination
  - orion\_input
  - orion\_output
  - orion\_reader
- Связи справа:
  - logic\_zone
  - orion\_zone
  - s2000m
  - sirius
  - sirius\_v200
- Возможность удаления: Нет
- Возможность создания: Нет
- Родительский (корневой) тип: