Модуль интеграции Орион Про

Версия 2.0

Руководство по эксплуатации

АО НВП «Болид» 2025 год

Оглавление

Модуль интеграции Орион Про....6
Установка и настройка....6
Интерфейс SOAP....8
Интерфейс JSON-RPC 2.0....8
Описание типов данных и методов SOAP и JSON-RPC интерфейсов....9
TServiceError....9
TOperationResult<T>....9
TServiceInfo....9
TItemTypeInfo....10
TEventType....10
TEvent....11
TAccessFlags....12
TEventsGroup....13
TTimeWindow....13
TTimeInterval....14
TAccessLevelItem....15
TAccessLevel....16
TAccessZone....17
TEntryPoint....17
TPersonData....18
TKeyData....20
TCompany....21
TDepartment....21
TPosition....22
TExitGroup....22
TComputer....22
TComPort....23
TDeviceItem....23
TDevice....24
TSection....24
TSectionsGroup....25
TItem....25
TReaderItem....26
TItemADCValue....27
TScript....27
TVisitData....27
TCarData....28
TPersonCustomField....29
TAbsenceReasonType....29
TAbsenceReason....29
TExternalEvent....30
TDbChangeLogItem....31
GetReplService(...)....31
GetServiceInfo(...)....32
GetAdditionalTypesForItemType(...)....33
GetEventTypes(...)....33
GetEventTypesForItemType(...)....34
GetEventsGroups(...)....36
GetEventsGroupById(...)....36
GetEventsGroupsByEventTypeId(...)....37
GetEvents(...)....38
GetEventsCount(...)....41
GetEventsFromId(...)....42
SubscribeEvents(...)....43
CheckEventSubscription(...)....44
UnsubscribeEvents(...)....45
GetAccessLevels(...)....46
GetAccessLevelsCount(...)....46

GetAccessLevelById(...) 47
GetAccessZones(...) 48
GetEntryPoints(...) 49
GetEntryPointsCount(...) 49
GetEntryPointById(...) 50
PutPassWithAccLevels(...) 51
PutPassWithEntryPoints(...) 53
BlockPass(...) 55
BlockPerson(...) 56
BlockPersonByPass(...) 57
DeletePass(...) 58
GetPersons(...) 59
GetPersonsCount(...) 61
SearchPersons(...) 62
AddPerson(...) 63
UpdatePerson(...) 65
GetPersonPassList(...) 68
GetPersonPassListByPass(...) 68
GetPersonById(...) 69
GetPersonByPass(...) 71
GetPersonByExternalId(...) 73
GetPersonByAdditionalExternalId(...) 74
GetPersonByTabNumber(...) 76
GetPersonsWithEmptyPassList(...) 78
GetPersonsForEntryPoint(...) 79
DeletePerson(...) 81
DeletePersonByPass(...) 81
StopPerson(...) 82
StopPersonByPass(...) 83
GetVisits(…) 84
GetVisitById(…) 85
GetVisitByImpersonalPersonId(…) 85
GetVisitByPersonId(…) 86
AddVisit(…) 87
DeleteVisit(…) 87
UpdateVisit(…) 88
GetCars(…) 89
GetCarById(…) 90
AddCar(…) 91
DeleteCar(…) 92
UpdateCar(…) 93
AddPersonCar(…) 94
DeletePersonCar(…) 95
GetPersonCars(…) 96
GetCarPersons(…) 97
GetCompanies(...) 97
SearchCompany(...) 98
GetCompanyById(...) 99
CreateCompany(...) 100
UpdateCompany(...) 101
DeleteCompany(...) 101
GetDepartments(...) 102
SearchDepartment(...) 103
GetDepartmentById(...) 104
CreateDepartment(...) 104
UpdateDepartment(...) 105
DeleteDepartment(...) 106
GetCompanyDepartments(...) 107
GetPositions(...) 108
SearchPosition(...) 109
GetPositionById(...) 109

CreatePosition(...) 110
UpdatePosition(...) 111
DeletePosition(...) 112
GetExitGroups(...) 113
SearchExitGroup(...) 113
GetExitGroupById(...) 114
CreateExitGroup(...) 115
UpdateExitGroup(...) 116
DeleteExitGroup(...) 117
CreateTimeWindow(...) 118
GetTimeWindowById(...) 119
DeleteTimeWindow(...) 120
EditTimeWindow(...) 121
GetTimeWindows(...) 122
AddTimeInterval(...) 124
DeleteTimeInterval(...) 125
EditTimeInterval(...) 126
CreateAccessLevel(...) 127
EditAccessLevel(...) 128
DeleteAccessLevel(...) 129
AddAccessLevelItem(...) 130
EditAccessLevelItem(...) 131
DeleteAccessLevelItem(...) 132
GetKeyData(...) 133
GetKeyDataById(...) 134
UpdateKeyData(...) 135
AddKey(...) 136
GetKeys(...) 137
GetKeysCount(...) 139
GetComputers(...) 140
GetComPorts(...) 140
GetDevices(...) 141
GetDeviceById(...) 142
GetDevicesItems(...) 143
GetDeviceItems(...) 144
GetDeviceItemById(...) 144
GetSections(...) 145
GetSectionsCount(…) 146
GetSectionById(...) 147
GetSectionsGroups(...) 148
GetSectionsGroupsCount(…) 149
GetSectionsGroupById(...) 150
GetSectionItems(...) 150
GetSectionItemsCount(…) 152
GetSectionsGroupItems(...) 152
GetSectionsGroupItemsCount(…) 153
GetItems(…) 154
GetItemsByTimeOfChange(…) 155
GetItemsCount(…) 157
GetItemById(...) 157
GetReaderItems(...) 158
GetReaderItemsCount(...) 159
AddReaderItem(...) 160
DeleteReaderItem(...) 161
GetLoginToken(…) 162
ExtendTokenExpiration(…) 163
GetItemsForLogin(…) 163
GetItemsStates(…) 164
ControlItems(…) 166
GetADCValues(...) 168
GetScripts(...) 169

SendPasswordToEmail(…)....169
GetRecognitionChannels(...)....170
GetRecognitionChannelsCount(...)....170
GetRecognitionChannelById(...)....171
GetPersonCustomFields(...)....172
SetPersonCustomFieldValue(...)....172
GetAbsenceReasonTypes(…)....173
GetAbsenceReasonTypeById(…)....173
AddAbsenceReasonType(…)....174
DeleteAbsenceReasonType(…)....175
GetAbsenceReasons(…)....175
AddAbsenceReason(…)....176
DeleteAbsenceReason(…)....177
AddExternalEvent(…)....177
GetBioPrintFromReader(…)....178
CancelBioPrintReading(…)....179
ConvertWiegandToTouchMemory(...)....179
ConvertPinToTouchMemory(...)....180
ConvertImageToBioTemplate(...)....181
GetPinTouchMemory(...)....181
GetChangedPersons(...)....182
GetChangedEntryPoints(...)....183
GetChangedKeys(...)....183
GetChangedAccessLevels(...)....184
GetChangedAccessLevelItems(...)....184
GetChangedTimeWindows(...)....185
GetChangedTimeIntervals(...)....185
GetRooms(...)....186
SearchRoom(...)....186
GetRoomById(...)....187
CreateRoom(...)....187
UpdateRoom(...)....188
DeleteRoom(...)....189

Модуль интеграции Орион Про

Модуль интеграции обеспечивает информационное взаимодействие стороннего программного обеспечения с пакетом программного обеспечения АРМ "Орион Про" версии 1.12 и выше для аппаратно-программного комплекса ИСО "Орион".

Модуль интеграции является Web-сервисом, доступ к которому осуществляется по протоколам SOAP и JSON-RPC.

Может быть запущен как служба Windows, так и как обыкновенное приложение. Сервис работает под управлением операционных систем Windows 7 (32 bit или 64 bit) и выше.

Взаимодействие стороннего приложения и "Орион ПРО" схематично изображено на рис. 1.1.

Рис 1.1.

Установка и настройка

- Распаковать содержимое архива IntegrServ.zip, содержащего дистрибутив модуля интеграции, в любое место на диске.
- Запустить программу выполнив integrServ.exe, развернуть окно программы, кликнув на значок в системной панели Windows, и отредактировать настройки программы. Настройки программы будут сохранены в файле: "%ProgramData%\BolidIntegrServ\settings.ini", который допускается редактировать вручную. Если требуется, чтобы настройки и логи модуля интеграции хранились непосредственно в директории установки модуля интеграции, то необходимо перенести в нее файл «settings.ini”.

[Sockets]

SoapLogging=1 – *логирование SOAP запросов (0 — выкл, 1 — вкл)*

[OrionPro]

SynchWithModules=1 – *синхронизация изменений в БД СКУД с модулями «Орион Про» (0 — выкл, 1 — вкл)*

[CoreOrion]

SynchWithCores=1 – *синхронизация изменений в БД СКУД с ядром опроса (0 — выкл, 1 — вкл) (зависит от параметра* SynchWithModules*)*

XmlRpcServerPort=8184 – *Порт xml-rpc сервера (используется только при SynchWithCores = 1)*

XmlRpcLogging=1 — *логирование xml-rpc запросов (0 — выкл, 1 — вкл)*

[Sockets]

SoapLogging=0 — *логирование SOAP запросов(0 — выкл, 1 — вкл)*

JsonRpcLogging=0 – логирование JSON-RPC запросов (0 — выкл, 1 — вкл)

LogsLifeTimeMinutes=4320 — длительность хранения логов в минутах

ListenPort=8090 – *порт HTTP модуля интеграции*

ListenPortSSL=8091 – *порт HTTPS модуля интеграции*

SSLCertFile=C:\ProgramData\BolidIntegrServ\certificate.pem – *путь к файлу, содержащему SSL-сертификат*

SSLKeyFile=C:\ProgramData\BolidIntegrServ\private.key – *путь к файлу, содержащему приватный ключ SSL*

SSLPassword=orion – *пароль ключа SSL*

[DBMS]

HostName=127.0.0.1:1433\SQLEXPRESS – *адрес СУБД MS SQL Server*

Database=orion_base – *название базы данных*

UserName=sa — *имя пользоваля для доступа к СУБД MS SQ Server*

Password=123456 — *пароль для доступа к СУБД MS SQL Server*

[HttpAuth]

IsNeedAuth=*0 – первичная авторизация (0 — выкл, 1 — вкл)*

Login=login – *логин*

Password=password – *пароль*

[RemoteControl] — *параметры вторичной авторизации. Данный функционал поддерживается АРМ «Орион Про» начиная с версии 1.20.3*

TokenLifeTime=300 — *время жизни токена авторизованного пользователя (сек.) (0 — бесконечно)*

IsTokenRequired=0 — *требуется вторичная авторизация (0 — выкл, 1 — вкл)*

[SMTP] — *данные параметры необходимы при использовании механизма восстановления пароля для отправки электронного письма с временным паролем.*

UserName=ivanov — *имя пользователя SMTP*

Password=123456 – *пароль SMTP*

Host=mail-serv.bolid.ru — *адрес SMTP сервера*

Port=25 — *порт STMP сервера*

FromText=ivanov@bolid.ru — *адрес отправителя*

SubjText=Восстановление пароля — *текст темы письма*

BodyText=Your password: "%PASSWORD%" — *текст письма, где %PASSOWRD% - временный пароль*

Следует обратить внимание, что при включенных параметрах **“SoapLogging”, “JsonRpcLogging”** и **“XmlRpcLogging”**, модуль интеграции сохраняет в текстовые файлы весь входящий и исходящий трафик, что может привести к переполнению дискового пространства.

- Если необходимо, чтобы модуль интеграции работал как служба Windows, в директории модуля интеграции выполнить следующую команду: "integrServ.exe /INSTALL", после чего запустить службу через "Оснастку" Windows.

- Для включения первичной авторизации (HTTP basic authentication) необходимо в файле settings.ini параметру ”IsNeedAuth” раздела “HttpAuth” присвоить значение “1” и указать логин и пароль (параметры “Login” и “Password”).
- Для включения вторичной авторизации (поддерживается АРМ «Орион Про» начиная с версии 1.20.3) необходимо в файле settings.ini параметру ”IsTokenRequired” раздела “RemoteControl” присвоить значение “1”. Для ее использования в АБД требуется добавить пароль с типом «удаленное управление», затем посредством вызова GetLoginToken() получить токен, который далее будет использоваться при вызове методов. Т.к. токен имеет ограниченное время действия (задается параметром ”TokenLifeTime”), необходимо его продливать посредством вызова метода ExtendTokenExpiration().

Для ряда методов (ControlItems, GetItemsForLogin, GetItemsStates, GetADCValues) вторичная авторизация необходима независимо от значения параметра “IsTokenRequired”.

Интерфейс SOAP

WSDL документ, описывающий SOAP интерфейс доступен по ссылке: http://127.0.0.1:8090/wsdl/iorionpro (при условии, что модуль интеграции запущен на локальном компьютере).

Взаимодействие с SOAP интерфейсом осуществляется по адресу: http://127.0.0.1:8090/soap/iorionpro.

Интерфейс JSON-RPC 2.0

Взаимодействие с JSON-RPC интерфейсом осуществляется по адресу: http://127.0.0.1:8090/jsonrpc/iorionpro.

Описание типов данных и методов SOAP и JSON-RPC интерфейсов

TServiceError

Свойства типа

Свойство	Тип	Описание
ErrorCode	**string**	Код ошибки
Description	**string**	Описание ошибки
InnerExceptionMessage	**string**	Сообщение об ошибке, содержащееся в исходном исключении

Пример описания в формате JSON:

```
{
  "errorCode": "0",
  "description": "Unrecognized error",
  "innerExceptionMessage": "Token not found"
}
```

TOperationResult<T>

Свойства типа

Свойство	Тип	Описание
Success	**bool**	Признак успешности операции
Result	**T**	Результат операции
Error	**TserviceError**	Информация об ошибке

В случае успеха - результат записывается в свойство **Result<T>** и имеет соответствующий тип **T**.

В случае ошибки — информация об ошибке записывается в **TOperationResult.Error**, а свойство **Result** равно Null.

Пример описания в формате JSON:

```
 {
      "success": true,
      "operationResult":       {
         "serviceVersion": "2.0.0.1",
         "softwareVersion": "1.20.3.8",
         "softwareName": "АРМ Орион Про",
         "databaseVersion": "MSSQL",
         "serverTime": "2025-05-06T15:47:30.000Z"
      }
 }
```

TServiceInfo

Свойства типа

Свойство	Тип	Описание
ServiceVersion	**string**	Версия модуля интеграции
SoftwareVersion	**string**	Версия ПО АРМ «Орион Про»
SoftwareName	**string**	Наименование АРМ «Орион Про»
DatabaseVersion	**string**	Версия БД АРМ «Орион Про»
ServerTime	**DateTime**	Текущее время на сервере

Пример описания в формате JSON:

```
{
  "serviceVersion": "2.0.0.1",
  "softwareVersion": "1.20.3.8",
  "softwareName": "АРМ Орион Про",
  "databaseVersion": "MSSQL",
  "serverTime": "2025-04-06T15:47:30.000Z"
}
```

TItemTypeInfo

Свойства типа

Свойство	Тип	Описание
ItemType	**string**	Тип элемента
AdditionalType	**int**	Подтип элемента
Name	**string**	Название
DeviceVersion	**int**	Версия устройства
EventGroupId	**int**	Группа событий для данного типа элемента

Пример описания в формате JSON:

```
{
  "itemType": "LOOP",
  "additionalType": 1,
  "name": "Охранный",
  "deviceVersion": 0,
  "eventGroupId": 48
}
```

TEventType

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор типа события
CharId	**string**	Международное обозначение
Description	**string**	Описание события
Comments	**string**	Комментарий
Category	**string**	Название категории события
HexCode	**string**	Код цвета события (RGB)
IsAlarm	**bool**	Признак тревожного события

Пример описания в формате JSON:

```
{
  "id": 1,
  "charId": "AR",
  "description": "Восстановление сети 220",
  "category": "Отключения\\включения",
  "hexColor": "#000000",
  "isAlarm": false,
  "comments": "Сеть 220В восстановлена"
```

}

TEvent

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор записи (ID) в журнале событий
EventId	**string**	Уникальный идентификатор записи (GUID) в журнале событий
EventTypeId	**int**	Идентификатор типа события (TEventType.Id)
EventDate	**DateTime**	Дата и время события
Description	**string**	Описание события (комментарий)
ComputerId	**int**	Идентификатор компьютера
ComPortNumber	**int**	Номер последовательного порта
PKUAddress	**int**	Адрес сетевого контроллера
DevAddress	**int**	Адрес устройства
ZoneAddress	**int**	Адрес зоны
ItemId	**int**	Идентификатор элемента, по которому произошло событие
ItemType	**string**	Тип элемента, по которому произошло событие ”SECTION” - зона; “LOOP” - вход (шлейф); “DEVICE” - устройство; “READER” - считыватель; “RELAY” - выход (реле); “ACCESSZONE” - зона доступа; “ACCESSPOINT” - точка доступа (дверь); “SECTIONGROUP” - группа зон.
AccessPointId	**int**	Идентификатор точки доступа (двери)
AccessPointName	**string**	Наименование точки доступа (двери)
AccessZoneId	**int**	Идентификатор зоны доступа
SectionId	**int**	Идентификатор зоны/группы зон в зависимости от значения поля ItemType: идентификатор группы зон, если свойство ItemType имеет значение «SECTIONGROUP”, в противном случае — идентификатор зоны.
PassMode	**int**	Режим прохода (0 – не определен, 1 – вход, 2 – выход)
CardNo	**string**	Номер карты доступа
PersonId	**int**	Идентификатор сотрудника
LastName	**string**	Фамилия владельца ключа
FirstName	**string**	Имя владельца ключа
MiddleName	**string**	Отчество владельца ключа
BirthData	**DateTime**	Дата рождения владельца ключа
TabNum	**string**	Табельный номер владельца ключа
AccessFlags	**TEvetAccess Flags**	Флаги доступа

StrAddr	**string**	Строковый адрес элемента

Пример описания в формате JSON:


```
{
  "id": 214286,
  "eventId": "{B8A4F78C-9125-F011-B7FF-107C61D31197}",
  "eventTypeId": 251,
  "eventDate": "2025-04-30T10:05:51.000Z",
  "description": "",
  "computerId": 1,
  "comPortNumber": 70,
  "pKUAddress": 174,
  "devAddress": 219,
  "zoneAddress": 29,
  "accessPointId": 0,
  "accessPointName": "",
  "accessZoneId": 0,
  "passMode": 0,
  "cardNo": "",
  "personId": 0,
  "lastName": "",
  "firstName": "",
  "middleName": "",
  "birthDate": "",
  "tabNum": "",
  "itemId": 0,
  "itemType": "DEVICE",
  "sectionId": 0,
  "readerId": -1,
  "accessFlags": {
     "isBlocked": false,
     "isNoRights": false,
     "isBrokenAntipassback": false,
     "isBrokenTimeWindow": false,
     "isExpired": false,
     "isAdditionalCodeInputError": false,
     "isDuressCode": false,
     "isConfirmationCodeError": false,
     "isConfirmationWaiting": false
  },
  "strAddr": "\\C2000-E:192.168.201.161"
}
```


TAccessFlags

Свойства типа

Свойство	Тип	Описание
IsBlocked	**bool**	Идентификатор заблокирован
IsNoRichts	**bool**	Нет прав доступа
IsBrokenAntipassback	**bool**	Нарушено правило antipassback
IsBrokenTimeWindow	**bool**	Нарушение окна времени
IsExpired	**bool**	Истек или не начался срок действия идентификатора
IsAdditionalCodeInputEror	**bool**	Ошибка ввода дополнительного кода

IsDuressCode	**bool**	Предъявлен код принуждения
IsConfirmationCodeError	**bool**	Ошибка подтверждающего кода
IsConfirmationWaiting	**bool**	Ожидание подтверждения по кнопке

Пример описания в формате JSON:

```
{
   "isBlocked": false,
   "isNoRights": false,
   "isBrokenAntipassback": false,
   "isBrokenTimeWindow": false,
   "isExpired": false,
   "isAdditionalCodeInputError": false,
   "isDuressCode": false,
   "isConfirmationCodeError": false,
   "isConfirmationWaiting": false
}
```

TEventsGroup

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор группы событий
Name	**int**	Название группы событий
Comment	**DateTime**	Комментарий
Events	**int[]**	Список кодов событий, входящих в группу

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "ГС для трансляции – ‘Пожары’",
  "comment": "",
  "events": [
    37,
    44,
    139,
    141,
    143,
    144,
    145,
    146,
    147,
    43
  ]
}
```

TTimeWindow

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор
Name	**string**	Наименование окна времени

Calendar	**byte[12*31]**	Календарь на весь год. На каждый месяц отведен 31 байт, каждый байт хранит код одного дня месяца. Коды: 0..13 — дни двух недель, описанные во временном интервале; 14 – праздник; 15 – обычный день (дни недели считаются по общепринятому календарю).
TimeIntervals	**TTimeInterval[]**	Временные интервалы окна времени

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Всегда",
  "calendar": [
     0,
     0
  ],
  "timeIntervals": [{
    "id": 1,
    "timeWindowId": 1,
    "startTime": "2005-01-01T00:00:00.000Z",
    "endTime": "2019-01-01T23:59:00.000Z",
    "days": [
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true
    ],
    "isEnterActivity": true,
    "isExitActivity": true,
    "isAccessSaturday": true,
    "isAccessSunday": true,
    "isAccessHoliday": true
  }]
}
```

TTimeInterval

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор
TimeWindowId	**int**	Уникальный идентификатор окна времени

StartTime	**DateTime**	Время начала действия временного интервала
EndTime	**DateTime**	Время окончания действия временного интервала
Days	**bool[15]**	Дни недели действия временного интервала. Байты 0..13 — флаги действия временного интервала в дни двух недель. Байт 14 — флаг действия временного интервала в праздник.
IsEnterActivity	**bool**	Признак активности временного интервала на вход
IsExitActivity	**bool**	Признак активности временного интервала на выход
IsAccessSaturday	**bool**	Доступ по субботам
IsAccessSunday	**bool**	Доступ по воскресеньям
IsAccessHoliday	**bool**	Доступ в выходные дни

Пример описания в формате JSON:


```
{
    "id": 1,
    "timeWindowId": 1,
    "startTime": "2005-01-01T00:00:00.000Z",
    "endTime": "2019-01-01T23:59:00.000Z",
    "days": [
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true,
      true
    ],
    "isEnterActivity": true,
    "isExitActivity": true,
    "isAccessSaturday": true,
    "isAccessSunday": true,
    "isAccessHoliday": true
}
```


TAccessLevelItem

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор
ItemType	**string**	Тип логического объекта СКУД/ОПС: ”SECTION” - зона; “ACCESSZONE” - зона доступа; “ACCESSPOINT” - точка доступа (дверь); “SECTIONGROUP” - группа зон.

ItemId	**int**	Идентификатор объекта СКУД/ОПС
Rights	**int**	Для точки доступа и зоны доступа: 0 — проход; 1 — вход; 2 — выход; 3 — вход/выход. Для зоны и группы зон. 1 — взятие; 2 — снятие; 8 — включение атвоматики; 16 — выключение автоматики; 32 — пуск пожаротушения; 64 — отмена пуска пожаротушения.
TimeWindowId	**int**	Идентификатор окна времени
Antipassback	**int**	Режим antipassback: 0 — нет; 1 — строгий; 2 — временной; 3 — мягкий.
LockTime	**int**	Время разблокировки для временного antipassback в минутах
IsZonalAntipassback	**bool**	Зональный antipassback
DoubleConfirmationId	**int**	ID подтверждающего уровня доступа для подтверждения по двум лицам
TripleConfirmationId	**int**	ID подтверждающего уровня доступа для потдверждения по трем лицам
IsConfirming	**bool**	Признак подтверждающего уровня доступа
IsConfirmationButton	**bool**	Подтверждение по кнопке
IsAddCode	**bool**	Дополнительный код.

Пример описания в формате JSON:

```
{
  "id": 1,
  "itemType": "SECTION",
  "itemId": 0,
  "timeWindowId": 1,
  "rights": 3,
  "antipassback": 0,
  "lockTime": 0,
  "isZonalAntipassback": false,
  "doubleConfirmationId": 0,
  "tripleConfirmationId": 0,
  "isConfirming": false,
  "isConfirmationButton": false,
  "isAddCode": false
}
```

TAccessLevel

Свойства типа

Свойство	Тип	Описание

Id	**int**	Уникальный идентификатор
Name	**string**	Наименование уровня доступа
Description	**string**	Описание уровня доступа (комментарий)
Items	**TAccessLevelItem[]**	Элементы уровня доступа

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Запрет",
  "description": "Запрещается проход через двери и управление разделами",
  "items": []
}
```

TAccessZone

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор
Name	**string**	Наименование зоны доступа

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Внешний мир"
}
```

TEntryPoint

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор
Name	**string**	Наименование точки доступа
EnterAccessZoneId	**int**	Идентификатор зоны доступа на вход
ExitAccessZoneId	**int**	Идентификатор зоны доступа на выход
EntryPointType	**int**	Тип точки доступа: 1 — однонаправленная дверь; 2 — одна дверь на вход/выход; 3 — турникет; 4 — шлагбаум; 5 — шлюз.

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Цоколь",
  "enterAccessZoneId": 2,
  "exitAccessZoneId": 1,
```

```
  "entryPointType": 2,
  "readers": [
     108,
     109
  ],
  "index": 1
}
```

TPersonData

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор
LastName	**string**	Фамилия
FirstName	**string**	Имя
MiddleName	**string**	Отчество
Address	**string**	Домашний адрес сотрудника
BirthDate	**DateTime**	Дата рождения
CompanyId	**int**	Идентификатор организации
DepartmentId	**int**	Идентификатор подразделения
PositionId	**int**	Идентификатор должности
Company	**string**	Наименование организации
Department	**string**	Наименование подразделения
Position	**string**	Наименование должности
TabNum	**string**	Табельный номер
Photo	**byte[]**	Фото (формат BMP или JPEG)
Phone	**string**	Контактный телефон
HomePhone	**string**	Домашний телефон
AccessLevelId	**int**	Идентификатор уровня доступа (для присваивания графика работы данного уровня доступа вновь создаваемому подразделению)
Status	**int**	Статус сотрудника 1 — владелец; 2 — администратор; 3 — дежурный офицер; 4 — дежурный оператор; 5 — хозорган; 6 — служащий; 7 — оператор бюро пропусков; 8 — посетитель.
ContactIdIndex	**int**	Индекс Contact ID
IsLockedDayCrossing	**bool**	Флаг - запрет перехода через сутки
IsFreeShedule	**bool**	Флаг - свободный график
ExternalId	**string**	Идентификатор сотрудника во внешней системе
AdditionalExternalId	**string**	Дополнительный идентификатор сотрудника во внешней системе
IsInArchive	**bool**	Признак того, что сотрудник помещен в архив

DocumentType	**int**	Тип документа
DocumentNumber	**string**	Номер документа
DocumentSerials	**string**	Серия документа
DocumentIssueData	**DateTime**	Дата выдачи документа
DocumentEndingDate	**DateTime**	Дата окончания действия документа
DocumentIsser	**string**	Кем выдан документ
DocumentIsserCode	**string**	Код подразделения, выдавшего документ
Sex	**int**	Пол
Birthplace	**string**	Место рождения
EmailList	**string**	Список e-mail адресов сотрудника
ArchivingTimeStamp	**DateTime**	Дата помещения сотрудника в архив
IsInBlackList	**bool**	Признак того, что сотрудник находится в черном списке
IsDismissed	**bool**	Признак того, что сотрудник уволен
BlackListComment	**string**	Причина занесения сотрудника в черный список
DismissedComment	**string**	Причина увольнения сотрудника
ChangeTime	**DateTime**	Дата и время последнего редактирования сотрудника
Itn	**string**	ИНН
RoomId	**int**	Идентификатор комнаты
ExitGroupId	**int**	Идентификатор группы эвакуации
ExitGroup	**string**	Наименование группы эвакуации

Пример описания в формате JSON:


```
{
  "id": 1,
  "lastName": "Иванов",
  "firstName": "Иван",
  "middleName": "Иванович",
  "birthDate": "",
  "company": "",
  "department": "",
  "position": "",
  "companyId": 0,
  "departmentId": 0,
  "positionId": 0,
  "tabNum": "1",
  "phone": "",
  "homePhone": "",
  "address": "",
  "photo": "",
  "accessLevelId": 0,
  "status": 1,
  "contactIdIndex": 0,
  "isLockedDayCrossing": false,
  "isFreeShedule": false,
  "externalId": "",
  "additionalExternalId": "",
  "isInArchive": false,
  "documentType": 0,
  "documentSerials": "",
```

```
  "documentNumber": "",
  "documentIssueDate": "",
  "documentEndingDate": "",
  "documentIsser": "",
  "documentIsserCode": "",
  "sex": 0,
  "birthplace": "",
  "emailList": "",
  "archivingTimeStamp": "",
  "isInBlackList": false,
  "isDismissed": false,
  "blackListComment": "",
  "changeTime": "2025-04-03T15:16:47.000Z",
  "itn": "",
  "dismissedComment": "",
  "roomId": 0,
  "exitGroupId": 0,
  "exitGroup": ""
}
```


TKeyData

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор ключа
CodeType	**int**	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
Code	**string**	Код ключа
AdditionalCodeType	**int**	Тип кода дополнительного ключа
AdditionalCode	**string**	Код дополнительного ключа
PersonId	**int**	Идентификатор владельца ключа
AccessLevelId	**int**	Уровень доступа ключа
StartDate	**DateTime**	Дата начала срока действия ключа
EndData	**DateTime**	Дата окончания срока действия ключа
IsBlocked	**bool**	Признак блокировки ключа
IsStoreInDevice	**bool**	Хранить ключ в приборе
IsStoreInS2000	**bool**	Хранить ключ в пульте С2000
IsInStopList	**bool**	Карта в стоп-листе

Comment	**string**	Комментарий

Пример описания в формате JSON:

```
{
  "id": 1,
  "codeType": 1,
  "additionalCodeType": 0,
  "code": "1",
  "additionalCode": "",
  "personId": 1,
  "accessLevelId": 2,
  "startDate": "2024-02-11T12:26:34.000Z",
  "endDate": "2035-02-11T12:26:34.000Z",
  "isBlocked": false,
  "isStoreInDevice": false,
  "isStoreInS2000": false,
  "isInStopList": false,
  "comment": "",
  "login": ""
}
```

TCompany

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор организации
Name	**string**	Наименование организации
Address	**string**	Адрес организации
Phone	**string**	Контактный телефон

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Организация",
  "address": "",
  "phone": "",
  "companyType": 0
}
```

TDepartment

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор подразделения
Name	**string**	Наименование подразделения
CompanyId	**int**	Идентификатор организации-владельца

AccessLevelId	**int**	Идентификатор уровня доступа (для присваивания графика работы данного уровня доступа вновь создаваемому подразделению)

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Подразделение",
  "description": "",
  "companyId": 1,
  "accessLevelId": 0
}
```

TPosition

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор должности
Name	**string**	Наименование должности

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Должность 1"
}
```

TExitGroup

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор группы эвакуации
Name	**string**	Наименование группы эвакуации
Comment	**string**	Комментарий

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Группа 1",
  "comment": ""
}
```

TComputer

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор компьютера
Ip	**string**	Ip-адрес компьютера
Name	**string**	Имя компьютера
Number	**int**	Пользовательский номер

Пример описания в формате JSON:

```
{
  "id": 1,
  "ip": "127.0.0.1",
  "name": "PC1",
  "number": 1
}
```

TComPort

Свойства типа

Свойство	Тип	Описание
Id	**int**	Уникальный идентификатор порта
Number	**int**	Номер порта
ComputerId	**int**	Идентификатор компьютера

Пример описания в формате JSON:

```
{
  "id": 1,
  "number": 1,
  "computerId": 1
}
```

TDeviceItem

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор элемента устройства
Name	**string**	Название элемента устройства
DeviceId	**int**	Идентификатор устройства
ItemType	**string**	Тип элемента устройства ”LOOP” - вход (шлейф); ”READER” - считыватель; ”RELAY” - выход (реле).
Address	**int**	Адрес элемента относительно устройства
ContactId	**int**	Номер зоны Contact ID
ComputerId	**int**	Идентификатор компьютера, которому принадлежит элемент устройства
Number	**int**	Пользовательский номер

Пример описания в формате JSON:

```
{
  "id": 2195,
  "name": "Считыватель 1, Прибор 126",
  "deviceId": 1,
  "itemType": "READER",
  "address": 1,
```

```
  "additionalType": 0,
  "remoteControlCategoryId": 5,
  "contactId": 0,
  "computerId": 1,
  "number": 3506
}
```


TDevice

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор устройства
Address	**int**	Адрес устройства
DevType	**int**	Тип устройства
Name	**string**	Название устройства
ComPortId	**int**	Идентификатор порта
PKUId	**int**	Идентификатор сетевого контроллера (С2000М, ППКУП ”Сириус”)
ComputerId	**int**	Идентификатор компьютера, которому принадлежит устройство
Number	**int**	Пользовательский номер

Пример описания в формате JSON:


```
{
  "id": 1,
  "address": 126,
  "devType": 0,
  "name": "C2000 (1)",
  "comPortId": 1,
  "pKUId": 0,
  "hwId": 0,
  "ipAddress": "",
  "computerId": 1,
  "number": 1
}
```


TSection

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор зоны
Number	**int**	Номер зоны
Name	**string**	Название зоны
ComputerId	**int**	Идентификатор компьютера

Пример описания в формате JSON:


```
{
  "id": 3,
  "number": 7,
  "name": "7 основной",
```

```
  "computerId": 1,
  "sectionTypeId": 0,
  "sectionTypeName": "",
  "items": [],
  "remoteControlCategoriesIds": [2]
}
```

TSectionsGroup

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор группы зон
Number	**int**	Номер группы зон
Name	**string**	Название группы зон
ComputerId	**int**	Идентификатор компьютера

Пример описания в формате JSON:

```
{
  "id": 3,
  "number": 111,
  "name": "Пожарка 1 этаж",
  "computerId": 1,
  "items": [],
  "remoteControlCategoriesIds": [2]
}
```

TItem

Свойства типа

Свойство	Тип	Описание
ItemType	**string**	Тип элемента системы ”SECTION” - зона; “LOOP” - вход (шлейф); “DEVICE” - устройство; “READER” - считыватель; “RELAY” - выход (реле); “ACCESSZONE” - зона доступа; “ACCESSPOINT” - точка доступа (дверь); “SECTIONGROUP” - группа зон.
ItemId	**int**	Идентификатор элемента системы
State	**int**	Код состояния элемента системы
ComputerId	**int**	Идентификатор компьютера
OwnerId	**int**	Идентификатор хоз. органа, изменившего состояние
TimeStamp	**DateTime**	Время последнего изменения состояния
Rights	**int**	Действия, разрешенные для текущего пользователя: Для точки доступа: 0 – проход; 1 – вход; 2 –выход.

		Для зоны или группы зон: 0 – просмотр; 1 – взятие; 2 – снятие; 3 – взятие/снятие.

Пример описания в формате JSON:

```
{
  "itemType": "LOOP",
  "itemId": 509,
  "rights": 0,
  "state": 47,
  "computerId": 1,
  "ownerId": 0,
  "timestamp": "2021-12-20T15:17:55.000Z"
}
```

TReaderItem

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор связи элемента со считывателем
ReaderId	**int**	Идентификатор считывателя
ItemId	**int**	Идентификатор элемента (зона доступа, точка доступа, зона, группа зон), связанного с данным считывателем
ItemType	**string**	Тип элемента, связанного с данным считывателем: ”SECTION” - зона; “ACCESSZONE” - зона доступа; “ACCESSPOINT” - точка доступа (дверь); “SECTIONGROUP” - группа зон.
Rights	**int**	Разрешенные действия с данного считывателя: Для точки доступа: 0 – проход; 1 – вход; 2 –выход. Для рзоны или группы зон: 0 – просмотр; 1 – взятие; 2 – снятие; 3 – взятие/снятие.

Пример описания в формате JSON:

```
{
  "id": 910,
  "readerId": 2086,
  "itemId": 1,
  "itemType": "ACCESSPOINT",
  "rights": 1
}
```

TItemADCValue

Свойства типа

Свойство	Тип	Описание
TimeStamp	**DateTime**	Дата и время получения значения АЦП
ComputerId	**int**	Идентификатор компьютера
ItemId	**int**	Идентификатор элемента
ItemType	**string**	Тип элемента системы “LOOP” - вход (шлейф); “DEVICE” - устройство; “READER” - считыватель; “RELAY” - выход (реле).
Value	**float**	Значение АЦП
ADCType	**string**	Тип данных: 1 — дымовые показатели; 2 – сопротивление; 3 – температура; 4 – влажность.

Пример описания в формате JSON:

```
{
  "timestamp": "2025-01-31T08:16:33.000Z",
  "computerId": 1,
  "itemId": 4,
  "itemType": "RELAY",
  "value": 10.38,
  "adcType": "Напряжение выхода"
}
```

TScript

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор сценария управления
Name	**string**	Идентификатор компьютера
Comment	**string**	Комментарий

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Сценарий 1",
  "comment": ""
}
```

TVisitData

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор посетителя

PersonId	**int**	Идентификатор информации о посетителе
VisitedPersonId	**int**	Идентификатор сотрудника, к которому идет посетитель
VisitedCompanyId	**int**	Идентификатор компании, в котороую идет посетитель
VisitDate	**DateTime**	Дата и время начала визита
VisitedRoom	**string**	Посещаемая комната
VisitedPersonPhone	**string**	Номер телефона сотрудника, к которому идет посетитель
VisitPurpose	**string**	Цель визита
ImpersonalPersonId	**int**	Идентификатор обезличенного сотрудника (шаблона посетителя)
AccessControlRule	**int**	Правило управления доступом: 0-й бит — при выходе установить «Запрет»; 1-й бит — установить шаблон и заблокировать; 2-й бит — установить шаблон; 3-й бит — установить шаблон и при выходе поместить в архив; 4-й бит — установить шаблон и срок действия ключа ограничить визитом.
VisitEndDateTime	**DateTime**	Дата и время окончания визита
InstructingPersonId	**int**	Ссылка на сотрудника, проведшего инструктаж
CarName	**string**	Марка автомобиля
CarNumber	**string**	Номер автомобиля
CarColor	**string**	Цвет автомобиля
IsCarOvernight	**bool**	Признак того, что автомобиль оставлен на ночь

Пример описания в формате JSON:

```
{
  "Id": 1,
  "PersonId": 43,
  "VisitedPersonId": 29,
  "VisitedCompanyId": 1,
  "VisitedDepartmentId": 1,
  "VisitDate": "2021-12-25T03:00:00.000Z",
  "VisitedRoom": "",
  "VisitedPersonPhone": "",
  "VisitPurpose": "",
  "ImpersonalPersonId": 1,
  "AccessControlRule": 11,
  "VisitEndDateTime": "2021-12-25T23:59:00.000Z",
  "InstructingPersonId": 1,
  "CarName": "",
  "CarNumber": "",
  "CarColor": "",
  "IsCarOvernight": true
}
```

TCarData

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор автомобиля
Model	**string**	Модель

Color	**string**	Цвет
Number	**string**	Гос. номер автомобиля
Vin	**string**	VIN автомобиля
UserIndex	**int**	Пользовательский номер

Пример описания в формате JSON:

```
{
  "id": 1,
  "model": "Tesla Model X",
  "color": "blue",
  "number": "a123aa799",
  "userIndex": 1,
  "vin": "GF545345HHS"
}
```

TPersonCustomField

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор пользовательского поля события
Name	**string**	Название пользовательского поля сотрудника
Value	**string**	Значение пользовательского поля сотрудника

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Пользовательское поле 1",
  "value": ""
}
```

TAbsenceReasonType

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор уважительной причины
Name	**string**	Описание уважительной причины

Пример описания в формате JSON:

```
{
  "id": 1,
  "name": "Уважительная причина"
}
```

TAbsenceReason

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор причины отсутствия сотрудника

PersonId	**int**	Идентификатор сотрудника
StartDate	**DateTime**	День начала причины отсутствия
EndDate	**DateTime**	День окончания причины отсутствия
ReasonTypeId	**int**	Идентификатор уважительной причины
Comment	**string**	Комментарий
StartTime	**DateTime**	Время начала причины отсутствия
EndTime	**DateTime**	Время окончания причины отсутствия
Operator	**string**	ФИО оператора добавившего/редактировавшего причину отсутствия
IsValidReason	**bool**	Флаг уважительной причины отсутствия
IsIgnoreInCalc	**bool**	Не учитывать при расчетах

Пример описания в формате JSON:

```
{
  "id": 1,
  "personId": 1,
  "startDate": "2024-12-19T00:00:00.000Z",
  "endDate": "2024-12-25T00:00:00.000Z",
  "reasonTypeId": 1,
  "comment": "",
  "startTime": "2024-12-25T09:00:00.000Z",
  "endTime": "2024-12-25T12:00:00.000Z",
  "operator": "",
  "isValidReason": true,
  "isIgnoreInCalc": true
}
```

TExternalEvent

Свойства типа

Свойство	Тип	Описание
Id	**int**	Идентификатор внешнего события
ItemId	**int**	Идентификатор элемента, по которому произошло событие
ItemType	**string**	Тип элемента, по которому произошло событие. ”SECTION” - зона; “LOOP” - вход (шлейф); “DEVICE” - устройство; “READER” - считыватель; “RELAY” - выход (реле); “ACCESSZONE” - зона доступа; “ACCESSPOINT” - точка доступа (дверь); “SECTIONGROUP” - группа зон.
Event	**int**	Код события
TimeStamp	**DateTime**	Время события
PersonId	**int**	Идентификатор сотрудника
KeyId	**int**	Идентификатор ключа
Text	**int**	Текст события
AccessMode	**int**	Направление прохода.

		0 — проход; 1 — вход; 2 — выход.

Пример описания в формате JSON:

```
{
  "id": 1,
  "itemId": 1,
  "itemType": "ACCESSPOINT",
  "event": 28,
  "timestamp": "2024-12-25T09:00:00.000Z",
  "personId": 1,
  "keyId": 1,
  "text": "",
  "accessMode": 1
}
```

TDbChangeLogItem

Свойства типа

Свойство	Тип	Описание
Id	**string**	Идентификатор события об изменении данных
TimeStamp	**DateStamp**	Время изменения
Operation	**int**	Выполненная операция. 0 — добавление; 1 — изменение; 2 — удаление.
ValueId	**id**	Идентификатор записи, над которой выполнена операция
OperatorId	**int**	Идентификатор оператора, выполнившего операцию
Number	**int**	Идентификатор события об изменении данных
XmlData	**string**	Данные в формате xml

Пример описания в формате JSON:

```
{
  "id": 1,
  "timestamp": "2024-12-25T09:00:00.000Z",
  "operation": 0,
  "valueId": 1,
  "operatorId": 1,
  "number": 1,
  "xmlData": "<inserted ID="1" Name="Внешний мир" GIndex="0"/>"
}
```

GetReplService(...)

`bool GetReplService()` - проверка доступности модуля интеграции, возможности подключения к нему.

Входные данные

Нет.

Результирующие данные

Метод всегда возвращает значение «1».

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getReplService",
  "params": {},
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result": true
}
```

GetServiceInfo(...)

`TOperationResult<TServiceInfo> GetServiceInfo()` - получение информации о модуле интеграции.

Входные данные

Нет.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TServiceInfo>**, содержащий информацию о модуле интеграции или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getServiceInfo",
  "params": {},
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result":   {
    "success": true,
    "operationResult":     {
      "serviceVersion": "1.0.0.88",
      "softwareVersion": "1.20.3.8",
      "softwareName": "АРМ Орион Про",
      "databaseVersion": "MSSQL",
      "serverTime": "2025-05-23T15:43:11.000Z"
    }
  }
```

}

GetAdditionalTypesForItemType(...)

`TOperationResult<TItemTypeInfo[]> GetAdditionalTypesForItemType(itemType, token)` - получение списка подтипов элемента для указанного типа из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
itemType	int	Тип элемента
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItemTypeInfo[]>**, содержащий список объектов с информацией о подтипе элемента или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAdditionalTypesForItemType",
  "params": {
    "itemType": "SECTION"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "itemType": "SECTION",
         "additionalType": 0,
         "name": "Зона",
         "deviceVersion": 0,
         "eventGroupId": 16
      }]
   }
}
```

GetEventTypes(...)

`TOperationResult<TEventType[]> GetEventTypes(token)` - получение списка типов событий из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется

		при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEventType[]>**, содержащий список типов событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventTypes",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "charId": "AR",
            "description": "Восстановление сети 220",
            "category": "Отключения\\включения",
            "hexColor": "#000000",
            "isAlarm": false,
            "comments": "Сеть 220В восстановлена"
         },
                  {
            "id": 2,
            "charId": "AT",
            "description": "Авария сети",
            "category": "Взятия\\снятия зон",
            "hexColor": "#000000",
            "isAlarm": false,
            "comments": "Отключение сети 220В"
         }
      ]
   }
}
```

GetEventTypesForItemType(...)

`TOperationResult<TEventType[]> GetEventTypesForItemType(itemType, additionalType, token)` - получение списка возможных типов событий для типа элемента из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание

itemType	string	Тип элемента: ”SECTION” - зона; “LOOP” - вход (шлейф); “DEVICE” - устройство; “READER” - считыватель; “RELAY” - выход (реле); “ACCESSZONE” - зона доступа; “ACCESSPOINT” - точка доступа (дверь); “SECTIONGROUP” - группа зон.
additionalType	int	Тип устройства, входа, выхода.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEventType[]>**, содержащий список типов событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventTypesForItemType",
  "params": {
    "itemType": "LOOP",
    "additionalType": 1
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 3,
            "charId": "BA",
            "description": "Тревога проникновения",
            "category": "Взятия\\снятия\\нормы",
            "hexColor": "#000000",
            "isAlarm": true,
            "comments": "Нарушение взятого на охрану охранного Входа"
         },
                  {
            "id": 17,
            "charId": "CI",
            "description": "Неудачное взятие",
            "category": "Взятия\\снятия зон",
            "hexColor": "#000000",
            "isAlarm": false,
            "comments": "В момент постановки на охрану Вход был нарушен или
неисправен"
         }
```

```
        ]
    }
}
```

GetEventsGroups(...)

`TOperationResult<TEventsGroup[]> GetEventsGroups(token)` - получение списка групп событий из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEventsGroup[]>**, содержащий список групп событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventsGroups",
  "params": {
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "name": "ГС для трансляции – 'Пожары'",
            "comment": "",
            "events":             [
               37,
               44
            ]
         },
      ]
   }
}
```

GetEventsGroupById(...)

`TOperationResult<TEventsGroup> GetEventsGroupById(id, token)` - получение группы событий по её идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Уникальный идентификатор группы событий
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEventsGroup>**, содержащий информацию о группе событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventsGroupById",
  "params": {
    "id": 1
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "ГС для трансляции – ‘Пожары’",
         "comment": "",
         "events":          [
            37,
            44
         ]
      }
   }
}
```

GetEventsGroupsByEventTypeId(...)

`TOperationResult<TEventsGroup[]> GetEventsGroupById(eventTypeId, token)` - получение списка групп событий по коду события.

Входные данные

Параметр	Тип	Описание
eventTypeId	int	Уникальный идентификатор типа события
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEventsGroup[]>**, содержащий список групп событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventsGroupsByEventTypeId",
  "params": {
    "eventTypeId": 250,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 3,
            "name": "ГС для трансляции – 'Неисправности'",
            "comment": "",
            "events":             [
               1,
               2
            ]
         }
      ]
   }
}
```

GetEvents(...)

`TOperationResult<TEvent[]> GetEvents(beginTime, endTime, eventTypes, offset, count, persons, entryPoints, sections, sectionsGroups, token)` - получение событий, согласно переданному фильтру.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
eventTypes	TeventType[]	Список типов событий для фильтрации
offset	int	Смещение относительно начала полного списка событий
count	int	Количество возвращаемых событий начиная с позиции offset
persons	TPersonData[]	Список сотрудников для фильтрации

entryPoints	TEntryPoint[]	Список точек доступа для фильтрации
sections	TSection[]	Список зон для фильтрации
sectionGroups	TSectionsGroup[]	Список групп зон для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**). При указании данного параметра метод возвращает события с учетом уровня доступа.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEvent[]>**, содержащий список событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEvents",
  "params": {
    "beginTime": "2025-01-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "eventTypes": [],
    "offset": 0,
    "count": 2,
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 314,
            "eventId": "{C0F66C40-A94C-4D96-8BEE-53B6E4817224}",
            "eventTypeId": 90,
            "eventDate": "2025-03-21T16:45:15.000Z",
            "description": "ППКУП Сириус ВЕДОМЫЙ 219",
            "computerId": 1,
            "comPortNumber": 222,
            "pKUAddress": 20,
            "devAddress": 40,
            "zoneAddress": 67,
            "accessPointId": 0,
            "accessPointName": "",
            "accessZoneId": 0,
            "passMode": 0,
            "cardNo": "",
            "personId": 0,
            "lastName": "",
            "firstName": "",
            "middleName": "",
```

```
            "birthDate": "",
            "tabNum": "",
            "itemId": 2,
            "itemType": "DEVICE",
            "sectionId": 14,
            "readerId": -1,
            "accessFlags":             {
               "isBlocked": false,
               "isNoRights": false,
               "isBrokenAntipassback": false,
               "isBrokenTimeWindow": false,
               "isExpired": false,
               "isAdditionalCodeInputError": false,
               "isDuressCode": false,
               "isConfirmationCodeError": false,
               "isConfirmationWaiting": false
            },
            "strAddr": "\\Сириус:192.168.201.226"
         },
                  {
            "id": 315,
            "eventId": "{2B70E2D0-425E-4E79-AF22-33A4182C409C}",
            "eventTypeId": 255,
            "eventDate": "2025-03-21T17:00:01.000Z",
            "description": "ППКУП Сириус ВЕДОМЫЙ 219",
            "computerId": 1,
            "comPortNumber": 222,
            "pKUAddress": 20,
            "devAddress": 40,
            "zoneAddress": 67,
            "accessPointId": 0,
            "accessPointName": "",
            "accessZoneId": 0,
            "passMode": 0,
            "cardNo": "",
            "personId": 0,
            "lastName": "",
            "firstName": "",
            "middleName": "",
            "birthDate": "",
            "tabNum": "",
            "itemId": 2,
            "itemType": "DEVICE",
            "sectionId": 14,
            "readerId": -1,
            "accessFlags":             {
               "isBlocked": false,
               "isNoRights": false,
               "isBrokenAntipassback": false,
               "isBrokenTimeWindow": false,
               "isExpired": false,
               "isAdditionalCodeInputError": false,
               "isDuressCode": false,
               "isConfirmationCodeError": false,
               "isConfirmationWaiting": false
            },
            "strAddr": "\\Сириус:192.168.201.226"
         }
      ]
   }
}
```

GetEventsCount(...)

`TOperationResult<int> GetEventsCount(beginTime, endTime, eventTypes, persons, entryPoints, sections, sectionsGroups, token)` - получение количества событий системе, удовлетворяющим заданному фильтру.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
eventTypes	TeventType[]	Список типов событий для фильтрации
persons	TPersonData[]	Список сотрудников для фильтрации
entryPoints	TEntryPoint[]	Список точек доступа для фильтрации
sections	TSection[]	Список зон для фильтрации
sectionsGroups	TSectionsGroup[]	Список групп зон для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**). При указании данного параметра метод возвращает количество событий с учетом уровня доступа.

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество событий в системе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventsCount",
  "params": {
    "beginTime": "2025-01-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "eventTypes": []
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1879
   }
}
```

GetEventsFromId(...)

`TOperationResult<TEvent[]> GetEvents(id, eventTypes, persons, entryPoints, sections, sectionsGroups, token)` - получение событий, согласно переданному фильтру, начиная с указанного id.

Входные данные

Параметр	Тип	Описание
id	int	Начальный идентификатор события
eventTypes	TeventType[]	Список типов событий для фильтрации
persons	TPersonData[]	Список сотрудников для фильтрации
entryPoints	TEntryPoint[]	Список точек доступа для фильтрации
sections	TSection[]	Список зон для фильтрации
sectionGroups	TSectionsGroup[]	Список групп зон для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**). При указании данного параметра метод возвращает события с учетом уровня доступа.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEvent[]>**, содержащий список событий или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventsFromId",
  "params": {
    "id": 0,
    "eventTypes": [],
    "persons": [],
    "entryPoints": [],
    "sections": [],
    "sectionsGroups": []
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1879,
         "eventId": "{46764F00-6940-F011-B807-107C61D31197}",
         "eventTypeId": 310,
         "eventDate": "2025-06-03T13:53:41.000Z",
         "description": "",
```

```
            "computerId": 1,
            "comPortNumber": 0,
            "pKUAddress": 0,
            "devAddress": 0,
            "zoneAddress": 0,
            "accessPointId": 0,
            "accessPointName": "",
            "accessZoneId": 0,
            "passMode": 0,
            "cardNo": "",
            "personId": 0,
            "lastName": "",
            "firstName": "",
            "middleName": "",
            "birthDate": "",
            "tabNum": "",
            "itemId": 0,
            "itemType": "",
            "sectionId": 0,
            "readerId": -1,
            "accessFlags":          {
               "isBlocked": false,
               "isNoRights": false,
               "isBrokenAntipassback": false,
               "isBrokenTimeWindow": false,
               "isExpired": false,
               "isAdditionalCodeInputError": false,
               "isDuressCode": false,
               "isConfirmationCodeError": false,
               "isConfirmationWaiting": false
            },
            "strAddr": ""
         }]
      }
}
```


SubscribeEvents(...)

`TOperationResult<string> SubscribeEvents(id, eventTypes, persons, entryPoints, sections, sectionsGroups, token)` – подписка на получение событий.

Входные данные

Параметр	Тип	Описание
callbackUrl	string	URL-адрес обратного вызова для получения событий
jsonRpcMethodName	string	Название метода JSON-RPC для обратного вызова
eventTypes	TeventType[]	Список типов событий для фильтрации
persons	TPersonData[]	Список сотрудников для фильтрации
entryPoints	TEntryPoint[]	Список точек доступа для фильтрации
sections	TSection[]	Список зон для фильтрации
sectionGroups	TSectionsGroup[]	Список групп зон для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**). При указании данного параметра метод возвращает события с

		учетом уровня доступа.

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий токен подписки.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "subscribeEvents",
  "params": {
      "callBackUrl": "http://127.0.0.1:8096/jsonrpc",
      "jsonRpcMethodName": "test"
  },
  "id": 3
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 3,
   "result":    {
      "success": true,
      "operationResult": "{00943490-90B0-4472-ADCB-1A151D6091A1}"
   }
}
```

CheckEventSubscription(...)

`TOperationResult<string> CheckEventsSubscription(subscriberToken, token)` – проверка токена подписки на активность.

Входные данные

Параметр	Тип	Описание
subscriberToken	string	Токен подписки
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**). При указании данного параметра метод возвращает события с учетом уровня доступа.

Результирующие данные

Метод возвращает объект класса **TOperationResult<bool>**, содержащий информацию об активности подписки.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "checkEventSubscription",
  "params": {
      "subscriberToken": "{6B89BD43-F92F-45F8-B659-959EF2F4E48D}",
```

```
      "token": ""
  },
  "id": 3
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 3,
   "result":    {
      "success": true,
      "operationResult": true
   }
}
```

UnsubscribeEvents(...)

`TOperationResult<string> UnsubscribeEvents(subscriberToken, token)` – отписаться от получения событий.

Входные данные

Параметр	Тип	Описание
subscriberToken	string	Токен подписки
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**). При указании данного параметра метод возвращает события с учетом уровня доступа.

Результирующие данные

Метод возвращает объект класса TOperationResult, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "unsubscribeEvents",
  "params": {
      "subscriberToken": "{6B89BD43-F92F-45F8-B659-959EF2F4E48D}",
      "token": ""
  },
  "id": 3
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 3,
   "result": {"success": true}
}
```

GetAccessLevels(...)

`TOperationResult<TAccessLevel[]> GetAccessLevels(offset, count, token)` - получение списка уровней доступа.

Входные данные

offset	int	Смещение относительно начала полного списка уровней доступа
count	int	Количество возвращаемых уровней доступа начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAccessLevel[]>**, содержащий список уровней доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAccessLevels",
  "params": {
    "offset": 0,
    "count": 1
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "Запрет",
         "description": "Запрещается проход через двери и управление зонами",
         "items": []
      }]
   }
}
```

GetAccessLevelsCount(...)

`TOperationResult<int> GetAccessLevelsCount(token)` - получение количества уровней доступа в системе.

Входные данные

Параметр	Тип	Описание

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество уровней доступа в системе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAccessLevelsCount",
  "params": {
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 2
   }
}
```

GetAccessLevelById(...)

`TOperationResult<TAccessLevel> GetAccessLevelById(id, token)` – поиск уровня доступа по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Уникальный идентификатор уровня доступа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAccessLevel>**, содержащий данные найденного уровня доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAccessLevelById",
  "params": {
    "id": 1
```

```
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "Запрет",
         "description": "Запрещается проход через двери и управление зонами",
         "items": []
      }
   }
}
```

GetAccessZones(...)

`TOperationResult<TAccessZone[]> GetAccessZones(token)` - получение списка зон доступа.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAccessZone[]>**, содержащий список зон доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAccessZones",
  "params": {
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "Внешний мир"
      }]
   }
```

```
}
```

GetEntryPoints(...)

`TOperationResult<TEntryPoint[]> GetEntryPoints(offset, count, token)` - получение списка точек доступа.

Входные данные

offset	int	Смещение относительно начала полного списка точек доступа
count	int	Количество возвращаемых точек доступа начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEntryPoint[]>**, содержащий список точек доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEntryPoints",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "Точка доступа  1",
         "enterAccessZoneId": 0,
         "exitAccessZoneId": 0,
         "entryPointType": 1,
         "readers": [1213],
         "index": 1
      }]
   }
}
```

GetEntryPointsCount(...)

`TOperationResult<int> GetEntryPointsCount(token)` - получение количества точек доступа в системе.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество точек доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEntryPointsCount",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
   }
}
```

GetEntryPointById(...)

`TOperationResult<TEntryPoint> GetEntryPointById(id, token)` - поиск точки доступа по ее идентификатору.

Входные данные

id	int	Уникальный идентификатор точки доступа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TEntryPoint>**, содержащий данные найденной точки доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEntryPointById",
```

```
  "params": {
    "id": 1
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "Точка доступа  1",
         "enterAccessZoneId": 0,
         "exitAccessZoneId": 0,
         "entryPointType": 1,
         "readers": [1213],
         "index": 1
      }
   }
}
```

PutPassWithAccLevels(...)

`TOperationResult<TPersonData> PutPassWithAccLevels(cardNo, personData, accessLevels, dateBegin, dateEnd, token, codeType)` – программирование сотрудника и ключа с использованием уровня доступа АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа
personData	TPersonData	Информация о сотруднике
accessLevels	TAccessLevel[]	Уровень доступа АРМ «Орион Про»
dateBegin	DateTime	Дата начала действия ключа
dateEnd	DateTime	Дата окончания действия ключа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)
codeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото;

		18 — QR-код; 19 — Шаблон лица SF6P.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий информацию о добавленном сотруднике или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "putPassWithAccLevels",
  "params": {
    "cardNo": "690000002C223801",
    "codeType": 4,
    "personData": {
      "lastName": "Сидоров",
      "firstName": "Сергей",
      "middleName": "Сергеевич",
      "company": "ООО ТКФ",
      "department": "Бухгалтерия",
      "position": "Бухгалтер",
      "tabNum": "443",
      "sex": 1,
      "status": 6,
      "accessLevelId": 0
    },
    "accessLevels": [
      {
        "id": 2
      }
    ],
    "dateBegin": "2018-09-14T00:00:00Z",
    "dateEnd": "2118-01-01T23:59:59Z"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 5,
         "lastName": "Сидоров",
         "firstName": "Сергей",
         "middleName": "Сергеевич",
         "birthDate": "",
         "company": "ООО ТКФ",
         "department": "Бухгалтерия",
         "position": "Бухгалтер",
         "companyId": 0,
         "departmentId": 0,
         "positionId": 0,
```

```
            "tabNum": "443",
            "phone": "",
            "homePhone": "",
            "address": "",
            "photo": "",
            "accessLevelId": 0,
            "status": 6,
            "contactIdIndex": 0,
            "isLockedDayCrossing": false,
            "isFreeShedule": false,
            "externalId": "",
            "additionalExternalId": "",
            "isInArchive": false,
            "documentType": 0,
            "documentSerials": "",
            "documentNumber": "",
            "documentIssueDate": "",
            "documentEndingDate": "",
            "documentIsser": "",
            "documentIsserCode": "",
            "sex": 0,
            "birthplace": "",
            "emailList": "",
            "archivingTimeStamp": "",
            "isInBlackList": false,
            "isDismissed": false,
            "blackListComment": "",
            "changeTime": "",
            "itn": "",
            "dismissedComment": "",
            "roomId": 0,
            "exitGroupId": 0,
            "exitGroup": ""
        }
    }
}
```


PutPassWithEntryPoints(...)

`TOperationResult<TPersonData> PutPassWithEntryPoints(cardNo, personData, entryPoints, accessZones, dateBegin, dateEnd, timeBegin, timeEnd, accessSaturday, accessSunday, accessHoliday, token)` – программирование сотрудника и ключа с использованием зон доступа и точек доступа АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа
personData	TPersonData	Информация о сотруднике
entryPoints	TAccessPoint[]	Точки доступа АРМ «Орион Про»
accessZones	TAccessZone[]	Зоны доступа АРМ «Орион Про»
dateBegin	DateTime	Дата начала действия ключа
dateEnd	DateTime	Дата окончания действия ключа
timeBegin	DateTime	Время начала действия временного интервала создаваемого окна времени

timeEnd	DateTime	Время окончания действия временного интервала создаваемого окна времени
accessSaturday	int	Признак активности временного интервала в субботу
accessSunday	int	Признак активности выходного интервала в воскресенье
accessHoliday	int	Признак активности временного интервала в праздничный день
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий информацию о добавленном сотруднике или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "putPassWithEntryPoints",
  "params": {
    "cardNo": "5500000FBA963F01",
    "personData": {
      "lastName": "Сидоров",
      "firstName": "Сергей",
      "middleName": "Сергеевич",
      "company": "ООО ТКФ",
      "department": "Бухгалтерия",
      "position": "Бухгалтер",
      "tabNum": "42043341",
      "sex": 1,
      "status": 6,
      "accessLevelId": 0
    },
    "entryPoints": [
      {
        "id": 1
      }
    ],
    "dateBegin": "2018-09-14T00:00:00Z",
    "dateEnd": "2118-01-01T23:59:59Z",
    "timeBegin": "2018-09-14T00:00:00Z",
    "timeEnd": "2018-09-14T00:00:00Z",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
```

```
    "operationResult":       {
       "id": 6,
       "lastName": "Сидоров",
       "firstName": "Сергей",
       "middleName": "Сергеевич",
       "birthDate": "",
       "company": "ООО ТКФ",
       "department": "Бухгалтерия",
       "position": "Бухгалтер",
       "companyId": 0,
       "departmentId": 0,
       "positionId": 0,
       "tabNum": "42043341",
       "phone": "",
       "homePhone": "",
       "address": "",
       "photo": "",
       "accessLevelId": 0,
       "status": 6,
       "contactIdIndex": 0,
       "isLockedDayCrossing": false,
       "isFreeShedule": false,
       "externalId": "",
       "additionalExternalId": "",
       "isInArchive": false,
       "documentType": 0,
       "documentSerials": "",
       "documentNumber": "",
       "documentIssueDate": "",
       "documentEndingDate": "",
       "documentIsser": "",
       "documentIsserCode": "",
       "sex": 0,
       "birthplace": "",
       "emailList": "",
       "archivingTimeStamp": "",
       "isInBlackList": false,
       "isDismissed": false,
       "blackListComment": "",
       "changeTime": "",
       "itn": "",
       "dismissedComment": "",
       "roomId": 0,
       "exitGroupId": 0,
       "exitGroup": ""
    }
}
```


BlockPass(...)

`TOperationResult<string> BlockPass(cardNo, blocked, token)` – блокировка/разблокировка ключа.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа
blocked	bool	Признак блокировки ключа

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий код заблокированного/разблокированного ключа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "blockPass",
  "params": {
    "cardNo": "5500000FBA963F01",
    "blocked": true
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "5500000FBA963F01"
   }
}
```

BlockPerson(...)

`TOperationResult<string[]> BlockPerson(personData, blocked, token)` – блокировка/разблокировка всех ключей, связанных с данным сотрудником.

Входные данные

Параметр	Тип	Описание
personData	TPersonData	Информация о сотруднике
blocked	bool	Признак блокировки ключей
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string[]>**, содержащий список кодов заблокированных/разблокированного ключей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
```

```
  "method": "blockPerson",
  "params": {
    "personData": {
      "id": 6
    },
    "blocked": true
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["5500000FBA963F01"]
   }
}
```

BlockPersonByPass(...)

`TOperationResult<string[]> BlockPersonByPass(cardNo, blocked, codeType, token)` – блокировка/разблокировка всех ключей, связанных с данным сотрудником. Поиск сотрудника осуществляется по одному из принадлежащих ему ключей.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа, принадлежащий сотруднику, которого необходимо заблокировать
blocked	bool	Признак блокировки ключей
codeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string[]>**, содержащий список кодов

заблокированных/разблокированного ключей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "blockPersonByPass",
  "params": {
    "cardNo": "5500000FBA963F01",
    "blocked": true,
    "codeType": 0
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["5500000FBA963F01"]
   }
}
```

DeletePass(...)

`TOperationResult<string> DeletePass(cardNo, codeType, token)` – удаление ключа.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа
CodeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий код удаленного ключа или

информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deletePass",
  "params": {
    "cardNo": "5500000FBA963F01"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "5500000FBA963F01"
   }
}
```

GetPersons(...)

`TOperationResult<TPersonData[]> GetPersons(withoutPhoto, offset, count, filter, isEmployees, isVisitors, token)` - получение списка сотрудников.

Входные данные

Параметр	Тип	Описание
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
offset	int	Смещение относительно начала полного списка сотрудников. Значение по умолчанию: 0.
count	int	Количество возвращаемых сотрудников начиная со смещения offset. При значениях данного параметра и параметра offset равным 0, возвращается полный список сотрудников. Значение по умолчанию: 0.
filter	string[]	Фильтр. Строки массива имеют следующий формат: “fieldName=fieldValue”.
isEmployees	bool	Если данный параметр имеет значение true, то в списке будут только сотрудники, к которым идет посетитель.
isVisitors	bool	Если данный параметр имеет значение true, то в списке будут только посетители.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData[]>**, содержащий список сотрудников или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersons",
  "params": {
    "withoutPhoto": true
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "lastName": "Иванов",
            "firstName": "Иван",
            "middleName": "Иванович",
            "birthDate": "",
            "company": "",
            "department": "",
            "position": "",
            "companyId": 0,
            "departmentId": 0,
            "positionId": 0,
            "tabNum": "1",
            "phone": "",
            "homePhone": "",
            "address": "",
            "photo": "",
            "accessLevelId": 0,
            "status": 1,
            "contactIdIndex": 0,
            "isLockedDayCrossing": false,
            "isFreeShedule": false,
            "externalId": "",
            "additionalExternalId": "",
            "isInArchive": false,
            "documentType": 0,
            "documentSerials": "",
            "documentNumber": "",
            "documentIssueDate": "",
            "documentEndingDate": "",
            "documentIsser": "",
            "documentIsserCode": "",
            "sex": 0,
            "birthplace": "",
            "emailList": "",
            "archivingTimeStamp": "2025-06-17T14:14:27.000Z",
```

```
            "isInBlackList": false,
            "isDismissed": false,
            "blackListComment": "",
            "changeTime": "2025-06-17T14:14:27.000Z",
            "itn": "",
            "dismissedComment": "",
            "roomId": 0,
            "exitGroupId": 0,
            "exitGroup": ""
        }
    ]
  }
}
```


GetPersonsCount(...)

`TOperationResult<int> GetPersonsCount(filter, isEmployees, isVisitors, token)`- получение количества сотрудников в системе.

Входные данные

Параметр	Тип	Описание
filter	string[]	Фильтр. Строки массива имеют следующий формат: “fieldName=fieldValue”.
isEmployees	bool	Если данный параметр имеет значение true, то в списке будут только сотрудники, к которым идет посетитель.
isVisitors	bool	Если данный параметр имеет значение true, то в списке будут только посетители.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество сотрудников в системе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonsCount",
  "params": {
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
```

```
    }
}
```

SearchPersons(...)

`TOperationResult<TPersonData[]> SearchPersons(filters, withoutPhoto, token)`- поиск сотрудников.

Входные данные

Параметр	Тип	Описание
filters	string[][]	Набор групп значений в формате “fieldName=fieldValue” для поиска. Каждая группа значений описывает одного сотрудника. Пример описания параметра в формате xml: *<Filters>* *<Person>* *<Item>LastName=Иванов</Item>* *<Item>FirstName=Иван</Item>* *</Person>* *<Person>* *<Item>LastName=Петров</Item>* *<Item>FirstName=Петр</Item>* *</Person>* *</Filters>*
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData[]>**, содержащий список сотрудников или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "searchPersons",
  "params": {
    "filters": [
            ["LastName=Иванов" , "FirstName=Иван"],
            [ "LastName=Петров" , "FirstName=Петр" ]
    ]
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
```

```
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "lastName": "Иванов",
         "firstName": "Иван",
         "middleName": "Иванович",
         "birthDate": "",
         "company": "",
         "department": "",
         "position": "",
         "companyId": 0,
         "departmentId": 0,
         "positionId": 0,
         "tabNum": "1",
         "phone": "",
         "homePhone": "",
         "address": "",
         "photo": "",
         "accessLevelId": 0,
         "status": 1,
         "contactIdIndex": 0,
         "isLockedDayCrossing": false,
         "isFreeShedule": false,
         "externalId": "",
         "additionalExternalId": "",
         "isInArchive": false,
         "documentType": 0,
         "documentSerials": "",
         "documentNumber": "",
         "documentIssueDate": "",
         "documentEndingDate": "",
         "documentIsser": "",
         "documentIsserCode": "",
         "sex": 0,
         "birthplace": "",
         "emailList": "",
         "archivingTimeStamp": "2025-06-17T14:14:27.000Z",
         "isInBlackList": false,
         "isDismissed": false,
         "blackListComment": "",
         "changeTime": "2025-06-17T14:14:27.000Z",
         "itn": "",
         "dismissedComment": "",
         "roomId": 0,
         "exitGroupId": 0,
         "exitGroup": ""
      }]
   }
}
```


AddPerson(...)

`TOperationResult<TPersonData> AddPerson(personData, token)` – добавление сотрудника.

Входные данные

Параметр	Тип	Описание
personData	TPersonData	Информация о сотруднике

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TPersonData>, содержащий информацию о добавленном сотруднике или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "addperson",
  "params": {
    "persondata": {
      "id": 0,
      "lastname": "Сидоров",
      "firstname": "Сергей",
      "middlename": "Сергеевич",
      "birthdate": "1979-05-20T00:00:00",
      "companyid": 2,
      "departmentid": 109,
      "positionid": 85,
      "tabnum": "34543576rrf",
      "phone": "",
      "homephone": "",
      "address": "",
      "photo": ""
      "accesslevelid": 0,
      "status": 6,
      "contactidindex": 0,
      "islockeddaycrossing": false,
      "isfreeshedule": false,
      "externalid": "5389e367-3db3-11ea-80c9-005056aab926",
      "isinarchive": false,
      "documenttype": 0,
      "documentserials": "",
      "documentnumber": "",
      "documentissuedate": "",
      "documentendingdate": "1899-12-30T00:00:00",
      "documentisser": "",
      "documentissercode": "",
      "sex": 2,
      "birthplace": "",
      "emaillist": "",
      "archivingtimestamp": "2021-02-09T00:00:00",
      "isinblacklist": false,
      "isdismissed": false,
      "blacklistcomment": "",
      "changetime": "2025-02-05T16:57:10",
      "itn": "",
      "dismissedcomment": "",
      "roomid": 0
    },
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 13,
         "lastName": "Сидоров",
         "firstName": "Сергей",
         "middleName": "Сергеевич",
         "birthDate": "1979-05-20T00:00:00.000Z",
         "company": "",
         "department": "",
         "position": "",
         "companyId": 2,
         "departmentId": 109,
         "positionId": 85,
         "tabNum": "34543576rrf",
         "phone": "",
         "homePhone": "",
         "address": "",
         "photo": "",
         "accessLevelId": 0,
         "status": 6,
         "contactIdIndex": 0,
         "isLockedDayCrossing": false,
         "isFreeShedule": false,
         "externalId": "5389e367-3db3-11ea-80c9-005056aab926",
         "additionalExternalId": "",
         "isInArchive": false,
         "documentType": 0,
         "documentSerials": "",
         "documentNumber": "",
         "documentIssueDate": "",
         "documentEndingDate": "",
         "documentIsser": "",
         "documentIsserCode": "",
         "sex": 2,
         "birthplace": "",
         "emailList": "",
         "archivingTimeStamp": "2025-06-25T15:54:21.000Z",
         "isInBlackList": false,
         "isDismissed": false,
         "blackListComment": "",
         "changeTime": "2025-06-25T15:54:21.000Z",
         "itn": "",
         "dismissedComment": "",
         "roomId": 0,
         "exitGroupId": 0,
         "exitGroup": ""
      }
   }
}
```


UpdatePerson(...)

`TOperationResult<TPersonData> UpdatePerson(personData,` allowClearPhoto, `token)` – обновление сотрудника.

Входные данные

Параметр	Тип	Описание
personData	TPersonData	Информация о сотруднике
allowClearPhoto	bool	false - если в поле Photo структуры personData отсутствуют данные, фотография сотрудника не изменяется; true - разрешить изменение.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TPersonData>, содержащий информацию об обновленном сотруднике или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "updatePerson",
  "params": {
    "personData": {
      "id": 8,
      "lastName": "Васильев",
      "firstName": "Василий",
      "middleName": "Васильевич",
      "birthDate": "1899-12-30T00:00:00.000+03:00",
      "company": "",
      "department": "",
      "position": "",
      "companyId": 4,
      "departmentId": 2,
      "positionId": 3,
      "tabNum": "54353453",
      "phone": "",
      "homePhone": "",
      "address": "",
      "photo": null,
      "accessLevelId": 0,
      "status": 1,
      "contactIdIndex": 0,
      "isLockedDayCrossing": false,
      "isFreeShedule": false,
      "externalId": "",
      "isInArchive": false,
      "documentType": 0,
      "documentSerials": "",
      "documentNumber": "",
      "documentIssueDate": "1899-12-30T00:00:00.000+03:00",
      "documentEndingDate": "1899-12-30T00:00:00.000+03:00",
      "documentIsser": "",
      "documentIsserCode": "",
      "sex": 0,
      "birthplace": "",
      "emailList": "",
      "archivingTimeStamp": "1899-12-30T00:00:00.000+03:00",
```

```
      "isInBlackList": false,
      "isDismissed": false,
      "blackListComment": "",
      "changeTime": "2020-09-16T12:25:13.000+03:00",
      "itn": "",
      "dismissedComment": ""
    },
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 8,
         "lastName": "Васильев",
         "firstName": "Василий",
         "middleName": "Васильевич",
         "birthDate": "1899-12-30T03:00:00.000Z",
         "company": "",
         "department": "",
         "position": "",
         "companyId": 4,
         "departmentId": 2,
         "positionId": 3,
         "tabNum": "54353453",
         "phone": "",
         "homePhone": "",
         "address": "",
         "photo": "",
         "accessLevelId": 0,
         "status": 1,
         "contactIdIndex": 0,
         "isLockedDayCrossing": false,
         "isFreeShedule": false,
         "externalId": "",
         "additionalExternalId": "",
         "isInArchive": false,
         "documentType": 0,
         "documentSerials": "",
         "documentNumber": "",
         "documentIssueDate": "1899-12-30T03:00:00.000Z",
         "documentEndingDate": "1899-12-30T03:00:00.000Z",
         "documentIsser": "",
         "documentIsserCode": "",
         "sex": 0,
         "birthplace": "",
         "emailList": "",
         "archivingTimeStamp": "1899-12-30T03:00:00.000Z",
         "isInBlackList": false,
         "isDismissed": false,
         "blackListComment": "",
         "changeTime": "2020-09-16T09:25:13.000Z",
         "itn": "",
         "dismissedComment": "",
         "roomId": 0,
```

```
            "exitGroupId": 0,
            "exitGroup": ""
        }
    }
}
```

GetPersonPassList(...)

`TOperationResult<string[]> GetPersonPassList(personData, token)` – получение списка кодов ключей сотрудника.

Входные данные

Параметр	Тип	Описание
personData	TPersonData	Информация о сотруднике, список ключей которого необходимо получить
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Метод возвращает объект класса TOperationResult<string[]>, содержащий список кодов ключей сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonPassList",
  "params": {
    "personData": {
      "id": 2
    }
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["F4FFFFFFF77777F0"]
   }
}
```

GetPersonPassListByPass(...)

`TOperationResult<string[]> GetPersonPassList(cardNo, codeType, token)` – получение списка кодов ключей сотрудника. Поиск сотрудника производится по одному из принадлежащих ему ключей.

Входные данные

Параметр	Тип	Описание

cardNo	string	Код ключа, принадлежащего сотруднику
codeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<string[]>, содержащий список кодов ключей сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonPassListByPass",
  "params": {
    "cardNo": "F4FFFFFFF77777F0",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["F4FFFFFFF77777F0"]
   }
}
```

GetPersonById(...)

`TOperationResult<TPersonData> GetPersonById(id, withoutPhoto, token)` – поиск сотрудника по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Уникальный идентификатор сотрудника
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий данные найденного сотрудника.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonById",
  "params": {
    "id": 1,
    "withoutPhoto": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "lastName": "Иванов",
         "firstName": "Иван",
         "middleName": "Иванович",
         "birthDate": "",
         "company": "",
         "department": "",
         "position": "",
         "companyId": 0,
         "departmentId": 0,
         "positionId": 0,
         "tabNum": "1",
         "phone": "",
         "homePhone": "",
         "address": "",
         "photo": "",
         "accessLevelId": 0,
         "status": 1,
         "contactIdIndex": 0,
         "isLockedDayCrossing": false,
         "isFreeShedule": false,
         "externalId": "",
         "additionalExternalId": "",
```

```
            "isInArchive": false,
            "documentType": 0,
            "documentSerials": "",
            "documentNumber": "",
            "documentIssueDate": "",
            "documentEndingDate": "",
            "documentIsser": "",
            "documentIsserCode": "",
            "sex": 0,
            "birthplace": "",
            "emailList": "",
            "archivingTimeStamp": "2025-06-17T14:14:27.000Z",
            "isInBlackList": false,
            "isDismissed": false,
            "blackListComment": "",
            "changeTime": "2025-06-17T14:14:27.000Z",
            "itn": "",
            "dismissedComment": "",
            "roomId": 0,
            "exitGroupId": 0,
            "exitGroup": ""
        }
    }
}
```


GetPersonByPass(...)

`TOperationResult<TPersonData> GetPersonByPass(cardNo, withoutPhoto, codeType, token)` – поиск сотрудника по коду ключа. Поиск сотрудника производится по одному из принадлежащих ему ключей.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа, принадлежащего сотруднику
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
codeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий данные найденного сотрудника.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonByPass",
  "params": {
    "cardNo": "F4FFFFFFF77777F0",
    "withoutPhoto": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "lastName": "user",
         "firstName": "-",
         "middleName": "-",
         "birthDate": "",
         "company": "",
         "department": "",
         "position": "",
         "companyId": 0,
         "departmentId": 0,
         "positionId": 0,
         "tabNum": "",
         "phone": "",
         "homePhone": "",
         "address": "",
         "photo": "",
         "accessLevelId": 0,
         "status": 1,
         "contactIdIndex": 0,
         "isLockedDayCrossing": false,
         "isFreeShedule": false,
         "externalId": "",
         "additionalExternalId": "",
         "isInArchive": false,
         "documentType": 0,
         "documentSerials": "",
         "documentNumber": "",
         "documentIssueDate": "",
         "documentEndingDate": "",
         "documentIsser": "",
         "documentIsserCode": "",
         "sex": 0,
         "birthplace": "",
```

```
            "emailList": "",
            "archivingTimeStamp": "2025-06-19T10:19:27.000Z",
            "isInBlackList": false,
            "isDismissed": false,
            "blackListComment": "",
            "changeTime": "2025-06-19T10:19:27.000Z",
            "itn": "",
            "dismissedComment": "",
            "roomId": 0,
            "exitGroupId": 0,
            "exitGroup": ""
        }
    }
}
```


GetPersonByExternalId(...)

`TOperationResult<TPersonData> GetPersonByExternalId(externalId, withoutPhoto, token)` – поиск сотрудника по идентификатору, присвоенному сторонней системой.

Входные данные

Параметр	Тип	Описание
externalId	string	Идентификатор сотрудника, присвоенный сторонней системой
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий данные найденного сотрудника.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonByExternalId",
  "params": {
    "externalId": "1111",
    "withoutPhoto": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
  "result":   {
    "success": true,
    "operationResult":      {
      "id": 8,
      "lastName": "Васильев",
      "firstName": "Василий",
      "middleName": "Васильевич",
      "birthDate": "",
      "company": "",
      "department": "",
      "position": "",
      "companyId": 4,
      "departmentId": 2,
      "positionId": 3,
      "tabNum": "54353453",
      "phone": "",
      "homePhone": "",
      "address": "",
      "photo": "",
      "accessLevelId": 0,
      "status": 1,
      "contactIdIndex": 0,
      "isLockedDayCrossing": false,
      "isFreeShedule": false,
      "externalId": "1111",
      "additionalExternalId": "1111",
      "isInArchive": false,
      "documentType": 0,
      "documentSerials": "",
      "documentNumber": "",
      "documentIssueDate": "1899-12-30T03:00:00.000Z",
      "documentEndingDate": "1899-12-30T03:00:00.000Z",
      "documentIsser": "",
      "documentIsserCode": "",
      "sex": 0,
      "birthplace": "",
      "emailList": "",
      "archivingTimeStamp": "2025-06-25T16:34:27.000Z",
      "isInBlackList": false,
      "isDismissed": false,
      "blackListComment": "",
      "changeTime": "2025-06-25T16:34:27.000Z",
      "itn": "",
      "dismissedComment": "",
      "roomId": 0,
      "exitGroupId": 0,
      "exitGroup": ""
    }
  }
}
```


GetPersonByAdditionalExternalId(...)

`TOperationResult<TPersonData> GetPersonByAdditionalExternalId(additionalExternalId, withoutPhoto, token)` – поиск сотрудника по дополнительному идентификатору, присвоенному сторонней системой.

Входные данные

Параметр	Тип	Описание
additionalExternalId	string	Дополнительный идентификатор сотрудника, присвоенный сторонней системой
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий данные найденного сотрудника.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonByAdditionalExternalId",
  "params": {
    "additionalExternalId": "1111",
    "withoutPhoto": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result":   {
    "success": true,
    "operationResult":      {
      "id": 8,
      "lastName": "Васильев",
      "firstName": "Василий",
      "middleName": "Васильевич",
      "birthDate": "",
      "company": "",
      "department": "",
      "position": "",
      "companyId": 4,
      "departmentId": 2,
      "positionId": 3,
      "tabNum": "54353453",
      "phone": "",
      "homePhone": "",
      "address": "",
      "photo": "",
      "accessLevelId": 0,
      "status": 1,
      "contactIdIndex": 0,
      "isLockedDayCrossing": false,
```

"isFreeShedule": false,
"externalId": "1111",
"additionalExternalId": "1111",
"isInArchive": false,
"documentType": 0,
"documentSerials": "",
"documentNumber": "",
"documentIssueDate": "1899-12-30T03:00:00.000Z",
"documentEndingDate": "1899-12-30T03:00:00.000Z",
"documentIsser": "",
"documentIsserCode": "",
"sex": 0,
"birthplace": "",
"emailList": "",
"archivingTimeStamp": "2025-06-25T16:34:27.000Z",
"isInBlackList": false,
"isDismissed": false,
"blackListComment": "",
"changeTime": "2025-06-25T16:34:27.000Z",
"itn": "",
"dismissedComment": "",
"roomId": 0,
"exitGroupId": 0,
"exitGroup": ""
}
}
}

GetPersonByTabNumber(...)

`TOperationResult<TPersonData> GetPersonByTabNumber(tabNum, withoutPhoto, token)` – поиск сотрудника по его табельному номеру.

Входные данные

Параметр	Тип	Описание
tabNum	int	Уникальный идентификатор сотрудника
withoutPhoto	bool	Если значение данного параметра true, то информация о сотрудниках возвращается без фото. Значение по умолчанию: false.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData>**, содержащий данные найденного сотрудника.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonByTabNumber",
  "params": {
    "tabNum": "54353453",
```

```
    "withoutPhoto": true,
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 8,
         "lastName": "Васильев",
         "firstName": "Василий",
         "middleName": "Васильевич",
         "birthDate": "",
         "company": "",
         "department": "Постоялец",
         "position": "",
         "companyId": 4,
         "departmentId": 2,
         "positionId": 3,
         "tabNum": "54353453",
         "phone": "",
         "homePhone": "",
         "address": "",
         "photo": "",
         "accessLevelId": 0,
         "status": 1,
         "contactIdIndex": 0,
         "isLockedDayCrossing": false,
         "isFreeShedule": false,
         "externalId": "1111",
         "additionalExternalId": "",
         "isInArchive": false,
         "documentType": 0,
         "documentSerials": "",
         "documentNumber": "",
         "documentIssueDate": "1899-12-30T03:00:00.000Z",
         "documentEndingDate": "1899-12-30T03:00:00.000Z",
         "documentIsser": "",
         "documentIsserCode": "",
         "sex": 0,
         "birthplace": "",
         "emailList": "",
         "archivingTimeStamp": "2025-06-25T16:34:27.000Z",
         "isInBlackList": false,
         "isDismissed": false,
         "blackListComment": "",
         "changeTime": "2025-06-25T16:34:27.000Z",
         "itn": "",
         "dismissedComment": "",
         "roomId": 0,
         "exitGroupId": 0,
         "exitGroup": ""
      }
   }
}
```

GetPersonsWithEmptyPassList(...)

`TOperationResult<TPersonData[]> GetPersonsWithEmptyPassList(token)` - получение списка сотрудников с пустым списком ключей.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData[]>**, содержащий список сотрудников или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonsWithEmptyPassList",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 12,
            "lastName": "",
            "firstName": "",
            "middleName": "",
            "birthDate": "",
            "company": "",
            "department": "",
            "position": "",
            "companyId": 0,
            "departmentId": 0,
            "positionId": 0,
            "tabNum": "",
            "phone": "",
            "homePhone": "",
            "address": "",
            "photo": "",
            "accessLevelId": 0,
            "status": 0,
            "contactIdIndex": 0,
            "isLockedDayCrossing": false,
```

```
            "isFreeShedule": false,
            "externalId": "",
            "additionalExternalId": "",
            "isInArchive": false,
            "documentType": 0,
            "documentSerials": "",
            "documentNumber": "",
            "documentIssueDate": "",
            "documentEndingDate": "",
            "documentIsser": "",
            "documentIsserCode": "",
            "sex": 0,
            "birthplace": "",
            "emailList": "",
            "archivingTimeStamp": "",
            "isInBlackList": false,
            "isDismissed": false,
            "blackListComment": "",
            "changeTime": "",
            "itn": "",
            "dismissedComment": "",
            "roomId": 0,
            "exitGroupId": 0,
            "exitGroup": ""
         }
      ]
   }
}
```


GetPersonsForEntryPoint(...)

`TOperationResult<TPersonData[]> GetPersonsForEntryPoint(entryPoint, token)`- получение списка сотрудников для точки доступа.

Входные данные

Параметр	Тип	Описание
entryPoint	TEntryPoint	Точка доступа, для которой запрашивается список сотрудников
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData[]>**, содержащий список сотрудников или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "getPersonsForEntryPoint",
  "params": {
    "entryPoint": {
      "id": 1,
      "name": "",
      "enterAccessZoneId": 0,
```

```
      "exitAccessZoneId": 0,
      "entryPointType": 0,
      "readers": [],
      "index": 0
    },
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 6,
            "lastName": "Сидоров",
            "firstName": "Сергей",
            "middleName": "Сергеевич",
            "birthDate": "",
            "company": "",
            "department": "",
            "position": "",
            "companyId": 0,
            "departmentId": 0,
            "positionId": 0,
            "tabNum": "42043341",
            "phone": "",
            "homePhone": "",
            "address": "",
            "photo": "",
            "accessLevelId": 0,
            "status": 6,
            "contactIdIndex": 0,
            "isLockedDayCrossing": false,
            "isFreeShedule": false,
            "externalId": "",
            "additionalExternalId": "",
            "isInArchive": false,
            "documentType": 0,
            "documentSerials": "",
            "documentNumber": "",
            "documentIssueDate": "",
            "documentEndingDate": "",
            "documentIsser": "",
            "documentIsserCode": "",
            "sex": 0,
            "birthplace": "",
            "emailList": "",
            "archivingTimeStamp": "",
            "isInBlackList": false,
            "isDismissed": false,
            "blackListComment": "",
            "changeTime": "",
            "itn": "",
            "dismissedComment": "",
            "roomId": 0,
```

```
            "exitGroupId": 0,
            "exitGroup": ""
        }
    ]
  }
}
```

DeletePerson(...)

`TOperationResult<string[]> DeletePerson(personData, token)` – удаление сотрудника и всех связанных с ним ключей.

Входные данные

Параметр	Тип	Описание
personData	TPersonData	Информация о сотруднике
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string[]>**, содержащий список удаленных ключей сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getEventsCount",
  "params": {
    "beginTime": "2025-01-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "eventTypes": []
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": []
   }
}
```

DeletePersonByPass(...)

`TOperationResult<string[]> DeletePersonByPass(cardNo, codeType, token)` – удаление сотрудника и всех связанных с ним ключей. Поиск удаляемого сотрудника производится по одному из принадлежащих ему ключей.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа, принадлежащего удаляемому сотруднику
codeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string[]>**, содержащий список удаленных ключей сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deletePersonByPass",
  "params": {
    "cardNo": "55000000BA963F01",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["55000000BA963F01"]
   }
}
```

StopPerson(...)

`TOperationResult<string[]> StopPerson(personData, token)` – добавление/удаление сотрудника в/из стоп-листа.

Входные данные

Параметр	Тип	Описание
personData	TPersonData	Информация о сотруднике
stop	bool	true – добавить в стоп-лист; false – удалить из стоп-листа;
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string[]>**, содержащий список кодов ключей, добавленных/извлеченных в/из стоп-листа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "stopPerson",
  "params": {
    "personData": {
      "id": 1
    },
    "stop": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["1"]
   }
}
```

StopPersonByPass(...)

`TOperationResult<string[]> StopPersonByPass(cardNo, codeType, token)` – добавление/удаление сотрудника в/из стоп-листа. Поиск сотрудника производится по одному из принадлежащих ему ключей.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа, принадлежащий сотруднику, которого необходимо добавить в стоп-лист
codeType	int	Тип кода ключа: 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory;

		4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<string[]>**, содержащий список кодов ключей, добавленных/извлеченных в/из стоп-листа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "stopPersonByPass",
  "params": {
    "cardNo": "5500000FBA963F01",
    "stop": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": ["5500000FBA963F01"]
   }
}
```

GetVisits(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit[]> GetVisits(token)` – получение списка посетителей из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit[]>**, содержащий список посетителей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getVisits",
  "params": {},
  "id": 1
}
```

GetVisitById(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit> GetVisitById(id, token)` – получение информации о посетителе по его идентификатору из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
Id	int	Идентификатор посетителя
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit>**, содержащий информацию о посетителе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getVisitById",
  "params": {
    "id": 1
  },
  "id": 1
}
```

GetVisitByImpersonalPersonId(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit> GetVisitByImpersonalPersonId(id, token)` – получение информации о посетителе по идентификатору связанного с ним обезличенного сотрудника (шаблона посетителя) из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
Id	int	Идентификатор обезличенного сотрудника (шаблона посетителя)
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit>**, содержащий информацию о посетителе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getVisitByImpersonalPersonId",
  "params": {
    "id": 0,
    "token": ""
  },
  "id": 1
}
```

GetVisitByPersonId(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit> GetVisitByPersonId(id, token)` – получение информации о посетителе по идентификатору сотрудника из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
Id	int	Идентификатор сотрудника, являющегося посетителем
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit>**, содержащий информацию о посетителе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getVisitByPersonId",
  "params": {
    "id": 0,
    "token": ""
  },
  "id": 1
}
```

AddVisit(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit> AddVisit(visit, token)` – добавление посетителя.

Входные данные

Параметр	Тип	Описание
visit	TVisit	Информация о добавляемом посетителе
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit>**, содержащий информацию о добавленном посетителе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addVisit",
  "params": {
    "visit": {
      "id": 123,
      "personId": 5,
      "visitedPersonId": 7,
      "visitedCompanyId": 101,
      "visitedDepartmentId": 202,
      "visitDate": "2025-04-01T10:00:00.000+03:00",
      "visitedRoom": "Conference Room 1",
      "visitedPersonPhone": "+79251233333",
      "visitPurpose": "Business meeting",
      "impersonalPersonId": 2,
      "accessControlRule": 2,
      "visitEndDateTime": "2025-04-01T12:00:00.000+03:00",
      "instructingPersonId": 404,
      "carName": "Toyota Camry",
      "carNumber": "ABC-123",
      "carColor": "Black",
      "isCarOvernight": false
    }
  },
  "id": 1
}
```

DeleteVisit(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit> DeleteVisit(visit, token)` – удаление посетителя.

Входные данные

Параметр	Тип	Описание
visit	TVisit	Информация об удаляемом посетителе. Поиск и удаление посетителя производится по значению поля “id” структуры TVisit.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit>**, содержащий информацию об удаленном посетителе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deletevisit",
  "params": {
    "visit": {
      "id": 10
    },
    "token": ""
  },
  "id": 1
}
```

UpdateVisit(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TVisit> DeleteVisit(visit, token)` – редактирование посетителя.

Входные данные

Параметр	Тип	Описание
visit	TVisit	Информация о редактируемом посетителе. Поиск и редактирование посетителя производится по значению поля “id” структуры TVisit.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TVisit>**, содержащий информацию об отредактированном посетителе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "updateVisit",
  "params": {
    "visit": {
```

```
      "id": 123,
      "personId": 5,
      "visitedPersonId": 7,
      "visitedCompanyId": 101,
      "visitedDepartmentId": 202,
      "visitDate": "2025-04-01T10:00:00.000+03:00",
      "visitedRoom": "Conference Room 1",
      "visitedPersonPhone": "+79251233333",
      "visitPurpose": "Business meeting",
      "impersonalPersonId": 2,
      "accessControlRule": 2,
      "visitEndDateTime": "2025-04-01T12:00:00.000+03:00",
      "instructingPersonId": 404,
      "carName": "Toyota Camry",
      "carNumber": "ABC-123",
      "carColor": "Black",
      "isCarOvernight": false
    },
    "token": ""
  },
  "id": 1
}
```


GetCars(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar[]> GetCars(token)` – получение списка автомобилей из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar[]>**, содержащий список автомобилей или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "getCars",
  "params": {
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
```

```
      "success": true,
      "operationResult": [      {
         "id": 1,
         "model": "Toyota Camry",
         "color": "Черный",
         "number": "A123BE77",
         "userIndex": 456,
         "vin": "1HGCM82633A123456"
      }]
   }
}
```


GetCarById(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar> GetCarById(id, token)` – получение информации об автомобиле по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор автомобиля
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar>**, содержащий информацию об автомобиле или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "getCarById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "model": "Toyota Camry",
         "color": "Черный",
         "number": "A123BE77",
         "userIndex": 456,
         "vin": "1HGCM82633A123456"
```

```
        }
    }
}
```

AddCar(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar> AddCar(car, token)` – добавление автомобиля а АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
car	TCar	Информация о добавляемом автомобиле
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar>**, содержащий информацию о добавленном автомобиле или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addcar",
  "params": {
    "car": {
      "id": 123,
      "model": "Toyota Camry",
      "color": "Черный",
      "number": "A123BE77",
      "userindex": 456,
      "vin": "1HGCM82633A123456"
    },
    "token": "your-auth-token"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "model": "Toyota Camry",
         "color": "Черный",
         "number": "A123BE77",
         "userIndex": 456,
         "vin": "1HGCM82633A123456"
      }
   }
```

```
}
```

DeleteCar(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar> DeleteCar(car, token)` – удаление автомобиля.

Входные данные

Параметр	Тип	Описание
car	TCar	Информация об удаляемом автомобиле. Поиск и удаление автомобиля производится по значению поля “id” структуры TCar.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar>**, содержащий информацию об удаленном автомобиле или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteCar",
  "params": {
    "car": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "model": "Toyota Camry",
         "color": "Черный",
         "number": "A123BE77",
         "userIndex": 456,
         "vin": "1HGCM82633A123456"
      }
   }
}
```

UpdateCar(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar> UpdateCar(car, token)` – редактирование автомобиля.

Входные данные

Параметр	Тип	Описание
car	TCar	Информация о редактируемом автомобиле. Поиск и редактирование автомобиля производится по значению поля “id” структуры TCar.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar>**, содержащий информацию об отредактированном автомобиле или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "updateCar",
  "params": {
    "car": {
      "id": 123,
      "model": "Toyota Camry",
      "color": "Черный",
      "number": "A123BE77",
      "userindex": 456,
      "vin": "1HGCM82633A123456"
    },
    "token": "your-auth-token"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "model": "Toyota Camry",
         "color": "Черный",
         "number": "A123BE77",
         "userIndex": 456,
         "vin": "1HGCM82633A123456"
      }
   }
}
```

AddPersonCar(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar> AddPersonCar(car, person, token)` – добавление связи сотрудника с автомобилем.

Входные данные

Параметр	Тип	Описание
car	TCar	Информация об автомобиле. Поиск автомобиля производится по значению поля “id” структуры TCar.
person	TPersonData	Информация о сотруднике. Поиск сотрудника производится по значению поля “id” структуры TPersonData.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar>**, содержащий информацию об автомобиле или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addPersonCar",
  "params": {
    "car": {
      "id": 1
    },
    "person": {
      "id": 12
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "model": "",
         "color": "",
         "number": "",
         "userIndex": 0,
         "vin": ""
      }
   }
}
```

DeletePersonCar(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar> DeletePersonCar(car, person, token)` – удаление связи сотрудника с автомобилем.

Входные данные

Параметр	Тип	Описание
car	TCar	Информация об автомобиле. Поиск автомобиля производится по значению поля “id” структуры TCar.
person	TPersonData	Информация о сотруднике. Поиск сотрудника производится по значению поля “id” структуры TPersonData.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar>**, содержащий информацию об автомобиле или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deletePersonCar",
  "params": {
    "car": {
      "id": 1
    },
    "person": {
      "id": 12
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "model": "",
         "color": "",
         "number": "",
         "userIndex": 0,
         "vin": ""
```

```
        }
    }
}
```

GetPersonCars(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TCar[]> GetPersonCars(person, token)` – получение списка автомобилей сотрудника.

Входные данные

Параметр	Тип	Описание
person	TPersonData	Информация о сотруднике. Поиск сотрудника производится по значению поля “id” структуры TPersonData.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCar[]>**, содержащий список автомобилей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonCars",
  "params": {
    "person": {
      "id": 3
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "model": "Toyota Camry",
            "color": "Черный",
            "number": "A123BE77",
            "userIndex": 456,
            "vin": "1HGCM82633A123456"
         }
      ]
```

```
    }
}
```

GetCarPersons(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.

`TOperationResult<TPersonData[]> GetCarPersons(car, token)` – получение списка сотрудников, связанных с автомобилем.

Входные данные

Параметр	Тип	Описание
car	TCar	Информация об автомобиле. Поиск автомобиля производится по значению поля “id” структуры TCar.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonData[]>**, содержащий список сотрудников или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getCarPersons",
  "params": {
    "car": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```

GetCompanies(...)

`TOperationResult<TCompany[]> GetCompanies(isEmployees, isVisitors, token)` – получение списка компаний из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
isEmployees	bool	Получить компании, связанные с сотрудниками
isVisitors	bool	Получить компании, связанные с посетителями
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCompany[]>**, содержащий список компаний или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getCompanies",
  "params": {
    "isEmployees": true,
    "isVisitors": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "name": "ООО",
            "address": "",
            "phone": "",
            "companyType": 0
         }
      ]
   }
}
```

SearchCompany(...)

`TOperationResult<TCompany[]> SearchCompany(name, token)` – получение отфильтрованного списка организаций из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
name	string	Подстрока для фильтрации списка организаций
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCompany[]>**, содержащий список организаций с учетом фильтра или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
```

```
  "jsonrpc": "2.0",
  "method": "searchCompany",
  "params": {
    "name": "ООО",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "ООО",
         "address": "",
         "phone": "",
         "companyType": 0
      }]
   }
}
```

GetCompanyById(...)

`TOperationResult<TCompany> GetCompanyById(id, token)` – получение информации о компании из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
Id	int	Идентификатор компании
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCompany>**, содержащий информацию о компании или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getCompanyById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "ООО",
         "address": "",
         "phone": "",
         "companyType": 0
      }
   }
}
```

CreateCompany(...)

`TOperationResult<TCompany> CreateCompany(company, token)` – добавление организации в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
company	TCompany	Информация о создаваемой организации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCompany>**, содержащий информацию о созданной организации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "CreateCompany",
  "params": {
    "company": {
         "id": 0,
         "name": "Компания",
         "address": "",
         "phone": "",
         "companyType": 0
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
```

```
      "success": true,
      "operationResult":       {
         "id": 3,
         "name": "Компания",
         "address": "",
         "phone": "",
         "companyType": 0
      }
   }
}
```


UpdateCompany(...)

`TOperationResult<TCompany> UpdateCompany(company, token)` – редактирование организации в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
company	TCompany	Информация об обновляемой организации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCompany>**, содержащий информацию об обновленной организации или информацию об ошибке.

DeleteCompany(...)

`TOperationResult<TCompany> DeleteCompany(company, token)` – удаление организации из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
company	TCompany	Информация об удаляемой организации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TCompany>**, содержащий информацию об удаленной организации или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "deleteCompany",
  "params": {
    "company": {
      "id": 1
    },
```

```
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "",
         "address": "",
         "phone": "",
         "companyType": 0
      }
   }
}
```

GetDepartments(...)

`TOperationResult<TDepartment[]> GetDepartments(token)` – получение списка подразделений из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment[]>**, содержащий список подразделений или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDepartments",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
```

```
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "name": "Бухгалтерия",
            "description": "",
            "companyId": 1,
            "accessLevelId": 0
         }
      ]
   }
}
```

SearchDepartment(...)

`TOperationResult<TDepartment[]> SearchDepartment(name, token)` – получение отфильтрованного списка подразделений из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
name	string	Подстрока для фильтрации списка подразделений
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment[]>**, содержащий список подразделений с учетом фильтра или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "searchDepartment",
  "params": {
    "name": "Бухгалтерия",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "Бухгалтерия",
         "description": "",
         "companyId": 1,
         "accessLevelId": 0
      }]
   }
}
```

GetDepartmentById(...)

`TOperationResult<TDepartment> GetDepartmentById(id, token)` – получение информации о подразделении по его идентификатору из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор подразделения
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment>**, содержащий информацию о подразделении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDepartmentById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "Бухгалтерия",
         "description": "",
         "companyId": 1,
         "accessLevelId": 0
      }
   }
}
```

CreateDepartment(...)

`TOperationResult<TDepartment> CreateDepartment(department, token)` – добавление подразделения в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
department	TDepartment	Информация о создаваемой организации

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment>**, содержащий информацию о созданном подразделении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "CreateDepartment",
  "params": {
    "department": {
      "name": "Отдел продаж"
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 3,
         "name": "Отдел продаж",
         "description": "",
         "companyId": 0,
         "accessLevelId": 0
      }
   }
}
```

UpdateDepartment(...)

`TOperationResult<TDepartment> UpdateDepartment(department, token)` – редактирование подразделения в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
department	TDepartment	Информация об обновляемом подразделении
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment>**, содержащий информацию об обновленном подразделении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "updateDepartment",
  "params": {
    "department": {
      "name": "Отдел продаж"
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 0,
         "name": "Отдел продаж",
         "description": "",
         "companyId": 0,
         "accessLevelId": 0
      }
   }
}
```

DeleteDepartment(...)

`TOperationResult<TDepartment> DeleteDepartment(department, token)` – удаление организации из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
department	TDepartment	Информация об удаляемом подразделении
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment>**, содержащий информацию об удаленном подразделении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
```

```
  "method": "deleteDepartment",
  "params": {
    "department": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "",
         "description": "",
         "companyId": 0,
         "accessLevelId": 0
      }
   }
}
```

GetCompanyDepartments(...)

`TOperationResult<TDepartment[]> GetCompanyDepartments(company, token)` – получение списка подразделений заданной организаци из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
company	TCompany	Информация о компании
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDepartment[]>**, содержащий список подразделений заданной организации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getCompanyDepartments",
  "params": {
    "company": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
```

```
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "name": "Бухгалтерия",
            "description": "",
            "companyId": 1,
            "accessLevelId": 0
         }
      ]
   }
}
```

GetPositions(...)

`TOperationResult<TPosition[]> GetPositions(token)` – получение списка должностей из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPosition[]>**, содержащий список должностей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPositions",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
```

```
      "operationResult": [      {
         "id": 1,
         "name": "Бухгалтер"
      }]
   }
}
```

SearchPosition(...)

`TOperationResult<TPosition[]> SearchPosition(name, token)` – получение отфильтрованного списка должностей из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
name	string	Подстрока для фильтрации списка должностей
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPosition[]>**, содержащий список должностей с учетом фильтра или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "searchPosition",
  "params": {
    "name": "Бухгалтер",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "Бухгалтер"
      }]
   }
}
```

GetPositionById(...)

`TOperationResult<TPosition> GetPositionById(id, token)` – получение информации о должности по ее идентификатору из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор должности
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPosition>**, содержащий информацию о должности или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPositionById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "Бухгалтер"
      }
   }
}
```

CreatePosition(...)

`TOperationResult<TPosition> CreatePosition(position, token)` – добавление должности в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
position	TPosition	Информация о создаваемой должности
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPosition>**, содержащий информацию о созданной должности или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "createPosition",
  "params": {
    "position": {
      "name": "Бухгалтер"
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "name": "Бухгалтер"
      }
   }
}
```

UpdatePosition(...)

`TOperationResult<TPosition> UpdatePosition(position, token)` – редактирование должности в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
position	TPosition	Информация об обновляемой должности
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPosition>**, содержащий информацию об обновленной должности или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "updatePosition",
  "params": {
    "position": {
      "name": "Бухгалтер"
```

```
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "name": "Бухгалтер"
      }
   }
}
```

DeletePosition(...)

`TOperationResult<Position> DeletePosition(position, token)` – удаление должности из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
position	TPosition	Информация об удаляемой должности
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPosition>**, содержащий информацию об удаленной должности или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deletePosition",
  "params": {
    "position": {
      "id": 2
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
```

```
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "name": ""
      }
   }
}
```

GetExitGroups(...)

`TOperationResult<TExitGroup[]> GetExitGroups(token)` – получение списка групп эвакуации из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TExitGroup[]>**, содержащий список групп эвакуации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getExitGroups",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "1",
         "comment": ""
      }]
   }
}
```

SearchExitGroup(...)

`TOperationResult<TExitGroup[]> SearchExitGroup(name, token)` – получение

отфильтрованного списка групп эвакуации из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
name	string	Подстрока для фильтрации списка групп эвакуации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TExitGroup[]>**, содержащий список групп эвакуации с учетом фильтра или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "searchExitGroup",
  "params": {
    "name": "1",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "name": "1",
         "comment": ""
      }]
   }
}
```

GetExitGroupById(...)

`TOperationResult<TExitGroup> GetExitGroupById(id, token)` – получение информации о группе эвакуации по ее идентификатору из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор группы эвакуации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TExitGroup>**, содержащий информацию о группе эвакуации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getExitGroupById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "1",
         "comment": ""
      }
   }
}
```

CreateExitGroup(...)

`TOperationResult<TExitGroup> CreateExitGroup(ExitGroup, token)` – добавление группы эвакуации в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
exitGroup	TExitGroup	Информация о создаваемой группе эвакуации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TExitGroup>**, содержащий информацию о созданной группе эвакуации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "createExitGroup",
  "params": {
```

```
    "exitGroup": {
      "id": 0,
      "name": "1",
      "comment": ""
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "1",
         "comment": ""
      }
   }
}
```

UpdateExitGroup(...)

`TOperationResult<TExitGroup> UpdateExitGroup(ExitGroup, token)` – редактирование группы эвакуации в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
exitGroup	TExitGroup	Информация об обновляемой группе эвакуации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TExitGroup>**, содержащий информацию об обновленной группе эвакуации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "updateExitGroup",
  "params": {
    "exitGroup": {
      "id": 0,
      "name": "1",
      "comment": ""
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "1",
         "comment": ""
      }
   }
}
```

DeleteExitGroup(...)

`TOperationResult<ExitGroup> DeleteExitGroup(ExitGroup, token)` – удаление группы эвакуации из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
exitGroup	TExitGroup	Информация об удаляемой группе эвакуации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TExitGroup>, содержащий информацию об удаленной группе эвакуации или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteExitGroup",
  "params": {
    "exitGroup": {
      "id": 1,
      "name": "",
      "comment": ""
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
```

```
      "operationResult":       {
         "id": 1,
         "name": "",
         "comment": ""
      }
   }
}
```


CreateTimeWindow(...)

`TOperationResult<TTimeWindow> CreateTimeWindow(timeWindow, token)` – создание окна времени.

Входные данные

Параметр	Тип	Описание
timeWindow	TTimeWindow	Информация об окне времени
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TTimeWindow>**, содержащий информацию о созданном окне времени или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "createTimeWindow",
  "params": {
    "timeWindow": {
      "name": "test",
      "calendar":
"BQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgA
QEBABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAE
CAwQFEAYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQ
FBgABAgMQBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgA
BAgMEBQYAAQIDBAUGAAECAwQQBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgM
EBQYAAQIDBAUGAAECAwQFBgABAhADBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFEBAQEBAQEBA
QEBAQEBAQEA==",
      "timeIntervals": [{}]
    },
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 16,
         "name": "",
```

```
        "calendar":
"BQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQF\r\
nBgAQEBABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFEAYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFBgABAgMQBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQQBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAhADBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFEBAQEBAQEBAQEBAQEBAQEA==",
        "timeIntervals": [         {
           "id": 0,
           "timeWindowId": 0,
           "startTime": "",
           "endTime": "",
           "days": [],
           "isEnterActivity": false,
           "isExitActivity": false,
           "isAccessSaturday": false,
           "isAccessSunday": false,
           "isAccessHoliday": false
        }]
      }
   }
}
```


GetTimeWindowById(...)

`TOperationResult<TTimeWindow> GetTimeWindowById(id, token)` – получение окна времени по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор окна времени
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TTimeWindow>, содержащий информацию об окне времени или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getTimeWindowById",
  "params": {
    "id": 1
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
```

```
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 1,
         "name": "Всегда",
         "calendar": "AAMIBwADAAM=",
         "timeIntervals": [         {
            "id": 1,
            "timeWindowId": 1,
            "startTime": "2005-01-01T00:00:00.000Z",
            "endTime": "2019-01-01T23:59:00.000Z",
            "days":             [
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true,
               true
            ],
            "isEnterActivity": true,
            "isExitActivity": true,
            "isAccessSaturday": true,
            "isAccessSunday": true,
            "isAccessHoliday": true
         }]
      }
   }
}
```


DeleteTimeWindow(...)

`TOperationResult<TTimeWindow> CreateTimeWindow(timeWindow, token)` – удаление окна времени.

Входные данные

Параметр	Тип	Описание
timeWindow	TTimeWindow	Информация об удаляемом окне времени. Поиск и удаление окна времени производится по значению поля “id” структуры TTimeWindow.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TTimeWindow>**, содержащий информацию об удаленном окне времени или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deletetimewindow",
  "params": {
    "timewindow": {
      "id": 16
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 16,
         "name": "",
         "calendar": "",
         "timeIntervals": []
      }
   }
}
```

EditTimeWindow(...)

`TOperationResult<TTimeWindow> EditTimeWindow(timeWindow, token)` – редактирование окна времени.

Входные данные

Параметр	Тип	Описание
timeWindow	TTimeWindow	Информация о редактируемом окне времени. Поиск и редактирование окна времени производится по значению поля “id” структуры TTimeWindow.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TTimeWindow>, содержащий информацию об отредактированном окне времени или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
```

```
  "method": "updateTimeWindow",
  "params": {
    "timeWindow": {
      "name": "test",
      "calendar":
"BQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgA
QEBABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAE
CAwQFEAYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQ
FBgABAgMQBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgA
BAgMEBQYAAQIDBAUGAAECAwQQBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgM
EBQYAAQIDBAUGAAECAwQFBgABAhADBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFEBAQEBAQEBA
QEBAQEBAQEA==",
      "timeIntervals": [{}]
    },
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 16,
         "name": "",
         "calendar":
"BQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQF\r\
nBgAQEBABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFEAYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFBgABAgMQBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQQBQYAAQIDBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFBgABAgMEBQYAAQIDBAUGAAECAwQFBgABAhADBAUGAAECAwQFBgABAgMEBQYAAQID\r\
nBAUGAAECAwQFEBAQEBAQEBAQEBAQEBAQEA==",
         "timeIntervals": [         {
            "id": 0,
            "timeWindowId": 0,
            "startTime": "",
            "endTime": "",
            "days": [],
            "isEnterActivity": false,
            "isExitActivity": false,
            "isAccessSaturday": false,
            "isAccessSunday": false,
            "isAccessHoliday": false
         }]
      }
   }
}
```


GetTimeWindows(...)

`TOperationResult<TTimeWindow[]> GetTimeWindows(token)` - получение списка окон времени.

Входные данные

Параметр	Тип	Описание

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TTimeWindow[]>**, содержащий список окон времени или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getTimeWindows",
  "params": {},
  "id": 1
}
```

Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "name": "Всегда",
            "calendar": "AAMIBwADAAM=",
            "timeIntervals": [            {
               "id": 1,
               "timeWindowId": 1,
               "startTime": "2005-01-01T00:00:00.000Z",
               "endTime": "2019-01-01T23:59:00.000Z",
               "days":                [
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true,
                  true
               ],
               "isEnterActivity": true,
               "isExitActivity": true,
               "isAccessSaturday": true,
               "isAccessSunday": true,
               "isAccessHoliday": true
            }]
         }
```

```
      ]
    }
}
```

AddTimeInterval(...)

`TOperationResult<TTimeInterval> AddTimeInterval(timeWindow, timeInterval, token)` – добавление временного интервала в окно времени.

Входные данные

Параметр	Тип	Описание
timeWindow	TTimeWindow	Окно времени для добавления временного интервала. Поиск окна времени производится по значению поля “id” структуры TTimeWindow.
timeInterval	TTimeInterval	Информация о добавляемом временном интервале
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TTimeInterval>, содержащий информацию о добавленном временном интервале или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addTimeInterval",
  "params": {
    "timeWindow": {
      "id": 3
    },
    "timeInterval": {
      "startTime": "2000-01-01T08:00:00.000+03:00",
      "endTime": "2000-01-01T18:00:00.000+03:00",
      "days": [
        true, true, true, true, true,
        false, false, false, false, false,
        false, false, false, false, false, false
      ],
      "isEnterActivity": true,
      "isExitActivity": true,
      "isAccessSaturday": false,
      "isAccessSunday": false,
      "isAccessHoliday": false
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
```

```
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 11,
         "timeWindowId": 0,
         "startTime": "2000-01-01T05:00:00.000Z",
         "endTime": "2000-01-01T15:00:00.000Z",
         "days":          [
            true,
            true,
            true,
            true,
            true,
            false,
            false,
            false,
            false,
            false,
            false,
            false,
            false,
            false,
            false,
            false
         ],
         "isEnterActivity": true,
         "isExitActivity": true,
         "isAccessSaturday": false,
         "isAccessSunday": false,
         "isAccessHoliday": false
      }
   }
}
```


DeleteTimeInterval(...)

`TOperationResult<TTimeInterval> DeleteTimeInterval(timeInterval, token)` – удаление временного интервала.

Входные данные

Параметр	Тип	Описание
timeInterval	TTimeInterval	Информация об удаляемом временном интервале. Поиск и удаление временного интервала производится по значению поля “id” структуры TTimeInterval.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TTimeInterval>, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteTimeInterval",
  "params": {
    "timeinterval": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result": {"success": true}
}
```

EditTimeInterval(...)

`TOperationResult<TTimeInterval> EditTimeInterval(timeInterval, token)` – редактирование временного интервала.

Входные данные

Параметр	Тип	Описание
timeInterval	TTimeInterval	Информация о редактируемом временном интервале. Поиск и редактирование временного интервала производится по значению поля “id” структуры TTimeInterval.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TTimeInterval>, содержащий информацию об отредактированном временном интервале или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "edittimeinterval",
  "params": {
    "timeinterval": {
      "id": 2,
      "timewindowid": 2,
      "starttime": "2018-09-14T00:00:00Z",
      "endtime": "2018-09-14T00:00:00Z"
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "timeWindowId": 2,
         "startTime": "2018-09-14T00:00:00.000Z",
         "endTime": "2018-09-14T00:00:00.000Z",
         "days": [],
         "isEnterActivity": false,
         "isExitActivity": false,
         "isAccessSaturday": false,
         "isAccessSunday": false,
         "isAccessHoliday": false
      }
   }
}
```

CreateAccessLevel(...)

`TOperationResult<TAccessLevel> CreateAccessLevel(accessLevel, token)` – создание уровня доступа.

Входные данные

Параметр	Тип	Описание
accessLevel	TAccessLevel	Информация об уровне доступа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAccessLevel>**, содержащий информацию о созданном уровне доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "createAccessLevel",
  "params": {
    "accessLevel": {
      "id": 0,
      "name": "тест1",
      "description": "?",
      "items": []
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 18,
         "name": "тест1",
         "description": "?",
         "items": []
      }
   }
}
```

EditAccessLevel(...)

`TOperationResult<TAccessLevel> CreateAccessLevel(accessLevel, token)` – редактирование уровня доступа.

Входные данные

Параметр	Тип	Описание
accessLevel	TAccessLevel	Информация о редактируемом уровне доступа. Поиск и редактирование уровня доступа производится по значению поля “id” структуры TAccessLevel.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TAccessLevel>, содержащий информацию об отредактированном уровне доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "editaccesslevel",
  "params": {
    "accesslevel": {
      "id": 10,
      "name": "test"
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
```

```
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 10,
         "name": "test",
         "description": "",
         "items": []
      }
   }
}
```


DeleteAccessLevel(...)

`TOperationResult<TAccessLevel> DeleteAccessLevel(accessLevel, token)` – удаление уровня доступа.

Входные данные

Параметр	Тип	Описание
accessLevel	TAccessLevel	Информация об удаляемом уровне доступа. Поиск и удаление уровня доступа производится по значению поля “id” структуры TAccessLevel.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TTimeWindow>**, содержащий информацию об удаленном уровне доступа или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "deleteAccessLevel",
  "params": {
    "accessLevel": {
      "id": 10
    },
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 10,
         "name": "",
         "description": "",
         "items": []
```

```
        }
    }
}
```

AddAccessLevelItem(...)

`TOperationResult<TAccessLevelItem> AddAccessLevelItem(accessLevel, item, token)` – добавление элемента (полномочия) в уровень доступа.

Входные данные

Параметр	Тип	Описание
accessLevel	TAccessLevel	Уровень доступа, в который необходимо добавить элемент (полномочие). Поиск уровня доступа производится по значению поля “id” структуры TAccessLevel.
Item	TAccessLevelItem	Добавляемый элемент (полномочие)
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TAccessLevelItem>, содержащий информацию об элементе (полномочии) уровня доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addAccessLevelItem",
  "params": {
    "accessLevel": {
      "id": 4
    },
    "item": {
      "id": 0,
      "itemType": "door",
      "itemId": 789,
      "timeWindowId": 101,
      "rights": 1,
      "antipassback": 0,
      "lockTime": 30,
      "isZonalAntipassback": true,
      "doubleConfirmationId": 202,
      "tripleConfirmationId": 303,
      "isConfirming": false,
      "isConfirmationButton": true
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 20,
         "itemType": "door",
         "itemId": 789,
         "timeWindowId": 101,
         "rights": 1,
         "antipassback": 0,
         "lockTime": 30,
         "isZonalAntipassback": true,
         "doubleConfirmationId": 202,
         "tripleConfirmationId": 303,
         "isConfirming": false,
         "isConfirmationButton": true,
         "isAddCode": false
      }
   }
}
```

EditAccessLevelItem(...)

`TOperationResult<TAccessLevelItem> AddAccessLevelItem(item, token)` – редактирование элемента (полномочия) уровня доступа.

Входные данные

Параметр	Тип	Описание
Item	TAccessLevelItem	Редактируемый элемент (полномочие) Поиск и редактирования элемента (полномочия) элемента производится по значению поля “id” структуры TAccessLevelItem.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TAccessLevelItem>, содержащий информацию об отредактированном элементе (полномочии) уровня доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "editaccesslevelitem",
  "params": {
    "item": {
      "id": 20
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 20,
         "itemType": "",
         "itemId": 0,
         "timeWindowId": 0,
         "rights": 0,
         "antipassback": 0,
         "lockTime": 0,
         "isZonalAntipassback": false,
         "doubleConfirmationId": 0,
         "tripleConfirmationId": 0,
         "isConfirming": false,
         "isConfirmationButton": false,
         "isAddCode": false
      }
   }
}
```

DeleteAccessLevelItem(...)

`TOperationResult DeleteAccessLevelItem(item, token)` – удаление элемента (полномочия) уровня доступа.

Входные данные

Параметр	Тип	Описание
item	TAccessLevelItem	Удаляемый элемент (полномочие) уровня доступа. Поиск и удаление элемента (полномочия) производится по значению поля “id” структуры TAccessLevelItem.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteAccessLevelItem",
  "params": {
    "item": {
      "id": 10
    },
    "token": ""
```

```
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result": {"success": true}
}
```

GetKeyData(...)

`TOperationResult<TKeyData> GetKeyData(cardNo, token)` – получение данных ключа по его коду.

Входные данные

Параметр	Тип	Описание
cardNo	string	Код ключа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TKeyData>**, содержащий данные ключа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getKeyData",
  "params": {
    "cardNo": "F4FFFFFFF12345F0",
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 3,
         "codeType": 12,
         "additionalCodeType": 0,
         "code": "F4FFFFFFF12345F0",
         "additionalCode": "",
```

```
            "personId": 3,
            "accessLevelId": 5,
            "startDate": "2025-06-19T00:00:00.000Z",
            "endDate": "2075-06-19T00:00:00.000Z",
            "isBlocked": false,
            "isStoreInDevice": true,
            "isStoreInS2000": true,
            "isInStopList": false,
            "comment": "",
            "login": ""
        }
    }
}
```


GetKeyDataById(...)

`TOperationResult<TKeyData> GetKeyDataById(id, token)` – получение данных ключа по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор ключа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult<TKeyData>, содержащий данные ключа или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "getKeyDataById",
  "params": {
    "id": 3,
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 3,
         "codeType": 12,
         "additionalCodeType": 0,
         "code": "F4FFFFFFF12345F0",
         "additionalCode": "",
```

```
        "personId": 3,
        "accessLevelId": 5,
        "startDate": "2025-06-19T00:00:00.000Z",
        "endDate": "2075-06-19T00:00:00.000Z",
        "isBlocked": false,
        "isStoreInDevice": true,
        "isStoreInS2000": true,
        "isInStopList": false,
        "comment": "",
        "login": ""
      }
    }
}
```


UpdateKeyData(...)

`TOperationResult<TKeyData> UpdateKeyData(keyData, token)` – редактирование данных ключа.

Входные данные

Параметр	Тип	Описание
keyData	TKeyData	Данные ключа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TKeyData>**, содержащий данные ключа или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "updateKeyData",
  "params": {
    "keyData": {
      "id": 9,
      "codeType": 4,
      "code": "55000000BA963F01",
      "personId": 10,
      "accessLevelId": 2,
      "startDate": "2018-09-14T00:00:00Z",
      "endDate": "2118-01-01T23:59:59Z",
      "isBlocked": false,
      "isStoreInDevice": true,
      "isStoreInS2000": false,
      "isInStopList": false
    },
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 9,
         "codeType": 4,
         "additionalCodeType": 0,
         "code": "55000000BA963F01",
         "additionalCode": "",
         "personId": 10,
         "accessLevelId": 2,
         "startDate": "2018-09-14T00:00:00.000Z",
         "endDate": "2118-01-01T23:59:59.000Z",
         "isBlocked": false,
         "isStoreInDevice": true,
         "isStoreInS2000": false,
         "isInStopList": false,
         "comment": "",
         "login": ""
      }
   }
}
```


AddKey(...)

`TOperationResult<TKeyData> AddKey(keyData, token)` – добавление ключа.

Входные данные

Параметр	Тип	Описание
keyData	TKeyData	Данные ключа
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TKeyData>**, содержащий данные добавленного ключа или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "addKey",
  "params": {
    "keyData": {
      "codeType": 4,
      "code": "690000002C223801",
      "personId": 2,
      "accessLevelId": 15,
      "startDate": "2023-01-01T00:00:00.000+03:00",
      "endDate": "2029-01-01T00:00:00.000+03:00",
      "isBlocked": false,
      "isStoreInDevice": true,
      "isStoreInS2000": false,
```

```
      "isInStopList": false
    }
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 12,
         "codeType": 4,
         "additionalCodeType": 0,
         "code": "690000002C223801",
         "additionalCode": "",
         "personId": 2,
         "accessLevelId": 15,
         "startDate": "2022-12-31T21:00:00.000Z",
         "endDate": "2028-12-31T21:00:00.000Z",
         "isBlocked": false,
         "isStoreInDevice": true,
         "isStoreInS2000": false,
         "isInStopList": false,
         "comment": "",
         "login": ""
      }
   }
}
```


GetKeys(...)

`TOperationResult<TKeyData> GetKeys(codeType, personId, persons, offset, count, token)` – получение списка ключей в системе.

Входные данные

Параметр	Тип	Описание
codeType	int	Тип кода ключа для фильтрации списка ключей. 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
personId	int	Идентификатор владельца для фильтрации списка

		ключей.
persons	TPersonData[]	Список сотрудников для фильтрации списка ключей.
offset	int	Смещение относительно начала полного списка ключей. Значение по умолчанию: 0.
count	int	Количество возвращаемых ключей начиная со смещения offset. При значениях данного параметра и параметра offset равным 0, возвращается полный список ключей. Значение по умолчанию: 0.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TKeyData[]>**, содержащий список ключей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getKeys",
  "params": {
    "offset": 0
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "codeType": 1,
            "additionalCodeType": 0,
            "code": "1",
            "additionalCode": "",
            "personId": 1,
            "accessLevelId": 2,
            "startDate": "2024-06-17T14:14:27.000Z",
            "endDate": "2035-06-17T14:14:27.000Z",
            "isBlocked": false,
            "isStoreInDevice": false,
            "isStoreInS2000": false,
            "isInStopList": true,
            "comment": "",
            "login": ""
         }
      ]
   }
}
```

GetKeysCount(...)

`TOperationResult<int> GetKeysCount(codeType, personId, persons, token)`- получение количества ключей в системе.

Входные данные

Параметр	Тип	Описание
codeType	int	Тип кода ключа для фильтрации списка ключей. 1 — пароль для программ; 2 — пин-код; 3 — брелок TochMemory; 4 — Proxy-карта; 5 — автомобильный номер; 7 — отпечаток пальца; 8 — шаблон лица SB101; 9 — шаблон ладони; 12 — пин-код2; 16 — Шаблон лица SF10/SF10T/SF50; 17 — Шаблон лица из фото; 18 — QR-код; 19 — Шаблон лица SF6P.
personId	int	Идентификатор владельца для фильтрации списка ключей.
persons	TPersonData[]	Список сотрудников для фильтрации списка ключей.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество ключей в системе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getKeysCount",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
```

```
      "operationResult": 11
   }
}
```

GetComputers(...)

`TOperationResult<TComputer[]> GetComputers(token)` - получение списка компьютеров.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TComputer[]>**, содержащий список компьютеров или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getComputers",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 1,
         "ip": "127.0.0.1",
         "name": "PROG-71",
         "number": 1
      }]
   }
}
```

GetComPorts(...)

`TOperationResult<TComPort[]> GetComPorts(token)` - получение списка портов.

В «Орион Про» начиная с версии 1.20.3.6 отсутствует такая сущность, как COM порт, поэтому данный запрос более неактуален.

Входные данные

Параметр	Тип	Описание

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TComPort[]>**, содержащий список портов или информацию об ошибке.

GetDevices(...)

`TOperationResult<TDevice[]> GetDevices(token)` - получение списка устройств.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDevice[]>**, содержащий список устройств или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDevices",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 1,
            "address": 0,
            "devType": 51,
            "name": "Драйвер Орион 2",
            "comPortId": 1,
            "pKUId": 0,
            "hwId": 0,
            "ipAddress": "127.0.0.1",
            "computerId": 1,
            "number": 1
         },
                  {
```

```
            "id": 6,
            "address": 0,
            "devType": 57,
            "name": "ППКУП Сириус",
            "comPortId": 2,
            "pKUId": 1,
            "hwId": 0,
            "ipAddress": "13.0.0.1",
            "computerId": 1,
            "number": 6
        },
                {
            "id": 7,
            "address": 1,
            "devType": 68,
            "name": "С2000-КПБ-С(1)",
            "comPortId": 2,
            "pKUId": 6,
            "hwId": 0,
            "ipAddress": "",
            "computerId": 1,
            "number": 7
        },
                {
            "id": 8,
            "address": 2,
            "devType": 50,
            "name": "МИП-24-С исп. 03(2)",
            "comPortId": 2,
            "pKUId": 6,
            "hwId": 0,
            "ipAddress": "",
            "computerId": 1,
            "number": 8
        },
                {
            "id": 9,
            "address": 3,
            "devType": 40,
            "name": "С2000-КДЛ-С(3)",
            "comPortId": 2,
            "pKUId": 6,
            "hwId": 0,
            "ipAddress": "",
            "computerId": 1,
            "number": 9
        }
    ]
  }
}
```


GetDeviceById(...)

`TOperationResult<TDevice> GetDeviceById(id, token)` - получение информации об устройстве.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор устройства

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDevice>**, содержащий информацию об устройстве или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDeviceById",
  "params": {
    "id": 6,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 6,
         "address": 0,
         "devType": 57,
         "name": "ППКУП Сириус",
         "comPortId": 2,
         "pKUId": 1,
         "hwId": 0,
         "ipAddress": "13.0.0.1",
         "computerId": 1,
         "number": 6
      }
   }
}
```

GetDevicesItems(...)

`TOperationResult<TDeviceItem[]> GetDevicesItems(itemTypes, token)` - получение списка всех элементов устройств системы.

Входные данные

Параметр	Тип	Описание
itemTypes	string[]	Список типов для фильтрации.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDeviceItem[]>**, содержащий список элементов всех устройств системы или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDevicesItems",
  "params": {
    "deviceId": 8,
    "token": ""
  },
  "id": 1
}
```

GetDeviceItems(...)

`TOperationResult<TDeviceItem[]> GetDeviceItems(deviceId, token)` - получение списка элементов устройства.

Входные данные

Параметр	Тип	Описание
deviceId	int	Идентификатор устройства
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDeviceItem[]>**, содержащий список элементов устройства или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDeviceItems",
  "params": {
    "deviceId": 8,
    "token": ""
  },
  "id": 1
}
```

GetDeviceItemById(...)

`TOperationResult<TDeviceItem> GetDeviceItems(Id, token)` - получение информации об элементе устройства.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор элемента устройства
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDeviceItem>**, содержащий информацию об элементе устройства или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getDeviceItemById",
  "params": {
      "id": 801
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 801,
         "name": "Реле 2, Прибор  1",
         "deviceId": 10,
         "itemType": "RELAY",
         "address": 2,
         "additionalType": 0,
         "remoteControlCategoryId": 6,
         "contactId": 0,
         "computerId": 1,
         "number": 801
      }
   }
}
```

GetSections(...)

`TOperationResult<TSection[]> GetSections(isIncludeItems, offset, count, token)`- получение списка зон.

Входные данные

Параметр	Тип	Описание
isIncludeItems	bool	Включать в результат элементы, входящие в зоны
offset	int	Смещение относительно начала полного списка зон
count	int	Количество возвращаемых зон начиная с позиции

		offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TSection[]>**, содержащий список зон или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSections",
  "params": {
    "isIncludeItems": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 2,
            "number": 3,
            "name": "Зона",
            "computerId": 1,
            "sectionTypeId": 0,
            "sectionTypeName": "",
            "items": [],
            "remoteControlCategoriesIds": []
         }
      ]
   }
}
```

GetSectionsCount(...)

`TOperationResult<int> GetSectionsCount(token)` - получение количества зон в системе.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество зон или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionsCount",
  "params": {
    "isIncludeItems": false,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
   }
}
```

GetSectionById(...)

`TOperationResult<TSection> GetSectionById(id, isIncludeItems, token)`- получение информации о зоне по ее идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор зоны
isIncludeItems	bool	Включать в результат элементы, входящие в зону
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TSection>**, содержащий информацию о зоне или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionById",
  "params": {
    "id": 2,
    "isIncludeItems": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "number": 3,
         "name": "Зона",
         "computerId": 1,
         "sectionTypeId": 0,
         "sectionTypeName": "",
         "items": [],
         "remoteControlCategoriesIds": []
      }
   }
}
```

GetSectionsGroups(...)

`TOperationResult<TSectionsGroup[]> GetSectionsGroups(isIncludeItems, offset, count, token)` - получение списка групп зон.

Входные данные

Параметр	Тип	Описание
isIncludeItems	bool	Включать в результат разделы, входящие в группы зон
offset	int	Смещение относительно начала полного списка групп зон
count	int	Количество возвращаемых групп зон начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TSectionsGroup[]>**, содержащий список групп зон или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionsGroups",
  "params": {
    "isIncludeItems": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
```

```
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 2,
            "number": 6,
            "name": "Все зоны",
            "computerId": 1,
            "items": [],
            "remoteControlCategoriesIds": []
         }
      ]
   }
}
```


GetSectionsGroupsCount(…)

`TOperationResult<int> GetSectionsGroupsCount(token)` - получение количества групп зон в системе.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество групп зон или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "getSectionsGroupsCount",
  "params": {
    "isIncludeItems": false,
    "token": ""
  },
  "id": 1
}
```


Пример результата запроса JSON-RPC:


```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
   }
}
```

GetSectionsGroupById(...)

`TOperationResult<TSectionsGroup> GetSectionsGroupById(id, isIncludeItems, token)` - получение информации о группе зон по ее идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор группы зон
isIncludeItems	bool	Включать в результат разделы, входящие в группу зон
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TSectionsGroup>**, содержащий информацию о группе зон или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionsGroupById",
  "params": {
    "id": 2,
    "isIncludeItems": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "number": 6,
         "name": "Все зоны",
         "computerId": 1,
         "items": [],
         "remoteControlCategoriesIds": []
      }
   }
}
```

GetSectionItems(...)

`TOperationResult<TDeviceItem[]> GetSectionItems(section, offset, count, token)`- получение списка элементов приборов, входящих в состав указанной зоны.

Входные данные

Параметр	Тип	Описание
section	TSection	Зона, для которого запрашивается список элементов
offset	int	Смещение относительно начала полного списка элементов
count	int	Количество возвращаемых элементов начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TDeviceItem[]>**, содержащий список элементов приборов, входящих в состав указанной зоны, или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionItems",
  "params": {
    "section": {
      "id": 4
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 8,
            "name": "МИП-24-С исп. 03(2)",
            "deviceId": 8,
            "itemType": "DEVICE",
            "address": 2,
            "additionalType": 0,
            "remoteControlCategoryId": 404,
            "contactId": 0,
            "computerId": 1,
            "number": 0
         }
      ]
   }
}
```

GetSectionItemsCount(...)

`TOperationResult<int> GetSectionItemsCount(section, token)` - получение количества элементов, входящих в состав указанной зоны.

Входные данные

Параметр	Тип	Описание
section	TSection	Зона, для которого запрашивается количество элементов
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество элементов или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionItemsCount",
  "params": {
    "section": {
      "id": 4
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
   }
}
```

GetSectionsGroupItems(...)

`TOperationResult<TSection[]> GetSectionsGroupItems(sectionsGroup, offset, count, token)` - получение списка зон, входящих в состав указанной группы зон.

Входные данные

Параметр	Тип	Описание
sectionsGroup	TSectionsGroup	Группа зон, для которой запрашивается список зон

offset	int	Смещение относительно начала полного списка зон
count	int	Количество возвращаемых зон начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TSection[]>**, содержащий список зон, входящих в состав указанной группы зон, или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionsGroupsItems",
  "params": {
    "sectionsGroup": {
      "id": 4
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 7,
         "number": 7,
         "name": "Зона АСПТ ШС",
         "computerId": 1,
         "sectionTypeId": 0,
         "sectionTypeName": "",
         "items": [],
         "remoteControlCategoriesIds": []
      }]
   }
}
```

GetSectionsGroupItemsCount(...)

`TOperationResult<int> GetSectionItemsCount(sectionsGroup, token)`- получение количества зон, входящих в состав указанной группы зон.

Входные данные

Параметр	Тип	Описание
sectionsGroup	TSectionsGroup	Группа зон, для которой запрашивается количество зон

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество зон или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getSectionsGroupsItemsCount",
  "params": {
    "sectionsGroup": {
      "id": 4,
      "number": 0,
      "name": "",
      "computerId": 0,
      "items": [],
      "remoteControlCategoriesIds": []
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
   }
}
```

GetItems(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TItem[]> GetItems(offset, count, token)` - получение списка элементов системы с информацией об их состоянии.

Входные данные

Параметр	Тип	Описание
offset	int	Смещение относительно начала полного списка элементов
count	int	Количество возвращаемых элементов начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItem[]>**, содержащий список элементов системы или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getItems",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "itemType": "DEVICE",
            "itemId": 1,
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:32.000Z"
         },
                  {
            "itemType": "DEVICE",
            "itemId": 14,
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:33.000Z"
         },
                  {
            "itemType": "DEVICE",
            "itemId": 13,
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:33.000Z"
         }
      ]
   }
}
```

GetItemsByTimeOfChange(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

```
TOperationResult<TItem[]> GetItemsByTimeOfChange(beginTime, endTime, token)-
```

получение списка элементов системы с информацией об их состоянии в соответствии с фильтром по времени последнего изменения.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItem[]>**, содержащий список элементов системы или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getItemsByTimeOfChange",
  "params": {
    "startTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "itemType": "DEVICE",
            "itemId": 1,
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:32.000Z"
         },
                  {
            "itemType": "DEVICE",
            "itemId": 14,
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:33.000Z"
         },
                  {
            "itemType": "DEVICE",
            "itemId": 13,
```

```
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:33.000Z"
        }
    ]
  }
}
```

GetItemsCount(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<int> GetItemsCount(token)` - получение количества элементов в системе.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество элементов или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getItemsCount",
  "params": {
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 3
   }
}
```

GetItemById(...)

`TOperationResult<TItem> GetItemById(id, itemType, token)` - получение состояния элемента по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор элемента
itemType	string	Тип элемента
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItem>**, содержащий информацию о состоянии элемента или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getItemById",
  "params": {
    "id": 1,
    "ItemType": "DEVICE"
    "isIncludeItems": true,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": {
            "itemType": "DEVICE",
            "itemId": 1,
            "rights": 0,
            "state": 250,
            "computerId": 1,
            "ownerId": 0,
            "timestamp": "2025-01-23T07:02:32.000Z"
      }
   }
}
```

GetReaderItems(...)

`TOperationResult<TReaderItem[]> GetReaderItems(readerId, token)` - получение списка элементов, связанных со считывателем.

Входные данные

Параметр	Тип	Описание
readerId	int	Идентификатор считывателя
offset	int	Смещение относительно начала полного списка

		элементов, связанных со считывателем
count	int	Количество возвращаемых элементов, связанных со считывателем, начиная со смещения offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TReaderItem[]>**, содержащий список элементов элементов, связанных со считывателем, или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getReaderItems",
  "params": {
    "readerId": 400,
    "offset": 0,
    "count": 0,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "id": 3,
         "readerId": 400,
         "itemId": 2,
         "itemType": "SECTIONGROUP",
         "rights": 3
      }]
   }
}
```

GetReaderItemsCount(...)

`TOperationResult<int> GetReaderItemsCount(readerId, token)` - получение количества элементов, связанных со считывателем.

Входные данные

Параметр	Тип	Описание
readerId	int	Идентификатор считывателя
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество элементов, связанных со считывателем или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getReaderItemsCount",
  "params": {
    "readerId": 400,
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": 1
   }
}
```

AddReaderItem(...)

`TOperationResult<TReaderItem> AddReaderItem(readerItem, token)`- добавление связи элемента (зоны, группы зон, точки доступа, зоны доступа) со считывателем.

Входные данные

Параметр	Тип	Описание
readerItem	TReaderItem	Связь элемента со считывателем.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TReaderItem>**, содержащий информацию о добавленной связи элемента со считывателем или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addReaderItem",
  "params": {
    "readerItem": {
      "id": 0,
      "readerId": 400,
```

```
      "itemId": 4,
      "itemType": "SECTIONGROUP",
      "rights": 0
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 8,
         "readerId": 400,
         "itemId": 4,
         "itemType": "SECTIONGROUP",
         "rights": 0
      }
   }
}
```

DeleteReaderItem(...)

`TOperationResult DeleteReaderItem(readerItem, token)` - удаление связи элемента (зоны, группы зон, точки доступа, зоны доступа) со считывателем.

Входные данные

Параметр	Тип	Описание
readerItem	TReaderItem	Связь элемента со считывателем.
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса TOperationResult, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteReaderItem",
  "params": {
    "readerItem": {
      "id": 8
    },
    "token": ""
  },
  "id": 1
```

```
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result": {"success": true}
}
```

GetLoginToken(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<string> GetLoginToken(login, md5Passw)` - получение токена для вызова методов. Время действия полученного токена составляет 300 секунд, до истечения этого времени необходимо продлить срок действия токена с помощью метода ExtendTokenExpiration().

Входные данные

Параметр	Тип	Описание
login	string	Логин пользователя. Указывается в АБД на вкладке «Пароли» для паролей с типом «Удаленное управление».
md5Passw	string	MD5 хэш пароля пользователя.

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий токен или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getLoginToken",
  "params": {
    "login": "test11",
    "md5Passw": "098f6bcd4621d373cade4e832627b4f6"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "{F1F0691C-7513-4851-A8CD-CB23600796DC}"
   }
}
```

ExtendTokenExpiration(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<string> ExtendTokenExpiration(token)` - обновление времени действия токена (300 сек.).

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя, время действия которого необходимо продлить.

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий токен или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "extendTokenExpiration",
  "params": {
    "token": "{F1F0691C-7513-4851-A8CD-CB23600796DC}"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "{F1F0691C-7513-4851-A8CD-CB23600796DC}"
   }
}
```

GetItemsForLogin(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TItem[]> GetItemsForLogin(token, login)` - получение элементов системы, доступных для пользователя с указанным логином. Список формируется с учетом уровня доступа прльзователя.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя.
login	string	Логин пользователя, по которому запрашивается информация.
items	TItem[]	Список элементов, по которым запрашивается информация. Если список пустой, метод возвращает список всех элементов, доступ к которым есть у

		пользователя.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItem[]>**, содержащий список элементов системы или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getItemsForLogin",
  "params": {
    "token": "{F1F0691C-7513-4851-A8CD-CB23600796DC}",
    "login": "test11"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "itemType": "SECTION",
            "itemId": 7,
            "rights": 3,
            "state": -1,
            "computerId": -1,
            "ownerId": -1,
            "timestamp": ""
         },
                  {
            "itemType": "SECTION",
            "itemId": 6,
            "rights": 3,
            "state": -1,
            "computerId": -1,
            "ownerId": -1,
            "timestamp": ""
         }
      ]
   }
}
```

GetItemsStates(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TItem[]> GetItemsStates(token, items)` - получение состояний элементов системы.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя.
items	TItem[]	Список элементов, для которых запрашиваются состояния. Если список пустой, метод возвращает состояния всех элементов системы, доступных авторизованному пользователю.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItem[]>**, содержащий список элементов системы с их состояниями или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getItemsStates",
  "params": {
    "token": "{E4F3163C-18D4-4B50-AEA3-025C8A21BAB3}"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "itemType": "LOOP",
            "itemId": 741,
            "rights": 0,
            "state": 112,
            "computerId": -1,
            "ownerId": -1,
            "timestamp": ""
         },
                  {
            "itemType": "LOOP",
            "itemId": 745,
            "rights": 0,
            "state": 112,
            "computerId": -1,
            "ownerId": -1,
            "timestamp": ""
         }
      ]
   }
}
```

ControlItems(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TItem[]> ControlItems(token, items, command, action, personId)`- управление элементами системы.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя.
items	TItem[]	Список элементов, которыми осуществляется управление.
command	int	Команда управления. Для входов: 2 — взятие; 3 — снятие; 4 — включение автоматики; 5 — выключение автоматики; 6 — пуск; 7 — отмена; 13 — отключение входа; 14 — включение входа; Для выходов: 0 — возврат в исходное состояние; 1 — включить; 2 — выключить; 3 — включить на время; 4 — выключить на время; 5 — мигать из состояния выключен; 6 — мигать из состояния включено; 7 — мигать из состояния выключено на время; 8 — мигать из состояния включено на время; 11 — пуск АСПТ; 12 — Пуск/стоп речевого оповещения; 13 — ручной пуск; 14 — ручной останов; 15 — отключение управления и контроля. 16 — включение управления и контроля. Для зон/групп зон: 1 — снятие; 2 — взятие; 13 — отключение; 14 — включение. Для дверей/считывателей: 0 — предоставление доступа; 1 — разрешение (восстановление) доступа; 2 — разрешение входа; 3 — разрешение выхода;

		4 — запрет доступа (входа и выхода); 5 — запрет входа; 6 — запрет выхода; 7 — открытие доступа; Для считывателей, если требуется подтвердить доступ (данные команды следует выполнять при получении события доступа с флагом IsConfirmationWaiting): 8 — подтверждение доступа; 9 — запрет доступа; 10 — разрешение доступа (разрешить доступ идентфиикатору, для которого проход запрещен).
action	int	Дополнительные данные для команты управления. Для выходов — номер маски мигания. Для дверей: 0 — проход; 1 — вход; 2 — выход.
personId	int	Идентификатор хозоргана
passwordId	int	Идентификатор пароля, которому предоставляется доступ

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItem[]>**, содержащий список элементов системы или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "controlItems",
  "params": {
    "token": "{57AAAE1E-6B4D-43E6-B2B6-EC5F899B7A24}",
    "items": [
      {
        "itemType": "ACCESSPOINT",
        "itemId": 1,
        "computerId": 1
      }
    ],
    "command": 0,
    "action": 1,
    "personId": 1,
    "passwordId": 1
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
```

```
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": [      {
         "itemType": "ACCESSPOINT",
         "itemId": 1,
         "rights": 0,
         "state": -1,
         "computerId": 1,
         "ownerId": -1,
         "timestamp": ""
      }]
   }
}
```


GetADCValues(...)

`TOperationResult<TItem[]> GetADCValues(beginTime, endTime, items, token)`-запрос АЦП.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации.
endTime	DateTime	Дата и время конца для фильтрации.
items	TItem[]	Список входов и выходов для запроса АЦП.
token	string	Токен авторизованного пользователя.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TItemADCValue[]>**, содержащий информацию об АЦП элементов системы или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "getAdcValues",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "items": [
      {
        "itemId": 3,
        "itemType": "RELAY"
      }
    ],
    "token": "{D3D4A3A2-5AC2-4A78-ADB7-CDB746BB559B}"
  },
  "id": 1
}
```

GetScripts(...)

`TOperationResult<TScript[]> GetScripts(token)` - получение списка сценариев управления.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя.

Результирующие данные

Метод возвращает объект класса **TOperationResult<TScript[]>**, содержащий список сценариев управления или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getScripts",
  "params": {
    "token": "{071E625C-6678-4CA4-9315-4557E08A7F02}"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       [
                  {
            "id": 3,
            "name": "Сценарий запроса",
            "comment": ""
         }
      ]
   }
}
```

SendPasswordToEmail(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult SendPasswordToEmail(login, email)` - восстановление пароля. На указанный e-mail отправляется письмо с временным паролем.

Входные данные

Параметр	Тип	Описание
login	string	Логин пользователя для восстановления пароля.
email	string	E-mail пользователя для восстановления пароля.

Результирующие данные

Метод возвращает объект класса **TOperationResult**, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "sendPasswordToEmail",
  "params": {
    "login": "test11",
    "email": "your e-mail",
    "token": ""
  },
  "id": 1
}
```

GetRecognitionChannels(...)

`TOperationResult<TRecognitionChannel[]> GetRecognitionChannels(offset, count, token)` - получение списка каналов распознавания.

Входные данные

Параметр	Тип	Описание
offset	int	Смещение относительно начала полного списка каналов распознавания
count	int	Количество возвращаемых каналов распознавания начиная с позиции offset
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRecognitionChannel[]>**, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getRecognitionChannels",
  "params": {
    "offset": 0,
    "count": 0,
    "token": ""
  },
  "id": 1
}
```

GetRecognitionChannelsCount(...)

`TOperationResult<int> GetRecognitionChannelsCount(token)` - получение количества каналов

распознавания.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<int>**, содержащий количество каналов распознавания или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getRecognitionChannelsCount",
  "params": {
    "token": ""
  },
  "id": 1
}
```

GetRecognitionChannelById(...)

`TOperationResult<TRecognitionChannel> GetRecognitionChannelById(id, token)`- получение канала распознавания по его идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор канала распознавания
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRecognitionChannel>**, содержащий информация о канале распознавания или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getRecognitionChannelById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

GetPersonCustomFields(...)

`TOperationResult<TPersonCustomField[]> GetPersonCustomFields(personId, token)`- получение списка пользовательских полей сотрудника

Входные данные

Параметр	Тип	Описание
personId	int	Идентификатор сотрудника
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonCustomField[]>**, содержащий список пользовательских полей сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPersonCustomFields",
  "params": {
    "personId": 2,
    "token": ""
  },
  "id": 1
}
```

SetPersonCustomFieldValue(...)

`TOperationResult<TPersonCustomField> SetPersonCustomFieldValue(personId, field, token)`- редактирование значения пользовательского поля сотрудника

Входные данные

Параметр	Тип	Описание
personId	int	Идентификатор сотрудника
field	TPersonCustomField	Информация о пользовательском поле сотрудника
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TPersonCustomField>**, содержащий информацию о пользовательском поле сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "setPersonCustomFieldValue",
```

```
  "params": {
    "personId": 3,
    "field": {
      "id": 1,
      "value": "test_field"
    },
    "token": ""
  },
  "id": 1
}
```

GetAbsenceReasonTypes(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReasonType[]> GetAbsenceReasonTypes(token)` - получение списка уважительных причин в системе.

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReasonType[]>**, содержащий список уважительных причин в системе или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAbsenceReasonTypes",
  "params": {
    "token": ""
  },
  "id": 1
}
```

GetAbsenceReasonTypeById(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReasonType> GetAbsenceReasonTypeById(id, token)`- получение уважительной причины по ее идентификатору.

Входные данные

Параметр	Тип	Описание
id	int	Идентификатор уважительной причины
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReasonType>**, содержащий информацию об уважительной причине или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAbsenceReasonTypeById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

AddAbsenceReasonType(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReasonType> AddAbsenceReasonType(absenceReasonType, token)` - добавление уважительной причины.

Входные данные

Параметр	Тип	Описание
absenceReasonType	TAbsenceReasonType	Информация о добавляемой уважительной причине
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReasonType>**, содержащий информацию о добавленной уважительной причине или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addAbsenceReasonType",
  "params": {
    "absenceReasonType": {
      "id": 0,
      "name": "test"
    },
    "token": ""
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult":       {
         "id": 2,
         "name": "test"
      }
   }
}
```

DeleteAbsenceReasonType(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReasonType> DeleteAbsenceReasonType(absenceReasonType, token)` - удаление уважительной причины.

Входные данные

Параметр	Тип	Описание
absenceReasonType	TAbsenceReasonType	Информация об удаляемой уважительной причине
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReasonType>**, содержащий информацию об удаленной уважительной причине или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteAbsenceReasonType",
  "params": {
    "absenceReasonType": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```

GetAbsenceReasons(…)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReasons> GetAbsenceReasons(person, token)` - получение списка уважительных причин сотрудника.

Входные данные

Параметр	Тип	Описание

person	TPersonData	Информация о сотруднике, для которого запрашивается список уважительных причин отсутствия
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReason>**, содержащий список уважительных причин отсутствия сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getAbsenceReasons",
  "params": {
    "person": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```

AddAbsenceReason(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReason> AddAbsenceReason(absenceReason, token)`- добавление уважительной причины отсутствия сотрудника.

Входные данные

Параметр	Тип	Описание
absenceReason	TAbsenceReason	Информация о добавляемой уважительной причине отсутствия сотрудника
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReason>**, содержащий добавленную уважительную причину отсутствия сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addAbsenceReason",
  "params": {
    "absenceReason": {
      "id": 0,
      "personId": 1,
      "startDate": "2024-10-30T10:28:18",
```

```
      "endDate": "2025-12-30T22:40:18",
      "reasonTypeId": 2,
      "comment": "",
      "startDate": "2024-10-30T10:28:18",
      "endDate": "2025-12-30T22:40:18",
      "operator": "",
      "isValidReason": true,
      "isIgnoreInCalc": false
    },
    "token": "?"
  },
  "id": 1
}
```


DeleteAbsenceReason(...)

Поддерживается в АРМ «Орион Про» начиная с версии 1.20.3.

`TOperationResult<TAbsenceReason> DeleteAbsenceReason(absenceReason, token)`- удаление уважительной причины отсутствия сотрудника.

Входные данные

Параметр	Тип	Описание
absenceReason	TAbsenceReason	Информация об удаляемой уважительной причине отсутствия сотрудника
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TAbsenceReason>**, содержащий удаленную уважительную причину отсутствия сотрудника или информацию об ошибке.

Пример запроса JSON-RPC:


```
{
  "jsonrpc": "2.0",
  "method": "deleteAbsenceReason",
  "params": {
    "absenceReason": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```


AddExternalEvent(...)

`TOperationResult<TExternalEvent> AddExternalEvent(externalEvent, token)`- добавление события.

Входные данные

Параметр	Тип	Описание

externalEvent	TExternalEvent	Информация о добавляемом событии
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TExternalEvent>**, содержащий информацию о добавленном событии или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "addExternalEvent",
  "params": {
    "externalEvent": {
      "id": 0,
      "itemId": 1,
      "itemType": "ACCESSPOINT",
      "event":31,
      "timestamp": "2025-03-01T12:00:00Z",
      "personId": 1,
      "keyId": 9
      "text": "",
      "accessMode": 0
    },
    "token": ""
  },
  "id": 1
}
```

GetBioPrintFromReader(...)

`TOperationResult<string> GetBioPrintFromReader(deviceId, printType, token)`- чтение отпечатка с биометрического считывателя.

Входные данные

Параметр	Тип	Описание
deviceId	int	Физический идентификатор устройства (хранится в устройстве)
printType	string	Тип отпечатка: ”FINGER” - отпечаток пальца; “FACE” - шаблон лица; “PALM” - шаблон ладони;
token	string	Токен авторизованного пользователя

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий данные отпечатка или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getBioPrintFromReader",
  "params": {
    "deviceId": 0,
    "printType": "FINGER",
    "token": ""
  },
  "id": 1
}
```

CancelBioPrintReading(…)

`TOperationResult CancelBioPrintReading(deviceId, printType, token)` - отмена чтения отпечатка с биометрического считывателя.

Входные данные

Параметр	Тип	Описание
deviceId	int	Физический идентификатор устройства (хранится в устройстве)
printType	string	Тип отпечатка: ”FINGER” - отпечаток пальца; “FACE” - шаблон лица; “PALM” - шаблон ладони;
token	string	Токен авторизованного пользователя

Результирующие данные

Метод возвращает объект класса **TOperationResult**, содержащий информацию об успешном выполнении или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "cancelBioPrintReading",
  "params": {
    "deviceId": 0,
    "printType": "FINGER",
    "token": ""
  },
  "id": 1
}
```

ConvertWiegandToTouchMemory(...)

`TOperationResultString ConvertWiegandToTouchMemory(code, codeSize)` - конвертация кода карты из формата Wiegand в формат Touch Memory.

Входные данные

Параметр	Тип	Описание
code	long	Код карты в формате Wiegand (значение в десятичном

		формате)
codeSize	int	Размер кода Wiegand в байтах

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий код в формате Touch Memory или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "convertWiegandToTouchMemory",
  "params": {
    "code": 6957002,
    "codeSize": 4
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "4F0000006A27CA01"
   }
}
```

ConvertPinToTouchMemory(...)

`TOperationResultString ConvertPinToTouchMemory(pin)` - конвертация PIN кода в формат Touch Memory.

Входные данные

Параметр	Тип	Описание
pin	string	PIN код

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий код в формате Touch Memory или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "convertPinToTouchMemory",
```

```
  "params": {
    "pin": "4321"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "EFFFFFFFFF4321F0"
   }
}
```

ConvertImageToBioTemplate(...)

`TOperationResultString ConvertImageToBioTemplate(image)` - конвертация фото в био шаблон.

Входные данные

Параметр	Тип	Описание
image	byte[]	Фото

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий данные био шаблона или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "convertImageToBioTemplate",
  "params": {
    "image": ""
  },
  "id": 1
}
```

GetPinTouchMemory(...)

`TOperationResultString GetPinFromTouchMemory(code)` - получение PIN кода из данных кода в формате Touch Memory.

Входные данные

Параметр	Тип	Описание
code	string	Код в формате Touch Memory

Результирующие данные

Метод возвращает объект класса **TOperationResult<string>**, содержащий PIN код или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getPinFromTouchMemory",
  "params": {
    "code": "EFFFFFFFFF4321F0"
  },
  "id": 1
}
```

Пример результата запроса JSON-RPC:

```
{
   "jsonrpc": "2.0",
   "id": 1,
   "result":    {
      "success": true,
      "operationResult": "4321"
   }
}
```

GetChangedPersons(...)

`TOperationResultDbChangeLogItems GetChangedPersons(beginTime, endTime)` - получение журнала изменений данных сотрудников.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении данных сотрудников или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedPersons",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "token": "?"
  },
  "id": 1
}
```

GetChangedEntryPoints(...)

`TOperationResultDbChangeLogItems GetChangedEntryPoints(beginTime, endTime)`- получение журнала изменений данных точек доступа.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении данных точек доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedEntryPoints",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "token": "?"
  },
  "id": 1
}
```

GetChangedKeys(...)

`TOperationResultDbChangeLogItems GetChangedKeys(beginTime, endTime)`- получение журнала изменений данных ключей.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении данных ключей или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedKeys",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
```

```
    "endTime": "2025-12-30T22:40:18",
    "token": "?"
  },
  "id": 1
```

GetChangedAccessLevels(...)

`TOperationResultDbChangeLogItems GetChangedAccessLevels(beginTime, endTime)`- получение журнала изменений уровней доступа.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении уровней доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedAccessLevels",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "token": ""
  },
  "id": 1
}
```

GetChangedAccessLevelItems(...)

`TOperationResultDbChangeLogItems GetChangedAccessLevelItems(beginTime, endTime)`- получение журнала изменений полномочий уровней доступа.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении полномочий уровней доступа или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedAccessLevelItems",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "token": ""
  },
  "id": 1
}
```

GetChangedTimeWindows(...)

`TOperationResultDbChangeLogItems GetChangedTimeWindows(beginTime, endTime)`- получение журнала изменений окон времени.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении окон времени или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedTimeWindows",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "token": ""
  },
  "id": 1
}
```

GetChangedTimeIntervals(...)

`TOperationResultDbChangeLogItems GetChangedTimeIntervals(beginTime, endTime)`- получение журнала изменений временных интервалов окон времени.

Входные данные

Параметр	Тип	Описание
beginTime	DateTime	Дата и время начала для фильтрации
endTime	DateTime	Дата и время конца для фильтрации

token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**).

Результирующие данные

Метод возвращает объект класса **`TOperationResult<TDbChangeLogItem[]>`**, содержащий информацию об изменении временных интервалов окон времени или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getChangedTimeIntervals",
  "params": {
    "beginTime": "2024-10-30T10:28:18",
    "endTime": "2025-12-30T22:40:18",
    "token": ""
  },
  "id": 1
}
```

GetRooms(...)

`TOperationResult<TRoom[]> GetRooms(token)` – получение списка комнат из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRoom[]>**, содержащий список комнат или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getRooms",
  "params": {
    "token": ""
  },
  "id": 1
}
```

SearchRoom(...)

`TOperationResult<TRoom[]> SearchRoom(name, token)` – получение отфильтрованного списка комнат из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
name	string	Подстрока для фильтрации списка комнат
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRoom[]>**, содержащий список комнат с учетом фильтра или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "searchRoom",
  "params": {
    "name": "",
    "token": ""
  },
  "id": 1
}
```

GetRoomById(...)

`TOperationResult<TRoom> GetRoomById(id, token)` – получение информации о комнате из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
Id	int	Идентификатор комнаты
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRoom>**, содержащий информацию о комнате или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "getRoomById",
  "params": {
    "id": 1,
    "token": ""
  },
  "id": 1
}
```

CreateRoom(...)

`TOperationResult<TRoom> CreateRoom(Room, token)` – добавление комнаты в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
room	TRoom	Информация о создаваемой комнате
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRoom>**, содержащий информацию о созданной комнате или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "createRoom",
  "params": {
    "room": {
      "id": 0,
      "name": "test_room"
    },
    "token": ""
  },
  "id": 1
}
```

UpdateRoom(...)

`TOperationResult<TRoom> UpdateRoom(Room, token)` – редактирование комнаты в АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
room	TRoom	Информация об обновляемой комнате
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRoom>**, содержащий информацию об обновленной комнате или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "updateRoom",
  "params": {
    "room": {
      "id": 1,
      "name": "test_room"
```

```
    },
    "token": ""
  },
  "id": 1
}
```

DeleteRoom(...)

`TOperationResult<TRoom> DeleteRoom(Room, token)` – удаление комнаты из АРМ «Орион Про».

Входные данные

Параметр	Тип	Описание
room	TRoom	Информация об удаляемой комнате
token	string	Токен авторизованного пользователя (используется при включенной опции **IsTokenRequired**)

Результирующие данные

Метод возвращает объект класса **TOperationResult<TRoom>**, содержащий информацию об удаленной комнате или информацию об ошибке.

Пример запроса JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "deleteRoom",
  "params": {
    "room": {
      "id": 1
    },
    "token": ""
  },
  "id": 1
}
```