



**Описание службы интеграции
Системы Контроля Доступа Parsec
(ПО PNSoft и PNOffice)**

версия от 13.01.2023

ОГЛАВЛЕНИЕ

| | |
|---|-----------|
| 1. ВВЕДЕНИЕ | 7 |
| Общие положения..... | 7 |
| Принцип работы сервиса | 7 |
| Функционал клиентского приложения | 7 |
| Некоторые рекомендации разработчикам | 8 |
| Состав комплекта разработчика..... | 9 |
| Поддержка функций интеграционного сервиса продуктами PNSoft и PNOffice | 9 |
| 2. ИСПОЛЬЗУЕМЫЕ ОБЪЕКТЫ | 14 |
| BaseResult | 14 |
| GuidResult | 14 |
| SessionResult | 14 |
| StringResult | 14 |
| ObjectResult..... | 15 |
| EventsHistoryResult | 15 |
| Session..... | 15 |
| BaseObject..... | 15 |
| BaseOrgUnit..... | 15 |
| OrgUnit | 16 |
| BasePerson | 16 |
| Person | 16 |
| PersonWithPhoto..... | 16 |
| PersonExtraFieldTemplate | 17 |
| PersonScheduleFix | 17 |
| ExtraFieldValue..... | 17 |
| VisitorRequest | 18 |
| Schedule..... | 18 |
| AccessSchedule | 18 |
| WorktimeSchedule | 19 |
| ScheduleDay | 19 |
| ScheduleFix..... | 19 |
| TimeInterval | 20 |
| WorktimeInterval..... | 20 |
| Holiday | 20 |
| BaseIdentifier | 20 |
| Identifier..... | 21 |
| IdentifierTemp..... | 21 |
| StockIdentifier..... | 21 |
| IdentifierExData..... | 22 |
| PassageRole | 22 |
| BaseTerritory | 22 |
| Territory | 23 |
| TerritoryWithComponent..... | 23 |
| AccessGroup | 24 |
| SubAccessGroup | 24 |
| Event | 24 |
| EventsHistory | 24 |
| EventObject | 25 |
| Domain..... | 25 |
| EventHistoryQueryParams..... | 25 |

| | |
|--|-----------|
| EventFilter | 27 |
| AdvancedEventFilter | 27 |
| HardwareState | 27 |
| TransactionClass | 27 |
| TransactionType | 28 |
| QRAdvancedGroup | 28 |
| QRAdvancedData | 28 |
| ВЕРСИЯ | 29 |
| Функция GetVersion | 29 |
| СЕССИЯ | 29 |
| Функция GetDomains..... | 29 |
| Функция OpenSession | 29 |
| Функция OpenSessionWithInLocale | 29 |
| Функция ContinueSession | 30 |
| Функция CloseSession..... | 30 |
| Функция CheckRole | 30 |
| Функция GetObjectName..... | 31 |
| ОБОРУДОВАНИЕ | 31 |
| Функция SendHardwareCommand | 31 |
| Функция SendVerificationCommand | 32 |
| Функция GetHardwareState | 32 |
| ПОДРАЗДЕЛЕНИЯ..... | 34 |
| Функция GetRootOrgUnit..... | 34 |
| Функция GetOrgUnitsHierarhy..... | 34 |
| Функция GetOrgUnitsHierarhyWithPersons | 34 |
| Функция GetOrgUnitsHierarhyWithVisitors | 34 |
| Функция GetOrgUnitsHierarhyWithVehicle | 34 |
| Функция GetOrgUnitSubItems..... | 35 |
| Функция GetOrgUnitSubItemsHierarhyWithPersons | 35 |
| Функция GetOrgUnitSubItemsHierarhyWithVisitors | 35 |
| Функция GetOrgUnitSubItemsHierarhyWithVehicle | 35 |
| Функция GetOrgUnit | 36 |
| РЕДАКТИРОВАНИЕ ПОДРАЗДЕЛЕНИЯ..... | 36 |
| Функция CreateOrgUnit..... | 36 |
| Функция OpenOrgUnitEditingSession | 36 |
| Функция CloseOrgUnitEditingSession | 36 |
| Функция SaveOrgUnit | 37 |
| Функция DeleteOrgUnit | 37 |
| ПЕРСОНАЛ | 37 |
| Функция GetPersonExtraFieldTemplates | 37 |
| Функция GetVisitorExtraFieldTemplates..... | 37 |
| Функция GetVehicleExtraFieldTemplates..... | 38 |
| Функция FindPeople..... | 38 |
| Функция FindVisitors..... | 38 |
| Функция FindVehicle | 38 |
| Функция FindPersonByIdentifier | 39 |
| Функция PersonSearch..... | 39 |
| Функция GetPerson | 40 |

| | |
|--|-----------|
| Функция GetMultiplePersons | 41 |
| Функция GetPersonsChangedAfter | 41 |
| Функция GetPersonExtraFieldValue | 41 |
| Функция GetPersonExtraFieldValues | 41 |
| Функция GetPersonExtraFieldValueString | 42 |
| Функция ValidateExtraFieldValue | 42 |
| Функция GetPersonScheduleFixes..... | 42 |
| Функция AddPersonScheduleFix | 42 |
| Функция SavePersonScheduleFix | 43 |
| Функция DeletePersonScheduleFix | 43 |
| Функция GetPersonWorktimeSchedule | 43 |
| Функция SetPersonWorktimeSchedule | 43 |
| Функция GetBlackList..... | 44 |
| Функция FindInBlackList..... | 44 |
| Функция AddToBlackList..... | 44 |
| Функция RemoveFromBlackList | 44 |
| РЕДАКТИРОВАНИЕ СУБЪЕКТА ДОСТУПА..... | 45 |
| Функция CreatePerson | 45 |
| Функция CreateVisitor | 45 |
| Функция CreateVehicle..... | 45 |
| Функция OpenPersonEditingSession | 45 |
| Функция ClosePersonEditingSession..... | 46 |
| Функция SavePerson | 46 |
| Функция SetPersonPhoto | 46 |
| Функция SetPersonOrgUnit | 47 |
| Функция SetPersonExtraFieldValue | 47 |
| Функция SetPersonExtraFieldValues..... | 48 |
| Функция DeletePerson | 48 |
| Функция BlockPerson | 48 |
| Функция UnblockPerson | 48 |
| ТОПОЛОГИЯ..... | 49 |
| Функция GetRootTerritory | 49 |
| Функция GetTerritoriesHierarhy..... | 49 |
| Функция GetTerritorySubItems | 49 |
| Функция GetTerritory..... | 49 |
| ИДЕНТИФИКАТОРЫ И ДОСТУП..... | 50 |
| Функция GetPersonIdentifiers | 50 |
| Функция DeleteIdentifier | 50 |
| Функция AddPersonIdentifier | 50 |
| Функция ChangePersonIdentifier | 51 |
| Функция SetIdentifierPrivileges | 52 |
| Функция GetIdentifierExtraData | 53 |
| Функция SetIdentifierExtraData | 53 |
| Функция GetPassageRoles | 53 |
| Функция CreatePassageRole | 53 |
| Функция SavePassageRole..... | 54 |
| Функция DeletePassageRole | 54 |
| Функция GetUnique4bCardCode | 54 |
| Функция GetCardCodeFromUID | 54 |

| | |
|---|-----------|
| Функция GenerateParsecQRCode | 55 |
| Функция GetAdvancedQRGroups | 55 |
| Функция GenerateAdvancedParsecQRCode | 55 |
| РАСПИСАНИЯ И ГРУППЫ ДОСТУПА | 55 |
| Функция GetAccessSchedules | 55 |
| Функция GetWorktimeSchedules | 56 |
| Функция GetScheduleIntervals | 56 |
| Функция CreateAccessSchedule | 56 |
| Функция CreateWorktimeSchedule | 57 |
| Функция GetSchedule | 57 |
| Функция SaveSchedule | 57 |
| Функция DeleteSchedule | 58 |
| Функция GetScheduleDetails | 58 |
| Функция SetScheduleDays | 58 |
| Функция SetScheduleFix | 58 |
| Функция DeleteScheduleDays | 59 |
| Функция GetHolidays | 59 |
| Функция SetHolidays | 59 |
| Функция DeleteHolidays | 59 |
| Функция GetAccessGroups | 60 |
| Функция CreateTempAccessGroup | 60 |
| Функция CreateAccessGroup | 60 |
| Функция CreateVehicleTempAccessGroup | 60 |
| Функция CreateVehicleAccessGroup | 61 |
| Функция DeleteAccessGroup | 61 |
| Функция AddSubAccessGroup | 61 |
| Функция DeleteSubAccessGroup | 62 |
| Функция GetSubAccessGroups | 62 |
| Функция GetInheritedAccessGroups | 62 |
| Функция SetInheritedAccessGroups | 62 |
| РАБОТА С ЗАЯВКАМИ БЮРО ПРОПУСКОВ | 63 |
| Функция GetAcceptedVisitorRequests | 63 |
| Функция FindVisitorRequest | 63 |
| Функция ActivateVisitorRequest | 63 |
| Функция CreateVisitorRequest | 64 |
| Функция GetVisitorRequest | 64 |
| Функция SaveVisitorRequest | 64 |
| Функция DeleteIssuedVisitorRequest | 64 |
| Функция GetIssuedVisitorRequests | 65 |
| Функция GetVisitorRequests | 65 |
| Функция CloseAllActiveVisitorRequests | 65 |
| Функция CloseVisitorRequest | 65 |
| Функция GetPersonVisitorRequests | 66 |
| СОБЫТИЯ СИСТЕМЫ | 66 |
| Функция GetEvents | 66 |
| Функция OpenEventHistorySession | 67 |
| Функция CloseEventHistorySession | 67 |
| Функция GetEventHistoryResultCount | 67 |
| Функция GetEventHistoryResult | 67 |

| | |
|---|-----------|
| Функция GetHardwareEvents | 69 |
| Функция GetHardwareEventsResolved | 72 |
| Функция EventsSubscribe | 74 |
| Функция EventsSubscribeCheck | 75 |
| Функция EventsUnsubscribe | 75 |
| Функция EventsUnsubscribeUrl | 76 |
| Функция GetTransactionClasses | 76 |
| Функция GetTransactionTypes | 76 |
| 3. ИСТОРИЯ ИЗМЕНЕНИЙ..... | 77 |
| 4. АЛФАВИТНЫЙ УКАЗАТЕЛЬ | 81 |

1. ВВЕДЕНИЕ

Общие положения

Данный документ описывает функционал интеграционного сервиса (далее просто «сервис») профессиональной системы контроля и управления доступом (СКУД) ParsecNET.

Сервис позволяет сторонним программным комплексам получить доступ к данным и событиям системы ParsecNET для реализации специфического функционала, отсутствующего в системе. Сервис устанавливается автоматически на сервере системы ParsecNET и входит в состав базовой версии ПО.

Для работы сервиса необходимо наличие лицензионного ключа защиты, однако, достаточно наличия базовой лицензии на ПО. Функциональность сервиса интеграции дополнительно не лицензируется.

Принцип работы сервиса

Сервис работает на компьютере, являющемся сервером системы ParsecNET, как служба Windows. Он реализован как XML/SOAP WEB-сервис, что позволяет обращаться к нему с любого компьютера в сети, а не только с компьютера-сервера системы. Соответственно, внешнее приложение может быть создано в любой среде разработки, поддерживающей технологию WEB-сервисов и, в частности, SOAP. Сервис обслуживается приложением MDO.Parsec.ParsecIntegrationServiceHost.exe.

Адрес размещения сервиса на сервере системы указывается в конфигурационном файле MDO.Parsec.ParsecIntegrationServiceHost.exe.config (по-умолчанию:
<http://localhost:10101/IntegrationService/IntegrationService.asmx>).

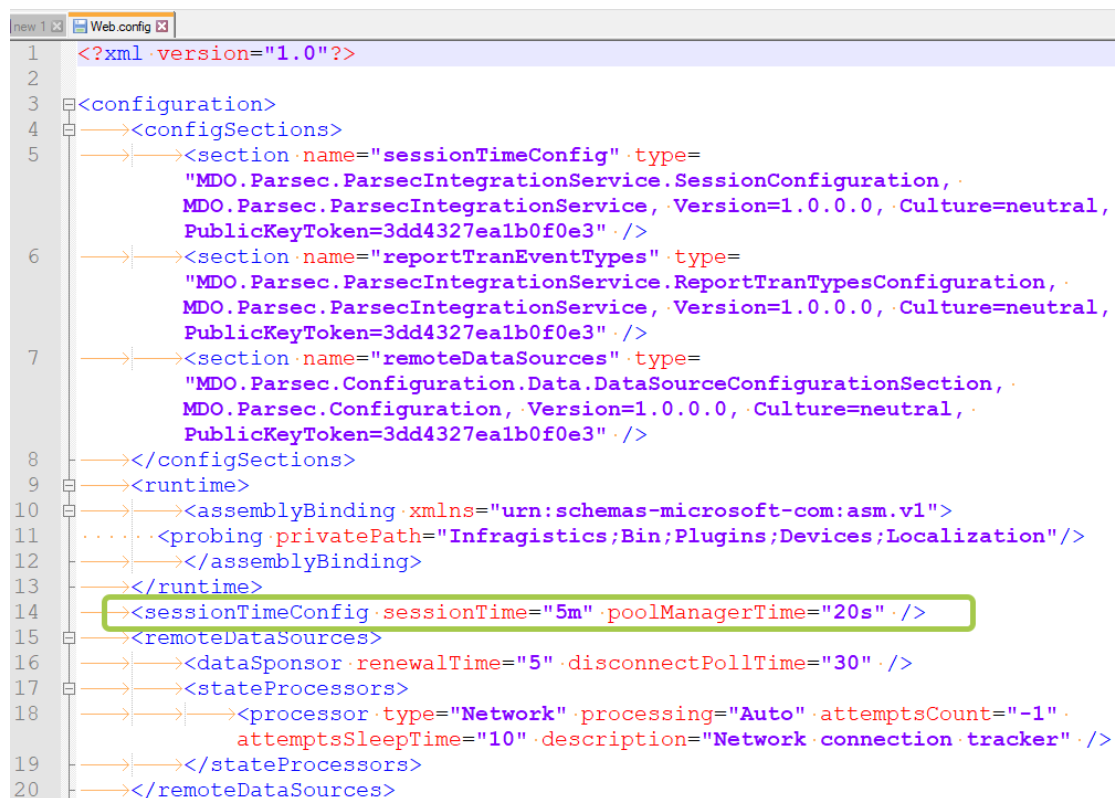
К документу WSDL можно получить доступ по адресу:
<http://localhost:10101/IntegrationService/IntegrationService.asmx?wsdl>

Функционал клиентского приложения

Клиентское приложение по правам эквивалентно оператору одной конкретно взятой организации системы. Для получения доступа к сервису в системе необходимо создать оператора с необходимыми правами и областями видимости. Результат выполнения функций интеграционного сервиса определяется правами оператора в системе Parsec, с которыми открыта сессия.

Работа с сервисом идет на уровне сессий: клиентское приложение после логина открывает сессию и в рамках этой сессии использует функции сервиса. Сервис автоматически закроет сессию, если более пяти минут клиент не использовал обращений к Функциям сервиса. Сессия автоматически продлевается при вызове любой функции с указанием ID сессии.

Таймаут закрытия сессии можно изменить в конфигурационном файле Web.config, расположенном в папке установки сервера ParsecNET 3/Parsec Office, в следующем месте:



```

1 <?xml version="1.0"?>
2
3 <configuration>
4   <configSections>
5     <section name="sessionTimeConfig" type=
      "MDO.Parsec.ParsecIntegrationService.SessionConfiguration,
      MDO.Parsec.ParsecIntegrationService, Version=1.0.0.0, Culture=neutral,
      PublicKeyToken=3dd4327ealb0f0e3" />
6     <section name="reportTranEventTypes" type=
      "MDO.Parsec.ParsecIntegrationService.ReportTranTypesConfiguration,
      MDO.Parsec.ParsecIntegrationService, Version=1.0.0.0, Culture=neutral,
      PublicKeyToken=3dd4327ealb0f0e3" />
7     <section name="remoteDataSources" type=
      "MDO.Parsec.Configuration.Data.DataSourceConfigurationSection,
      MDO.Parsec.Configuration, Version=1.0.0.0, Culture=neutral,
      PublicKeyToken=3dd4327ealb0f0e3" />
8   </configSections>
9   <runtime>
10    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
11      <probing privatePath="Infragistics;Bin;Plugins;Devices;Localization"/>
12    </assemblyBinding>
13  </runtime>
14  <sessionTimeConfig sessionTime="5m" poolManagerTime="20s" />
15  <remoteDataSources>
16    <dataSponsor renewalTime="5" disconnectPollTime="30" />
17    <stateProcessors>
18      <processor type="Network" processing="Auto" attemptsCount="-1"
        attemptsSleepTime="10" description="Network connection tracker" />
19    </stateProcessors>
20  </remoteDataSources>

```

Клиентское приложение с помощью сервиса может получить доступ к структуре персонала системы ParsecNET, включая всю иерархию подразделений в рамках конкретной организации.

Сервис предоставляет достаточно функций для того, чтобы можно было создать, отредактировать или удалить пользователя системы ParsecNET. При этом, как и при работе с приложениями системы, любое сделанное изменение без участия клиента реплицируется на все объекты системы, включая контроллеры.

Сервис позволяет создавать как временные, так и обычные (постоянные) группы доступа. Сервис может предоставить полный список групп доступа, чтобы назначить персоналу в клиентском приложении необходимые права доступа к территориям объекта.

Кроме работы с персоналом сервис дает доступ к событиям системы.

Каждый получаемый из системы объект сопровождается его уникальным ключом, который необходим для идентификации этого объекта. Ключи имеют тип GUID – статистически уникальный 128-битный идентификатор.

Некоторые рекомендации разработчикам

Для корректной работы интеграционного сервиса в ОС Windows должен быть включен компонент ASP.NET 4.x.

Данные из системы можно получать с использованием разных функций сервиса. Разработчик сам выбирает стратегию получения данных в зависимости от стоящих перед ним задач.

Однако следует иметь в виду, что некоторые функции могут в крупных системах возвращать очень большие массивы данных (в зависимости от масштаба самой системы). Соответственно, операция может выполняться достаточно долго и потреблять необоснованно много ресурсов.

Поэтому предпочтительно сначала получать общую иерархию сущности (например, дерево подразделений), а затем получать данные о персонале не для

всей системы сразу, а для подразделения, с которым клиентское приложение реально в настоящее время работает.

Состав комплекта разработчика

В комплект разработчика входят следующие компоненты:

- Данное руководство;
- Пример клиентского приложения на языке C# с исходными текстами и работающим откомпилированным примером;
- Пример клиентского приложения на языке Object Pascal с исходными текстами и работающим откомпилированным примером (среда разработки Delphi, Borland Developer Studio 2007. Более ранние версии Delphi не обеспечивают в полной мере поддержки требуемых технологий);

Поддержка функций интеграционного сервиса продуктами PNSoft и PNOffice

Интеграционный сервис, входящий в комплект программного обеспечения PNOffice (ParsecNET Office) по причине отсутствия части функционала, например, бюро пропусков, поддерживает подмножество описанных в данном документе функций.

В таблице ниже приведены данные по совместимости функций в программных продуктах PNSoft и PNOffice.

Также для удобства функции в таблице сгруппированы в разделы в соответствии с функционалом, который они предоставляют разработчику.

| PNSoft | PNOffice |
|--|-------------------|
| Работа с сессией | |
| CloseSession | |
| ContinueSession | |
| OpenSession | |
| OpenSessionWithInLocale | |
| Работа с группами доступа | |
| AddSubAccessGroup | не поддерживается |
| CreateAccessGroup | не поддерживается |
| CreateVehicleTempAccessGroup | не поддерживается |
| CreateVehicleAccessGroup | не поддерживается |
| CreateTempAccessGroup | |
| DeleteAccessGroup | не поддерживается |
| DeleteSubAccessGroup | не поддерживается |
| GetAccessGroups | |
| Наследование групп доступа | |
| GetInheritedAccessGroups | не поддерживается |
| SetInheritedAccessGroups | не поддерживается |
| GetSubAccessGroups | не поддерживается |
| Роли группового прохода | |
| CreatePassageRole | не поддерживается |
| SavePassageRole | не поддерживается |
| DeletePassageRole | не поддерживается |
| GetPassageRoles | не поддерживается |

| PNSoft | PNOffice |
|--|-------------------|
| Получение событий системы из архива БД | |
| OpenEventHistorySession | |
| GetEventHistoryResult | |
| GetEventHistoryResultCount | |
| CloseEventHistorySession | |
| GetEvents (устаревшая, не рекомендуется к использованию) | |
| Получение оперативных событий и статусов контроллеров доступа | |
| GetHardwareEvents | не поддерживается |
| GetHardwareEventsResolved | не поддерживается |
| EventsSubscribe | не поддерживается |
| EventsSubscribeCheck | не поддерживается |
| EventsUnsubscribe | не поддерживается |
| EventsUnsubscribeUrl | не поддерживается |
| GetHardwareState | не поддерживается |
| Получение классов и типов событий, генерируемых системой | |
| GetTransactionClasses | не поддерживается |
| GetTransactionTypes | не поддерживается |
| Команды прямого управления контроллерами доступа | |
| SendHardwareCommand | не поддерживается |
| SendVerificationCommand | не поддерживается |
| Работа с подразделениями, сотрудниками, транспортными средствами и идентификаторами | |
| <i>Подразделения</i> | |
| OpenOrgUnitEditingSession | |
| CloseOrgUnitEditingSession | |
| CreateOrgUnit | |
| SaveOrgUnit | |
| GetOrgUnit | |
| DeleteOrgUnit | |
| GetOrgUnitSubItems | |
| GetOrgUnitsHierarchy | |
| GetRootOrgUnit | |
| <i>Сотрудники и идентификаторы</i> | |
| OpenPersonEditingSession | |
| ClosePersonEditingSession | |
| AddPersonIdentifier | |
| ChangePersonIdentifier | |
| GetPersonIdentifiers | |
| GetIdentifierExtraData | не поддерживается |
| SetIdentifierExtraData | не поддерживается |
| CreatePerson | |
| SavePerson | |
| DeletePerson | |
| DeleteIdentifier | |
| GetOrgUnitsHierarchyWithPersons | |

| PNSoft | PNOffice |
|---|-------------------|
| GetOrgUnitSubItemsHierarhyWithPersons | |
| GetPerson | |
| GetMultiplePersons | не поддерживается |
| SetPersonOrgUnit | |
| GetPersonWorktimeSchedule | |
| SetPersonWorktimeSchedule | |
| GetBlackList | не поддерживается |
| FindInBlackList | не поддерживается |
| AddToBlackList | не поддерживается |
| RemoveFromBlackList | не поддерживается |
| SetPersonPhoto | |
| SetIdentifierPrivileges | не поддерживается |
| Поиск субъектов доступа | |
| FindPeople | |
| FindPersonByIdentifier | не поддерживается |
| FindVehicle | не поддерживается |
| PersonSearch | не поддерживается |
| GetPersonsChangedAfter | |
| Транспортные средства | |
| CreateVehicle | не поддерживается |
| GetOrgUnitsHierarhyWithVehicle | не поддерживается |
| GetOrgUnitSubItemsHierarhyWithVehicle | не поддерживается |
| Генерация QR-кодов и работа с кодами идентификаторов | |
| GenerateParsecQRCode | не поддерживается |
| GetCardCodeFromUID | не поддерживается |
| GetUnique4bCardCode | не поддерживается |
| GetAdvancedQRGroups | не поддерживается |
| GenerateAdvancedParsecQRCode | не поддерживается |
| Дополнительные поля субъекта доступа | |
| GetPersonExtraFieldTemplates | |
| GetPersonExtraFieldValue | |
| GetPersonExtraFieldValueString | |
| GetPersonExtraFieldValues | |
| SetPersonExtraFieldValue | |
| SetPersonExtraFieldValues | |
| ValidateExtraFieldValue | |
| GetVehicleExtraFieldTemplates | не поддерживается |
| Блокировка/разблокировка доступа | |
| BlockPerson | |
| UnblockPerson | |
| Работа с топологией | |
| GetRootTerritory | |
| GetTerritoriesHierarhy | |
| GetTerritory | |
| GetTerritorySubItems | |
| Расписания, праздники | |

| PNSoft | PNOffice |
|---|-------------------|
| CreateAccessSchedule | не поддерживается |
| CreateWorktimeSchedule | не поддерживается |
| DeleteSchedule | не поддерживается |
| DeleteScheduleDays | не поддерживается |
| GetAccessSchedules | |
| GetWorktimeSchedules | |
| GetSchedule | не поддерживается |
| GetScheduleDetails | не поддерживается |
| GetScheduleIntervals | |
| SaveSchedule | не поддерживается |
| SetScheduleDays | не поддерживается |
| SetScheduleFix | не поддерживается |
| Праздники | |
| GetHolidays | не поддерживается |
| SetHolidays | не поддерживается |
| DeleteHolidays | не поддерживается |
| Поправки рабочего времени | |
| AddPersonScheduleFix | не поддерживается |
| DeletePersonScheduleFix | не поддерживается |
| GetPersonScheduleFixes | не поддерживается |
| SavePersonScheduleFix | не поддерживается |
| Бюро пропусков, посетители | |
| ActivateVisitorRequest | не поддерживается |
| CloseAllActiveVisitorRequests | не поддерживается |
| CloseVisitorRequest | не поддерживается |
| CreateVisitor | не поддерживается |
| CreateVisitorRequest | не поддерживается |
| DeleteIssuedVisitorRequest | не поддерживается |
| FindVisitors | не поддерживается |
| FindVisitorRequest | не поддерживается |
| GetAcceptedVisitorRequests | не поддерживается |
| GetIssuedVisitorRequests | не поддерживается |
| GetPersonVisitorRequests | не поддерживается |
| GetVisitorRequest | не поддерживается |
| SaveVisitorRequest | не поддерживается |
| GetOrgUnitsHierarchyWithVisitors | не поддерживается |
| GetOrgUnitSubItemsHierarchyWithVisitors | не поддерживается |
| GetVisitorExtraFieldTemplates | не поддерживается |
| Безопасность | |
| CheckRole | не поддерживается |
| GetDomains | |
| Получение версии SDK (версии ПО) | |
| GetVersion | |
| Другое | |
| GetObjectName | не поддерживается |

2. ИСПОЛЬЗУЕМЫЕ ОБЪЕКТЫ

BaseResult

Базовый класс, используется в качестве результата исполнения операции.

| | |
|----------------------------------|--|
| <code>int</code> Result | Результат выполнения операции. |
| <code>string</code> ErrorMessage | Описание ошибки, произошедшей при выполнении операции. |

Результат выполнения может быть следующим:

0 – операция выполнена успешно;

-1 – операция выполнена с ошибкой;

Значения >0 планируется использовать для кодов специфических ошибок.

GuidResult

Базовый класс: [BaseResult](#).

Класс используется в качестве результата в функциях, возвращающих Guid.

| | |
|----------------------------------|--|
| <code>int</code> Result | Результат выполнения операции. 0 – операция выполнена успешно, -1 – операция выполнена с ошибкой. Значения >0 планируется использовать для кодов специфических ошибок. |
| <code>string</code> ErrorMessage | Описание ошибки, произошедшей при выполнении операции. |
| <code>Guid</code> Value | Результирующий ID. |

SessionResult

Базовый класс: [BaseResult](#).

Класс используется в качестве результата функции OpenSession.

| | |
|----------------------------------|--|
| <code>int</code> Result | Результат выполнения операции. 0 – операция выполнена успешно, -1 – операция выполнена с ошибкой. Значения >0 планируется использовать для кодов специфических ошибок. |
| <code>string</code> ErrorMessage | Описание ошибки, произошедшей при выполнении операции. |
| <code>Session</code> Value | Результат выполнения операции. |

StringResult

Базовый класс: [BaseResult](#).

Класс используется в качестве результата в функциях, возвращающих String.

| | |
|-------------------------|--|
| <code>int</code> Result | Результат выполнения операции. 0 – операция выполнена успешно, -1 – операция выполнена с ошибкой. Значения >0 планируется использовать для кодов специфических ошибок. |
|-------------------------|--|

| | |
|----------------------------------|--|
| <code>string</code> ErrorMessage | Описание ошибки, произошедшей при выполнении операции. |
| <code>String</code> Value | Результат выполнения операции. |

ObjectResult

Базовый класс: [BaseResult](#).

Класс используется в качестве результата в функциях, возвращающих Object.

| | |
|----------------------------------|--|
| <code>int</code> Result | Результат выполнения операции. 0 – операция выполнена успешно, -1 – операция выполнена с ошибкой. Значения >0 планируется использовать для кодов специфических ошибок. |
| <code>string</code> ErrorMessage | Описание ошибки, произошедшей при выполнении операции. |
| <code>Object</code> Value | Результат выполнения операции. |

EventsHistoryResult

Базовый класс: [BaseResult](#).

Класс используется в качестве результата функции GetEvents.

| | |
|----------------------------------|--|
| <code>int</code> Result | Результат выполнения операции. 0 – операция выполнена успешно, -1 – операция выполнена с ошибкой. Значения >0 планируется использовать для кодов специфических ошибок. |
| <code>string</code> ErrorMessage | Описание ошибки, произошедшей при выполнении операции. |
| <code>EventsHistory</code> Value | Результат выполнения операции. |

Session

Класс, содержащий информацию для сессии.

| | |
|-----------------------------------|--|
| <code>Guid</code> SessionID | Уникальный ключ сессии, используемый при выполнении дальнейших операций. |
| <code>Guid</code> RootOrgUnitID | ID корневого элемента дерева персонала. |
| <code>Guid</code> RootTerritoryID | ID корневого элемента дерева территорий. |

BaseObject

Класс, используемый в качестве базового для основных классов сервиса. Собственных членов не имеет.

BaseOrgUnit

Базовый класс: [BaseObject](#).

Класс, используемый для описания подразделения.

| | |
|----------------------|--------------------------------|
| <code>Guid</code> ID | Уникальный ключ подразделения. |
|----------------------|--------------------------------|

| | |
|--------------------------|-------------------------|
| <code>string</code> NAME | Название подразделения. |
| <code>string</code> DESC | Описание подразделения. |

OrgUnit

Базовый класс: [BaseOrgUnit](#).

Класс, используемый для описания подразделения.

| | |
|-----------------------------|--|
| <code>Guid</code> ID | Уникальный ключ подразделения. |
| <code>string</code> NAME | Название подразделения. |
| <code>string</code> DESC | Описание подразделения. |
| <code>Guid</code> PARENT_ID | Уникальный ключ родительского подразделения. |

BasePerson

Базовый класс: [BaseObject](#).

Класс, используемый для описания сотрудника.

| | |
|---------------------------------|-----------------------------|
| <code>Guid</code> ID | Уникальный ключ сотрудника. |
| <code>string</code> LAST_NAME | Фамилия. |
| <code>string</code> FIRST_NAME | Имя. |
| <code>string</code> MIDDLE_NAME | Отчество. |
| <code>string</code> TAB_NUM | Табельный номер. |

Person

Базовый класс: [BasePerson](#).

Класс, используемый для описания сотрудника.

| | |
|---------------------------------|---|
| <code>Guid</code> ID | Уникальный ключ сотрудника. |
| <code>string</code> LAST_NAME | Фамилия. |
| <code>string</code> FIRST_NAME | Имя. |
| <code>string</code> MIDDLE_NAME | Отчество. |
| <code>string</code> TAB_NUM | Табельный номер. |
| <code>Guid</code> ORG_ID | Уникальный ключ подразделения для сотрудника. |

PersonWithPhoto

Базовый класс: [Person](#).

Класс, используемый для описания сотрудника.

| | |
|--------------------------------|-----------------------------|
| <code>Guid</code> ID | Уникальный ключ сотрудника. |
| <code>string</code> LAST_NAME | Фамилия. |
| <code>string</code> FIRST_NAME | Имя. |

| | |
|---------------------------------|---|
| <code>string</code> MIDDLE_NAME | Отчество. |
| <code>string</code> TAB_NUM | Табельный номер. |
| <code>Guid</code> ORG_ID | Уникальный ключ подразделения для сотрудника. |
| <code>byte[]</code> PHOTO | Фотография сотрудника. |

PersonExtraFieldTemplate

Класс, используемый для определения дополнительного поля данных персонала.

| | |
|-------------------------------|----------------------------|
| <code>Guid</code> ID | Уникальный ключ шаблона. |
| <code>XmlTypeCode</code> TYPE | Тип представленных данных. |
| <code>string</code> NAME | Наименование шаблона. |

PersonScheduleFix

Класс, используемый для определения поправки к рабочему времени сотрудника.

| | |
|-----------------------------|--|
| <code>Guid</code> FIX_ID | Уникальный ключ поправки. |
| <code>Guid</code> PERSON_ID | Уникальный ключ сотрудника. |
| <code>int</code> TYPE_ID | Тип поправки; может принимать значения: 4 – Больничный; 5 – Командировка; 6 – Отпуск; 7 – Принят на работу; 8 – Уволен; 9 – Отпуск без сохранения содержания; 10 – Полный рабочий день; 11 – Присутствие (только в этой поправке может передаваться ненулевое время, остальные поправки должны быть с временем 00:00). |
| <code>DateTime</code> START | Дата и время начала интервала поправки. |
| <code>DateTime</code> END | Дата и время завершения интервала поправки. |
| <code>string</code> COMMENT | Комментарий к поправке рабочего времени. |

ExtraFieldValue

Базовый класс: [BaseObject](#).

Класс, используемый для описания значения дополнительного поля сотрудника.

| | |
|-------------------------------|--------------------------|
| <code>Guid</code> TEMPLATE_ID | Уникальный ключ шаблона. |
| <code>Object</code> VALUE | Значение поля. |

VisitorRequest

Класс, используемый для описания заявок Бюро Пропусков

| | |
|-----------------------------------|---|
| <code>Guid</code> ID | Уникальный ключ заявки. |
| <code>int</code> NUMBER | Уникальный номер заявки. |
| <code>DateTime</code> DATE | Дата создания заявки. |
| <code>Guid</code> ORGUNIT_ID | Уникальный ключ подразделения, для посещения которого создана заявка. |
| <code>Guid</code> PERSON_ID | Уникальный ключ посетителя. |
| <code>string</code> PERSON_INFO | Дополнительная информация о посетителе. |
| <code>string</code> PURPOSE | Цель визита. |
| <code>int</code> STATUS | Статус заявки. |
| <code>DateTime</code> ADMIT_START | Дата начала разрешенного посещения. |
| <code>DateTime</code> ADMIT_END | Дата окончания разрешенного посещения. |

Статус заявки может принимать одно из значений:

- 0 – Ожидание согласования;
- 1 – Согласована (можно выдавать идентификатор посетителю);
- 2 – Отклонена (можно только закрыть заявку);
- 3 – Выдан пропуск (идентификатор на руках у посетителя);
- 4 – Закрыта.

Schedule

Базовый класс: [BaseObject](#).

Класс, используемый для представления расписания системы

| | |
|--------------------------|-----------------------------|
| <code>Guid</code> ID | Уникальный ключ расписания. |
| <code>string</code> NAME | Наименование. |

AccessSchedule

Базовый класс: [Schedule](#).

Класс, используемый для представления расписания доступа.

| | |
|----------------------------------|---|
| <code>Guid</code> ID | Уникальный ключ расписания. |
| <code>string</code> NAME | Наименование. |
| <code>string</code> DESC | Описание расписания. |
| <code>bool</code> IS_WEEK | Признак, является ли расписание недельным. |
| <code>int</code> HOLIDAYS_ACTION | Тип применения праздничных дней к расписанию: 0 – Применять с заменой 1 – Применять со вставкой 2 – Не применять |

WorktimeSchedule

Базовый класс: [AccessSchedule](#).

Класс, используемый для представления расписания рабочего времени.

| | |
|----------------------------------|---|
| <code>Guid</code> ID | Уникальный ключ расписания. |
| <code>string</code> NAME | Наименование. |
| <code>string</code> DESC | Описание расписания. |
| <code>bool</code> IS_WEEK | Признак, является ли расписание недельным. |
| <code>int</code> HOLIDAYS_ACTION | Способ применения праздничных дней к расписанию: 0 – Применять с заменой; 1 – Применять со вставкой; 2 – Не применять. |
| <code>int</code> HOURS_PER_WEEK | Норма отработки в неделю (часы). |
| <code>int</code> HOURS_PER_DAY | Норма отработки в день (часы). |

ScheduleDay

Базовый класс: [BaseObject](#).

Класс, используемый для описания шаблона дня в цикле расписания.

| | |
|---------------------------------------|--|
| <code>DateTime</code> DATE | Дата начала цикла расписания. |
| <code>int</code> INDEX | Номер дня в цикле расписания. Индекс первого дня в цикле имеет значение «1». |
| <code>TimeInterval[]</code> INTERVALS | Массив временных интервалов в шаблоне дня. |

ScheduleFix

Базовый класс: [ScheduleDay](#).

Класс, используемый для описания дня-поправки.

| | |
|---------------------------------------|--|
| <code>DateTime</code> DATE | Дата поправки. |
| <code>int</code> INDEX | Не используется. |
| <code>TimeInterval[]</code> INTERVALS | Массив временных интервалов в дне-поправке. |
| <code>int</code> ACTION | Способ применения дня-поправки в расписании: 0 – Применить с заменой; 1 – Применить со вставкой. Для неделных расписаний значение всегда «0». |

TimeInterval

Базовый класс: [BaseObject](#).

Класс, используемый для представления временного интервала.

| | |
|-----------------------------|----------------------|
| <code>DateTime</code> START | Начало интервала. |
| <code>DateTime</code> END | Окончание интервала. |

WorkTimeInterval

Базовый класс: [TimeInterval](#).

Класс, используемый для представления временного интервала расписания рабочего времени.

| | |
|-----------------------------|--|
| <code>DateTime</code> START | Начало интервала. |
| <code>DateTime</code> END | Окончание интервала. |
| <code>int</code> TYPE | Тип временного интервала: 0 - Рабочее время (доступ разрешен); 1 - Ночная смена; 2 - Перерыв (обед); 3 - Обязательное рабочее время. |

Holiday

Базовый класс: [BaseObject](#).

Класс, используемый для описания праздничных дней.

| | |
|--------------------------|-------------------------|
| <code>string</code> NAME | Название праздника. |
| <code>byte</code> MONTH | Месяц (даты праздника). |
| <code>byte</code> DAY | Число (даты праздника). |

BaseIdentifier

Базовый класс: [BaseObject](#).

Класс, используемый для описания идентификатора.

| | |
|------------------------------|---|
| <code>string</code> CODE | Код идентификатора (строка, содержащая число в шестнадцатеричном формате, длиной ровно 8 символов). |
| <code>Guid</code> PERSON_ID | Уникальный ключ сотрудника. |
| <code>bool</code> IS_PRIMARY | Признак, является ли идентификатор первичным. |

Identifier

Базовый класс: [BaseIdentifier](#).

Класс, используемый для описания идентификатора.

| | |
|----------------------------------|---|
| <code>string</code> CODE | Код идентификатора (строка, содержащая число в шестнадцатиричном формате, длиной ровно 8 символов). |
| <code>Guid</code> PERSON_ID | Уникальный ключ сотрудника. |
| <code>bool</code> IS_PRIMARY | Признак, является ли идентификатор первичным. |
| <code>Guid</code> ACCGROUP_ID | Уникальный ключ группы доступа идентификатора. |
| <code>long</code> PRIVILEGE_MASK | Маска привилегий. |
| <code>int</code> IDENTIFYTYPE | Тип идентификатора. (Значения: 0 - Подсистема доступа «Parsec»; 1 – Автомобильный номер). |
| <code>string</code> NAME | Наименование идентификатора. Используется в информационных целях. |

IdentifierTemp

Базовый класс: [Identifier](#).

Класс, используемый для описания временного идентификатора.

| | |
|----------------------------------|---|
| <code>string</code> CODE | Код идентификатора (строка, содержащая число в шестнадцатиричном формате, длиной ровно 8 символов). |
| <code>Guid</code> PERSON_ID | Уникальный ключ сотрудника. |
| <code>bool</code> IS_PRIMARY | Признак, является ли идентификатор первичным. |
| <code>Guid</code> ACCGROUP_ID | Уникальный ключ группы доступа идентификатора. |
| <code>long</code> PRIVILEGE_MASK | Маска привилегий. |
| <code>int</code> IDENTIFYTYPE | Тип идентификатора. (Значения: 0 - Подсистема доступа «Parsec»; 1 – Автомобильный номер). |
| <code>string</code> NAME | Наименование идентификатора. Используется в информационных целях. |
| <code>DateTime</code> VALID_FROM | Дата начала действия идентификатора. |
| <code>DateTime</code> VALID_TO | Дата окончания действия идентификатора. |

StockIdentifier

Базовый класс: [IdentifierTemp](#).

Класс, используемый для описания идентификатора из пула.

| | |
|------------------------------|---|
| <code>string</code> CODE | Код идентификатора (строка, содержащая число в шестнадцатиричном формате, длиной ровно 8 символов). |
| <code>Guid</code> PERSON_ID | Уникальный ключ сотрудника. |
| <code>bool</code> IS_PRIMARY | Признак, является ли идентификатор первичным. |

| | |
|----------------------------------|---|
| <code>Guid</code> ACCGROUP_ID | Уникальный ключ группы доступа идентификатора. |
| <code>long</code> PRIVILEGE_MASK | Маска привилегий. |
| <code>int</code> IDENTIFYTYPE | Тип идентификатора. (Значения: 0 - Подсистема доступа «Parsec»; 1 – Автомобильный номер). |
| <code>string</code> NAME | Наименование идентификатора. Используется в информационных целях. |
| <code>DateTime</code> VALID_FROM | Дата начала действия идентификатора. |
| <code>DateTime</code> VALID_TO | Дата окончания действия идентификатора. |

IdentifierExData

Базовый класс: [BaseObject](#).

Класс, используемый для описания дополнительных свойств идентификатора.

| | |
|--------------------------------------|---|
| <code>Guid</code> PASSAGE_ROLE_ID | Уникальный ключ роли группового прохода. |
| <code>int</code> ENTRY_LIMIT | Максимальное разрешенное количество проходов (значение «-1» - не ограниченное количество проходов; «0» - доступ запрещен; «127» – максимально возможное ограничение). |
| <code>Guid</code> OWNED_COMPONENT_ID | Уникальный ключ компонента «Дверь», владельцем которого является идентификатор. (Используется в функционале «Владелец кабинета») |

PassageRole

Базовый класс: [BaseObject](#).

Класс, используемый для описания роли группового прохода.

| | |
|---------------------------------|-----------------------|
| <code>Guid</code> ID | Уникальный ключ роли. |
| <code>string</code> NAME | Название роли. |
| <code>string</code> DESCRIPTION | Описание роли |

BaseTerritory

Базовый класс, используемый для описания территории.

| | |
|--------------------------|-----------------------------|
| <code>Guid</code> ID | Уникальный ключ территории. |
| <code>byte</code> TYPE | Тип объекта территории. |
| <code>string</code> NAME | Название территории. |
| <code>string</code> DESC | Описание территории. |

Тип объекта территории может принимать значение:

- 0 – папка;
- 1 – дверь;
- 2 – деталь (пока не используется);

Territory

Базовый класс: [BaseTerritory](#).

Класс, используемый для описания территории.

| | |
|-----------------------------|--|
| <code>Guid</code> ID | Уникальный ключ территории. |
| <code>byte</code> TYPE | Тип объекта территории. |
| <code>string</code> NAME | Название территории. |
| <code>string</code> DESC | Описание территории. |
| <code>Guid</code> PARENT_ID | Уникальный ключ родительской территории. |

TerritoryWithComponent

Базовый класс: [Territory](#).

Класс, используемый для описания территории с информацией о связанном с ней компоненте.

| | |
|--------------------------------|--|
| <code>Guid</code> ID | Уникальный ключ территории. |
| <code>byte</code> TYPE | Тип объекта территории. |
| <code>string</code> NAME | Название территории. |
| <code>string</code> DESC | Описание территории. |
| <code>Guid</code> PARENT_ID | Уникальный ключ родительской территории. |
| <code>Guid</code> COMPONENT_ID | Уникальный ключ компонента. |
| <code>long</code> FEATURE_MASK | Битовая маска свойств компонента. Значение битов описано в таблице ниже. |

Маска свойств компонента:

| Номер бита | Значение |
|------------|---|
| 0 | Дверь |
| 1 | 2 считывателя |
| 2 | АПБ |
| 3 | поддерживает сложные расписания |
| 4 | используется как настольный считыватель |
| 5 | охранная область |
| 6 | источник видео потока |
| 7 | турникет |
| 8 | -- Зарезервирован -- |
| 9 | не поддерживает расписания |
| 10 | программный контроллер |
| 11 | -- Зарезервирован -- |
| 12 | не ограничивается лицензией |

| | |
|----|---|
| 13 | лифтовой контроллер |
| 14 | -- Зарезервирован -- |
| 62 | отключен оператором |
| 63 | не обслуживается по ограничениям лицензии |

AccessGroup

Базовый класс: [BaseObject](#).

Класс, используемый для описания группы доступа.

| | |
|-------------------------------|---|
| <code>Guid</code> ID | Уникальный ключ группы доступа. |
| <code>string</code> NAME | Название группы доступа. |
| <code>int</code> IDENTIFYTYPE | Тип группы доступа. (Значения: 0 - Подсистема доступа «Parsec»; 1 – Автомобильный номер). |

SubAccessGroup

Базовый класс: [BaseObject](#).

Класс, используемый для описания подгруппы доступа.

| | |
|---------------------------------|-----------------------------------|
| <code>Guid</code> SubGroupID | Уникальный ключ группы компонент. |
| <code>Guid</code> ScheduleID | Уникальный ключ расписания. |
| <code>Guid[]</code> Territories | Массив ключей территорий. |

Event

Базовый класс: [BaseObject](#).

Класс, используемый для описания события системы.

| | |
|---|------------------------------------|
| <code>DateTime</code> EventDate | Дата события. |
| <code>int</code> EventType | Тип события (0-Вход; 1-Выход). |
| <code>int</code> EventPersonIndex | Индекс сотрудника в EventsHistory. |
| <code>string</code> CODE | Код идентификатора. |
| <code>int</code> EventTerritoryIndex | Индекс территории в EventsHistory. |

Тип события может принимать значение:

0 – Вход;

1 – Выход.

EventsHistory

Класс, используемый для описания событий системы.

| | |
|--|---------------------------------------|
| <code>Event[]</code> Events | Массив событий. |
| <code>Guid[]</code> Persons | Массив уникальных ключей сотрудников. |
| <code>string[]</code> PersonFullNames | Массив Ф.И.О. сотрудников. |

| | |
|--------------------------------------|--------------------------------------|
| <code>Guid[] Territories</code> | Массив уникальных ключей территорий. |
| <code>string[] TerritoryNames</code> | Массив названий территорий. |

EventObject

Базовый класс: [BaseObject](#).

Класс, используемый для описания данных события.

| | |
|------------------------------|---|
| <code>Object[] Values</code> | Массив значений полей, описывающих событие. |
|------------------------------|---|

Domain

Базовый класс: [BaseObject](#).

Класс, используемый для описания организации.

| | |
|-----------------------------------|--|
| <code>string NAME</code> | Наименование организации. |
| <code>string DESCRIPTION</code> | Описание организации. |
| <code>bool VISITOR_CONTROL</code> | Признак Бюро Пропусков (не используется начиная с версии 3.2). |
| <code>bool IS_SYSTEM</code> | Признак организации SYSTEM. |

EventHistoryQueryParams

Класс, используемый для описания критериев формируемого отчета по событиям. В качестве любого параметра может быть передан null. В таком случае данный критерий не будет использован при отборе событий в отчет.

Параметры даты и времени передаются в UTC при значении параметра `UseLocalTime == false` (по умолчанию). Подробнее параметр описан в таблице ниже.

| | |
|-----------------------------------|--|
| <code>Guid[] IDs</code> | Массив ключей событий, которые будут отображены в отчет. Если задан, все остальные параметры игнорируются. |
| <code>Guid[] ParentEventID</code> | Массив ключей «первичных» событий. В отчет будут отображены все события, связанные с первичными событиями. Если задан, все параметры (кроме <code>IDs</code>) игнорируются. Значение <code>ParentEventId={Guid.Empty}</code> интерпретируется так же, как значение <code>{null}</code> (для совместимости с Delphi). |
| <code>DateTime StartDate</code> | Дата начала временного периода, за который создается отчет. |
| <code>DateTime EndDate</code> | Дата окончания временного периода, за который создается отчет. |
| <code>DateTime StartTime0</code> | Время начала часового диапазона внутри временного периода. В отчет будут отображены только события, произошедшие в течение этого диапазона. |

| | |
|---|--|
| <code>DateTime</code> <code>EndTime0</code> | Время окончания часового диапазона внутри временного периода. В отчет будут отображены только события, произошедшие в течение этого диапазона. |
| <code>DateTime</code> <code>StartTime1</code> | Время начала второго часового диапазона внутри временного периода. Задействуется, если в диапазон попадает окончание суток.* |
| <code>DateTime</code> <code>EndTime1</code> | Время окончания второго часового диапазона внутри временного периода. Задействуется, если в диапазон попадает окончание суток.* |
| <code>Guid[]</code> <code>Territories</code> | Массив ключей территорий, по которым будет сформирован отчет. |
| <code>Guid[]</code> <code>Operators</code> | Параметр предназначен для внутреннего использования. |
| <code>UInt[]</code> <code>TransactionTypes</code> | Массив ключей типов транзакций. В качестве параметра можно использовать один или несколько ключей транзакций из приведенной ниже таблицы. |
| <code>Guid[]</code> <code>Organizations</code> | Массив ключей подразделений, по которым будет сформирован отчет. |
| <code>Guid[]</code> <code>Users</code> | Массив ключей пользователей, по событиям которых будет сформирован отчет. |
| <code>bool</code> <code>EventsWithoutUser</code> | Параметр предназначен для внутреннего использования. |
| <code>int</code> <code>MaxResultSize</code> | Максимальное количество событий, отображенных в отчете. При передаче null отображается количество событий по умолчанию (5000 шт.). |
| <code>bool</code> <code>UseLocalTime</code> | При установке данного параметра в true события будут фильтроваться и возвращаться с учетом часового пояса контроллеров. |

* Если в часовой диапазон попадает момент окончания суток (24.00), то такой диапазон нужно разбить на два: с XX часов по 24.00 и с 00.00 часов до YY. При этом для первой части диапазона используются параметры `StartTime0` и `EndTime0`, а для второй части - `StartTime1` и `EndTime1`.

Ключи транзакций:

| Ключ | Описание транзакции |
|--------|------------------------------|
| 590144 | Нормальный вход по ключу |
| 590152 | Фактический вход |
| 590145 | Нормальный выход по ключу |
| 590153 | Фактический выход |
| 590146 | Выход вне временного профиля |
| 590244 | Нормальный выход посетителя |
| 590245 | Фактический выход посетителя |

EventFilter

Класс, используемый для отбора событий по подписке по типу транзакций.

| | |
|--------------------------------------|---|
| <code>UInt[] TransactionTypes</code> | Массив ключей типов транзакций. В качестве параметра можно использовать один или несколько ключей транзакций. |
|--------------------------------------|---|

AdvancedEventFilter

Базовый класс: [EventFilterBaseObject](#).

Расширенный класс, используемый для отбора событий по подписке.

| | |
|--------------------------------------|---|
| <code>UInt[] TransactionTypes</code> | Массив ключей типов транзакций. В качестве параметра можно использовать один или несколько ключей транзакций. |
| <code>Guid[] Territories</code> | Массив ключей территорий, по которым будут отбираться события. |
| <code>Guid[] Organizations</code> | Массив ключей подразделений, по которым будут отбираться события. |
| <code>Guid[] Users</code> | Массив ключей пользователей, по которым будут отбираться события. |

HardwareState

Базовый класс: [BaseObject](#).

Класс, используемый для описания состояния территории.

| | |
|-------------------------------|--|
| <code>Guid TerritoryID</code> | Уникальный ключ территории. |
| <code>ulong State</code> | Набор состояний территории – битовая маска размером 4 байта. |

TransactionClass

Базовый класс: [BaseObject](#).

Класс, используемый для описания категорий событий системы.

Каждое событие в системе ParsecNET принадлежит одной или более категориям.

| | |
|---------------------------|--|
| <code>long ID</code> | ID категории события системы. Всегда равен 1 поразрядно сдвинутой влево на N позиций, где N принимает значения от 0 до 63. Таким образом каждая категория – это 1 бит в битовой маске события. |
| <code>bool NAME</code> | Название категории событий системы. |
| <code>bool IS_USER</code> | Признак пользовательской категории: 0 – системная категория; 1 – категория, созданная пользователем. |

TransactionType

Базовый класс: [BaseObject](#).

Класс, используемый для описания типов событий системы.

| | |
|------------------------------|--|
| <code>int</code> ID | Уникальный идентификатор события. |
| <code>long</code> CLASS_MASK | Битовая маска, задающая набор категорий, к которым относится данное событие. |
| <code>String</code> NAME | Название события. |

QRAdvancedGroup

Базовый класс: [BaseObject](#).

Класс, используемый для описания группы контроллеров расширенных QR-кодов.

| | |
|--------------------------|------------------------------------|
| <code>short</code> ID | Идентификатор группы контроллеров. |
| <code>String</code> NAME | Название группы. |

QRAdvancedData

Базовый класс: [BaseObject](#).

Класс, используемый для задания прав доступа расширенного QR-кода.

| | |
|--------------------------------|---|
| <code>DateTime</code> DateFrom | Дата начала срока действия QR-кода. |
| <code>DateTime</code> DateTo | Дата окончания срока действия QR-кода. |
| <code>DateTime</code> TimeFrom | Начало временного интервала внутри срока действия QR-кода. Необязательный параметр. |
| <code>DateTime</code> TimeTo | Конец временного интервала внутри срока действия QR-кода. Необязательный параметр. |
| <code>short[]</code> Groups | Массив ключей групп контроллеров, через которые QR-код имеет право прохода. |
| <code>uint</code> CardID | Идентификатор субъекта доступа. |

ВЕРСИЯ

Функция **GetVersion**

```
string GetVersion()
```

Параметры: без параметров.

Результат: Возвращает версию сервиса интеграции.

Описание: Функция возвращает версию сервиса интеграции.

СЕССИЯ

Функция **GetDomains**

```
Domain[] GetDomains()
```

Параметры: Нет

Результат: Возвращает массив [Domain](#).

Описание: Возвращает массив организаций.

Функция **OpenSession**

```
SessionResult OpenSession( string domain, string userName, string password )
```

Параметры:

| | |
|------------------------------|---|
| <code>string domain</code> | Название организации для входа. Для входа в системную организацию можно использовать пустую строку. |
| <code>string userName</code> | Имя оператора. |
| <code>string password</code> | Пароль оператора. |

Результат: Возвращает класс.

Описание: Данная функция используется для аутентификации оператора в сервисе интеграции. Полученный ключ сессии используется в дальнейшем для выполнения всех операций. Каждая сессия открывается на 5 минут, при выполнении любой операции на сервере интеграции время сессии продлевается.

Функция **OpenSessionWithInLocale**

```
SessionResult OpenSessionWithInLocale( string domain, string userName, string password, string locale )
```

Параметры:

| | |
|------------------------------|---|
| <code>string domain</code> | Название организации для входа. Для входа в системную организацию можно использовать пустую строку. |
| <code>string userName</code> | Имя оператора. |
| <code>string password</code> | Пароль оператора. |
| <code>string locale</code> | Требуемый язык. Возможны значения «ru-RU», «en-US» или «es-ES». |

Результат: Возвращает класс.

Описание: Данная функция используется для аутентификации оператора в сервисе интеграции с указанием языка интерфейса. Полученный ключ сессии используется в дальнейшем для выполнения всех операций. Каждая сессия открывается на 5 минут, при выполнении любой операции на сервере интеграции время сессии продлевается.

Функция ContinueSession

```
int ContinueSession( Guid sessionId )
```

Параметры:

`Guid sessionId` Уникальный ключ сессии.

Результат: В качестве результата возвращается значение: 0 – если операция прошла успешно; -1 – если операция выполнена с ошибками.

Описание: Используется для продления сессии.

Функция CloseSession

```
void CloseSession( Guid sessionId )
```

Параметры:

`Guid sessionId` Уникальный ключ сессии.

Результат: –

Описание: Используется для закрытия сессии.

Функция CheckRole

```
BaseResult CheckRole( Guid sessionId, string roleName )
```

Параметры:

`Guid sessionId` Уникальный ключ сессии.

`string roleName` Наименование права на совершение операций.

Результаты: Возвращает [BaseResult](#).

Описание: Проверяет доступность действий для текущего оператора. В интеграционном сервисе используются следующие права:

| | |
|-----------------------------|---|
| "EmployeeReader" | Право получения сведений о персонале и структуре подразделений. |
| "EmployeeWriter" | Право на удаление/изменение сведений о персонале и структуре подразделений. |
| "PersonDelete" | Право на удаление субъектов доступа. |
| "AccessGroupReader" | Право получения сведений о группах доступа. |
| "AccessGroupWriter" | Право на изменение групп доступа. |
| "GuestReader" | Право получения сведений о посетителях. |
| "VisitorRequestCreator" | Право на создание, редактирование и удаление заявок и посетителей. |
| "VisitorRequestCoordinator" | Право на согласование заявок для посетителей. |

| | |
|--------------------------|--|
| "VisitorPassDistributor" | Право на выдачу пропусков для посетителей и закрытие заявок. |
| "TimesheetReader" | Право просмотра сведений о расписании. |
| "TimesheetWriter" | Право на создание, удаление и изменение расписаний. |
| "HardwareControl" | Право на отправку команд управления оборудованием доступа. |
| "AlarmHardwareControl" | Право на отправку команд постановки и снятия с охраны. |
| "HardwareWriter" | Право «Полный доступ» к Редактору оборудования (в части управления ролями группового прохода). |
| "MonitorReader" | Право для получения событий системы и статусов оборудования в реальном времени. |
| "VideoVerification" | Право на подтверждение доступа. |

Функция GetObjectName

`StringResult GetObjectName (Guid sessionID, Guid objectID)`

Параметры:

| | |
|-----------------------------|--------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid objectID</code> | Уникальный ключ объекта. |

Результат: Возвращает [StringResult](#).

Описание: Возвращает наименование любого объекта системы по его ключу.

ОБОРУДОВАНИЕ

Функция SendHardwareCommand

`BaseResult SendHardwareCommand(Guid sessionID, Guid territoryID, int command)`

Параметры:

| | |
|-------------------------------|-----------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid territoryID</code> | Уникальный ключ территории. |
| <code>int command</code> | Код посылаемой команды. |

Результат: Возвращает [BaseResult](#).

Описание: Посылает команду устройству, представленному территорией. Коды команд и устройства описаны в таблице ниже.

| Команды | | Контрол. доступа Parsec | Охранная область AC-08 | Охранный раздел Болид |
|---------|--|-------------------------|------------------------|-----------------------|
| Код | Описание | | | |
| 1 | Включить реле на вход* \ Открыть дверь | ✓ | | |
| 2 | Включить реле на выход* | ✓ | | |

| | | | | |
|--|-------------------------------------|---|---|---|
| 4 | Заккрыть дверь | ✓ | | |
| 8 | Установить относительную блокировку | ✓ | | |
| 16 | Снять относительную блокировку | ✓ | | |
| 32 | Установить абсолютную блокировку | ✓ | | |
| 64 | Снять абсолютную блокировку | ✓ | | |
| 128 | Установить на охрану | ✓ | ✓ | ✓ |
| 256 | Снять с охраны | ✓ | ✓ | ✓ |
| 512 | Включить доп. реле | ✓ | | |
| 1024 | Выключить доп. реле | ✓ | | |
| 2048 | Сброс антипассбека | ✓ | | |
| * Команды работают только для контроллеров в режиме турникета. | | | | |

Функция **SendVerificationCommand**

```
BaseResult SendVerificationCommand( Guid sessionID, Guid territoryID, Guid
personID, bool passAllow )
```

Параметры:

| | |
|-------------------------------|-------------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid territoryID</code> | Уникальный ключ территории. |
| <code>Guid personID</code> | Уникальный ключ субъекта доступа. |
| <code>bool passAllow</code> | Флаг разрешения/запрещения прохода. |

Результат: Возвращает массив `BaseResult`.

Описание: Посылает команду устройству, представленному территорией, на подтверждение или запрет прохода. В настройках контроллера должно быть выбрано программное подтверждение прохода.

Функция **GetHardwareState**

```
HardwareState[] GetHardwareState(Guid sessionID, Guid[] territoryIDs)
```

Параметры:

| | |
|----------------------------------|---------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid[] territoryIDs</code> | Массив ключей территорий. |

Результат: Возвращает массив битовых масок `HardwareState` (каждая маска размером 8 байт).

Описание: Выдает набор состояний для выбранных территорий. Коды состояний описаны в таблицах ниже. Функция применима только к территориям, порождаемым контроллерами доступа серии NC, охранном контроллером AC-08 и охранной системой Болид.

Биты состояния территории для контроллеров серии NC

| № бита | Значение | Состояние при значении бита | |
|---|-------------------------------------|-----------------------------|--------------------|
| | | 0 | 1 |
| 0 | Аккумулятор | Разряжен | Норма |
| 1 | Сетевое питание | Отключено | Норма |
| 2 | Батарея | Неисправна | Норма |
| 3 | Корпус | Открыт | Закрит |
| 4 | Реле на вход* \ Замок | Выключено* \ Закрит | Включено* \ Открыт |
| 5 | Реле на выход* | Выключено* | Включено* |
| 6 | Доп. реле | Выключено | Включено |
| 7 | Реле картоприемника | Выключено | Включено |
| 8 | Абсолютная блокировка | Выключена | Включена |
| 9 | Относительная блокировка | Выключена | Включена |
| 10 | Экстренное открывание двери | Выключено | Включено |
| 11 | Охрана | Снята | На охране |
| 12 | Охранный датчик | Норма | Активирован |
| 13 | Датчик входа** \ Дверной контакт | Норма | Активирован |
| 14 | Датчик выхода** | Норма | Активирован |
| 28 | Выключен | Нет | Да |
| 31 | Недоступен | Нет | Да |
| * Значения указаны для контроллеров в режиме турникета. | | | |
| **Значения указаны только для контроллера NC-100K. | | | |

Биты состояния территории-охранной области контроллера AC-08 и охранной системы Bolid

| № бита | Значение | Состояние при значении бита | |
|--------|--------------------|-----------------------------|-------------|
| | | 0 | 1 |
| 0 | Аккумулятор* | Разряжен | Норма |
| 1 | Сетевое питание* | Отключено | Норма |
| 3 | Корпус* | Открыт | Закрит |
| 11 | Охрана | Снята | На охране |
| 12 | Охранный датчик | Норма | Активирован |
| 24 | Требуется внимания | Нет | Да |

| | | | |
|---|------------|-----|----|
| 28 | Выключен | Нет | Да |
| 31 | Недоступен | Нет | Да |
| * Значения указаны для контроллера AC-08. | | | |

ПОДРАЗДЕЛЕНИЯ

Функция GetRootOrgUnit

`OrgUnit GetRootOrgUnit(Guid sessionID)`

Параметры:

`Guid sessionID` Уникальный ключ сессии.

Результат: Возвращает корневую OrgUnit.

Описание: данная функция предназначена для получения корневого объекта дерева подразделений.

Функция GetOrgUnitsHierarchy

`OrgUnit[] GetOrgUnitsHierarchy(Guid sessionID)`

Параметры:

`Guid sessionID` Уникальный ключ сессии.

Результат: Массив OrgUnit.

Описание: Возвращает полную иерархию подразделений.

Функция GetOrgUnitsHierarchyWithPersons

`BaseObject[] GetOrgUnitsHierarchyWithPersons(Guid sessionID)`

Параметры:

`Guid sessionID` Уникальный ключ сессии.

Результат: Массив BaseObject, элементы массива могут быть OrgUnit или Person.

Описание: Возвращает полную иерархию подразделений вместе с сотрудниками.

Функция GetOrgUnitsHierarchyWithVisitors

`BaseObject[] GetOrgUnitsHierarchyWithVisitors(Guid sessionID)`

Параметры:

`Guid sessionID` Уникальный ключ сессии.

Результат: Массив BaseObject, элементы массива могут быть OrgUnit или Person.

Описание: Возвращает полную иерархию подразделений вместе с посетителями.

Функция GetOrgUnitsHierarchyWithVehicle

`BaseObject[] GetOrgUnitsHierarchyWithVehicle(Guid sessionID)`

Параметры:

`Guid sessionID` Уникальный ключ сессии.

Результат: Массив `BaseObject`, элементы массива могут быть `OrgUnit` или `Person`.

Описание: Возвращает полную иерархию подразделений вместе с автомобилями.

Функция `GetOrgUnitSubItems`

```
BaseObject[] GetOrgUnitSubItems( Guid sessionID, Guid orgUnitID )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.

`Guid orgUnitID` Уникальный ключ подразделения.

Результат: Массив `BaseObject`, элементы массива могут быть `BaseOrgUnit` или `BasePerson`.

Описание: Возвращает массив подразделений и сотрудников, принадлежащих подразделению с указанным ключом.

Функция `GetOrgUnitSubItemsHierarchyWithPersons`

```
BaseObject[] GetOrgUnitSubItemsHierarchyWithPersons( Guid sessionID, Guid orgUnitID )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.

`Guid orgUnitID` Уникальный ключ подразделения.

Результат: Массив `BaseObject`, элементы массива могут быть `OrgUnit` или `Person`.

Описание: Возвращает полную иерархию подразделений и их сотрудников, начиная с подразделения с указанным ключом.

Функция `GetOrgUnitSubItemsHierarchyWithVisitors`

```
BaseObject[] GetOrgUnitSubItemsHierarchyWithVisitors( Guid sessionID, Guid orgUnitID )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.

`Guid orgUnitID` Уникальный ключ подразделения.

Результат: Массив `BaseObject`, элементы массива могут быть `OrgUnit` или `Person`.

Описание: Возвращает полную иерархию подразделений, начиная с подразделения с указанным ключом, и посетителей этих подразделений.

Функция `GetOrgUnitSubItemsHierarchyWithVehicle`

```
BaseObject[] GetOrgUnitSubItemsHierarchyWithVehicle( Guid sessionID, Guid orgUnitID )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.

`Guid orgUnitID` Уникальный ключ подразделения.

Результат: Массив BaseObject, элементы массива могут быть OrgUnit или Person.

Описание: Возвращает полную иерархию подразделений, начиная с подразделения с указанным ключом, и транспортных средств этих подразделений.

Функция GetOrgUnit

```
OrgUnit GetOrgUnit( Guid sessionId, Guid orgUnitID )
```

Параметры:

`Guid sessionId` Уникальный ключ сессии.

`Guid orgUnitID` Уникальный ключ подразделения.

Результат: Возвращает OrgUnit для подразделения с указанным ключом или null, если указанное подразделение не найдено.

Описание: Возвращает информацию о подразделении с указанным ключом.

РЕДАКТИРОВАНИЕ ПОДРАЗДЕЛЕНИЯ

Функция CreateOrgUnit

```
GuidResult CreateOrgUnit( Guid sessionId, OrgUnit orgUnit )
```

Параметры:

`Guid sessionId` Уникальный ключ сессии.

`OrgUnit orgUnit` Параметры подразделения.

Результат: Возвращает GuidResult.

Описание: Создает подразделение с указанными данными. Возвращает ключ вновь созданного подразделения. Если поле ID в структуре OrgUnit равен Guid.Empty (00000000-0000-0000-0000-000000000000), то ID генерируется автоматически и возвращается. Иначе используется заданное значение ID.

Функция OpenOrgUnitEditingSession

```
GuidResult OpenOrgUnitEditingSession( Guid sessionId, Guid orgUnitID )
```

Параметры:

`Guid sessionId` Уникальный ключ сессии.

`Guid orgUnitID` Уникальный ключ подразделения.

Результат: Возвращает GuidResult.

Описание: Открывает сессию редактирования подразделения. Возвращает ключ вновь созданной сессии редактирования подразделения.

Функция CloseOrgUnitEditingSession

```
void CloseOrgUnitEditingSession( Guid orgUnitEditSessionID )
```

Параметры:

`Guid orgUnitEditSessionID` Уникальный ключ сессии редактирования подразделения.

Результат: –

Описание: Закрывает сессию редактирования подразделения.

Функция SaveOrgUnit

```
BaseResult SaveOrgUnit( Guid orgUnitEditSessionID, BaseOrgUnit orgUnit )
```

Параметры:

| | |
|--|--|
| <code>Guid orgUnitEditSessionID</code> | Уникальный ключ сессии редактирования подразделения. |
| <code>BaseOrgUnit orgUnit</code> | Параметры подразделения. |

Результат: Возвращает BaseResult.

Описание: Изменяет параметры подразделения. Параметр orgUnit может быть BaseOrgUnit или OrgUnit.

При передаче в запросе в качестве параметров подразделения структуры OrgUnit необходимо в обязательном порядке указывать в элементе *orgUnit* атрибут *xsi:type="OrgUnit"*.

После удачного сохранения подразделения сессия редактирования закрывается.

Функция DeleteOrgUnit

```
BaseResult DeleteOrgUnit( Guid sessionID, Guid orgUnitID )
```

Параметры:

| | |
|-----------------------------|--------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid orgUnitID</code> | Уникальный ключ подразделения. |

Результат: Возвращает BaseResult.

Описание: Удаляет подразделение с указанным ключом.

ПЕРСОНАЛ

Функция GetPersonExtraFieldTemplates

```
PersonExtraFieldTemplate[] GetPersonExtraFieldTemplates( Guid sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив PersonExtraFieldTemplate.

Описание: Возвращает набор шаблонов дополнительных полей у сотрудников.

Функция GetVisitorExtraFieldTemplates

```
PersonExtraFieldTemplate [] GetVisitorExtraFieldTemplates( Guid sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив PersonExtraFieldTemplate.

Описание: Возвращает набор шаблонов дополнительных полей у посетителей.

Функция GetVehicleExtraFieldTemplates

```
PersonExtraFieldTemplate [] GetVehicleExtraFieldTemplates( Guid  
sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив PersonExtraFieldTemplate.

Описание: Возвращает набор шаблонов дополнительных полей у транспортных средств.

Функция FindPeople

```
BasePerson[] FindPeople( Guid sessionID, string lastname, string  
firstname, string middlename )
```

Параметры:

| | |
|--------------------------------|-------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>string lastname</code> | Значение фамилии для поиска. |
| <code>string firstname</code> | Значение имени для поиска. |
| <code>string middlename</code> | Значение отчества для поиска. |

Результат: Возвращает массив Person.

Описание: Выдает набор сотрудников, подходящих под переданные критерии. Поиск может производиться как по любому параметру отдельно, так и по всем параметрам. Должен быть указан хотя бы один параметр.

Функция FindVisitors

```
BasePerson[] FindVisitors( Guid sessionID, string lastname, string  
firstname, string middlename )
```

Параметры:

| | |
|--------------------------------|-------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>string lastname</code> | Значение фамилии для поиска. |
| <code>string firstname</code> | Значение имени для поиска. |
| <code>string middlename</code> | Значение отчества для поиска. |

Результат: Возвращает массив Person.

Описание: Выдает набор посетителей, подходящих под переданные критерии. Поиск может производиться как по любому параметру отдельно, так и по всем параметрам. Должен быть указан хотя бы один параметр.

Функция FindVehicle

```
BasePerson[] FindVehicle( Guid sessionID, string number, string model,  
string color )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

| | |
|----------------------------|--|
| <code>string</code> number | Значение номера автомобиля для поиска. |
| <code>string</code> model | Значение модели автомобиля для поиска. |
| <code>string</code> color | Значение цвета для поиска. |

Результат: Возвращает массив Person.

Описание: Выдает набор автомобилей, подходящих под переданные критерии. Поиск может производиться как по любому параметру отдельно, так и по всем параметрам. Должен быть указан хотя бы один параметр.

Функция FindPersonByIdentifier

```
Person FindPersonByIdentifier( Guid sessionId, string cardCode )
```

Параметры:

| | |
|------------------------------|--|
| <code>Guid</code> sessionId | Уникальный ключ сессии. |
| <code>string</code> cardCode | Код идентификатора карты в 16-ричном формате в верхнем регистре (Например: A12345BCF). |

Результат: Возвращает Person.

Описание: Возвращает информацию о субъекте доступа с указанным кодом идентификатора.

Функция PersonSearch

```
Person[] PersonSearch( Guid sessionId, Guid fieldID, int relation, object value, object value1 )
```

Параметры:

| | |
|-----------------------------|---|
| <code>Guid</code> sessionId | Уникальный ключ сессии. |
| <code>Guid</code> fieldID | Уникальный ключ поля, используемого для поиска. |
| <code>int</code> relation | Критерий поиска. |
| <code>object</code> value | Искомое значение или первое значение для критерия поиска «между». |
| <code>object</code> value1 | Второе значение для критерия поиска «между». |

Результат: Возвращает массив Person.

Описание: Возвращает информацию о субъектах доступа, удовлетворяющим параметрам поиска.

Ключи полей, используемые для поиска

| Guid | Описание | Тип значения |
|--------------------------------------|----------------------|------------------------|
| 0de358e0-c91b-4333-b902-000000000003 | Фамилия / Номер авто | string |
| 0de358e0-c91b-4333-b902-000000000001 | Имя / Модель авто | string |
| 0de358e0-c91b-4333-b902-000000000002 | Отчество / Цвет авто | string |

| Guid | Описание | Тип значения |
|--------------------------------------|-----------------------------|--------------------------|
| 0de358e0-c91b-4333-b902-000000000006 | Табель | string |
| 0de358e0-c91b-4333-b902-000000000004 | Подразделение | string |
| 0a679144-d5ce-476d-a56e-0a696f079b71 | Описание подразделения | string |
| 0de358e0-c91b-4333-b902-00000000000a | Группа доступа | string |
| 0de358e0-c91b-4333-b902-000000000005 | Код карты | string |
| 644E1B95-E87B-415D-91BF-C3242B6C3AE8 | Наименование идентификатора | string |
| 0de358e0-c91b-4333-b902-000000000007 | Время действия с | DateTime |
| 0de358e0-c91b-4333-b902-000000000008 | Время действия по | DateTime |
| 6FCFA1BB-9624-4248-A2D5-AA84901C53C8 | Субъект в чёрном списке | bool |
| 07AF86B3-23FC-44EF-8438-6EE601B2FCB0 | Доступ запрещен | bool |

Также можно использовать уникальный ключ шаблона дополнительного поля с соответствующим типом значения.

Критерии поиска

| Значение | Описание |
|----------|-----------------------|
| 0 | Равно (=) |
| 1 | Меньше или равно (<=) |
| 2 | Меньше (<) |
| 3 | Больше или равно (>=) |
| 4 | Больше (>) |
| 5 | Между |
| 6 | Содержит |
| 7 | Пусто |
| 8 | Не пусто |

Функция GetPerson

```
PersonWithPhoto GetPerson( Guid sessionID, Guid personID )
```

Параметры:

| | |
|-----------------------------|-----------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid personID</code> | Уникальный ключ субъекта доступа. |

Результат: Возвращает PersonWithPhoto.

Описание: Возвращает информацию о субъекте доступа с указанным ключом.

Функция GetMultiplePersons

`PersonWithPhoto[] GetMultiplePersons(Guid sessionID, Guid[] personIDs)`

Параметры:

| | |
|-------------------------------|----------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid[] personIDs</code> | Массив ключей субъектов доступа. |

Результат: Возвращает массив `PersonWithPhoto`.

Описание: Возвращает информацию о субъектах доступа с указанными ключами.

Функция GetPersonsChangedAfter

`Person[] GetPersonsChangedAfter(Guid sessionID, Guid orgID, DateTime dateFrom, bool includeSubOrg)`

Параметры:

| | |
|---------------------------------|---|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid orgID</code> | Уникальный ключ подразделения. |
| <code>DateTime dateFrom</code> | Начальная дата для анализа. |
| <code>bool includeSubOrg</code> | Признак поиска по вложенным подразделениям. |

Результат: Возвращает массив `Person`.

Описание: Возвращает набор субъектов доступа (сотрудник, посетитель, автомобиль), чьи данные были изменены, начиная с указанной даты. Поиск производится либо в указанном подразделении, либо в указанном и во всех вложенных.

Функция GetPersonExtraFieldValue

`ObjectResult GetPersonExtraFieldValue(Guid sessionID, Guid personID, Guid templateID)`

Параметры:

| | |
|------------------------------|-----------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid personID</code> | Уникальный ключ субъекта доступа. |
| <code>Guid templateID</code> | Уникальный ключ шаблона. |

Результат: Возвращает `ObjectResult`.

Описание: Возвращает значение указанного дополнительного поля субъекта доступа. Тип значения определяется типом поля.

Функция GetPersonExtraFieldValues

`ExtraFieldValue[] GetPersonExtraFieldValues(Guid sessionID, Guid personID)`

Параметры:

| | |
|-----------------------------|-----------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid personID</code> | Уникальный ключ субъекта доступа. |

Результат: Возвращает массив ExtraFieldValue.

Описание: Возвращает массив значений дополнительных полей субъекта доступа.

Функция GetPersonExtraFieldValueString

```
StringResult GetPersonExtraFieldValueString( Guid sessionID, Guid  
personID, Guid templateID )
```

Параметры:

| | |
|-----------------|-----------------------------------|
| Guid sessionID | Уникальный ключ сессии. |
| Guid personID | Уникальный ключ субъекта доступа. |
| Guid templateID | Уникальный ключ шаблона. |

Результат: Возвращает StringResult.

Описание: Возвращает значение дополнительного поля субъекта доступа в виде форматированной строки.

Функция ValidateExtraFieldValue

```
BaseResult ValidateExtraFieldValue( Guid sessionID, ExtraFieldValue  
value )
```

Параметры:

| | |
|-----------------------|--------------------------------|
| Guid sessionID | Уникальный ключ сессии. |
| ExtraFieldValue value | Значение дополнительного поля. |

Результат: Возвращает BaseResult.

Описание: Проверяет валидность значения дополнительного поля.

В запросе, в элементе *value* необходимо указывать тип значения в атрибуте *xsi:type*. Например, *xsi:type="xsd:string"* (см. пример из описания ф-ии [SetPersonExtraFieldValue](#)).

Функция GetPersonScheduleFixes

```
PersonScheduleFix[] GetPersonScheduleFixes( Guid sessionID, Guid  
personID )
```

Параметры:

| | |
|----------------|-----------------------------|
| Guid sessionID | Уникальный ключ сессии. |
| Guid personID | Уникальный ключ сотрудника. |

Результат: Возвращает массив PersonScheduleFix.

Описание: Возвращает список существующих поправок к рабочему времени сотрудника.

Функция AddPersonScheduleFix

```
GuidResult AddPersonScheduleFix( Guid sessionID, PersonScheduleFix fix  
)
```

Параметры:

| | |
|----------------|-------------------------|
| Guid sessionID | Уникальный ключ сессии. |
|----------------|-------------------------|

`PersonScheduleFix` Параметры поправки рабочего времени.
`fix`

Результат: Возвращает `GuidResult`.

Описание: Добавляет поправку рабочего времени сотрудника.

Функция **SavePersonScheduleFix**

```
BaseResult SavePersonScheduleFix( Guid sessionID, PersonScheduleFix fix )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.
`PersonScheduleFix` Параметры поправки рабочего времени.
`fix`

Результат: Возвращает `BaseResult`.

Описание: Изменяет параметры существующей поправки рабочего времени сотрудника.

Функция **DeletePersonScheduleFix**

```
BaseResult DeletePersonScheduleFix( Guid sessionID, Guid personID, Guid fixID )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.
`Guid personID` Уникальный ключ сотрудника.
`Guid fixID` Уникальный ключ поправки.

Результат: Возвращает `BaseResult`.

Описание: Удаляет поправку к рабочему времени сотрудника.

Функция **GetPersonWorktimeSchedule**

```
GuidResult GetPersonWorktimeSchedule( Guid sessionID, Guid personID )
```

Параметры:

`Guid sessionID` Уникальный ключ сессии.
`Guid personID` Уникальный ключ сотрудника.

Результат: Возвращает `GuidResult`.

Описание: Возвращает уникальный ключ расписания рабочего времени, назначенного сотруднику. Если расписание не назначено, возвращает `Guid.Empty`.

Функция **SetPersonWorktimeSchedule**

```
BaseResult setPersonWorktimeSchedule( Guid personEditSessionID, Guid scheduleID )
```

Параметры:

`Guid` Уникальный ключ сессии редактирования
`personEditSessionID` сотрудника.

`Guid` scheduleID

Уникальный ключ расписания рабочего времени.

Результат: Возвращает `BaseResult`.

Описание: Задает сотруднику персональное расписание рабочего времени, с указанным ключом.

Функция `GetBlackList`

```
BasePerson[] GetBlackList( Guid sessionID )
```

Параметры:

`Guid` sessionID

Уникальный ключ сессии.

Результат: Возвращает массив `Person`.

Описание: Выдает набор субъектов доступа, находящихся в черном списке.

Функция `FindInBlackList`

```
BasePerson[] FindInBlackList( Guid sessionID, string lastname, string  
firstname, string middlename )
```

Параметры:

`Guid` sessionID

Уникальный ключ сессии.

`string` lastname

Значение фамилии для поиска.

`string` firstname

Значение имени для поиска.

`string` middlename

Значение отчества для поиска.

Результат: Возвращает массив `Person`.

Описание: Выдает набор субъектов доступа, подходящих под переданные критерии и находящихся в черном списке. Поиск может производиться как по любому параметру отдельно, так и по всем параметрам. Должен быть указан хотя бы один параметр.

Функция `AddToBlackList`

```
BaseResult AddToBlackList( Guid sessionID, Guid personID )
```

Параметры:

`Guid` sessionID

Уникальный ключ сессии.

`Guid` personID

Уникальный ключ субъекта доступа.

Результат: Возвращает `BaseResult`.

Описание: Добавляет субъект доступа с указанным ID в черный список.

Функция `RemoveFromBlackList`

```
BaseResult RemoveFromBlackList( Guid sessionID, Guid personID )
```

Параметры:

`Guid` sessionID

Уникальный ключ сессии.

`Guid` personID

Уникальный ключ субъекта доступа.

Результат: Возвращает BaseResult.

Описание: Удаляет субъект доступа с указанным ID из черного списка.

РЕДАКТИРОВАНИЕ СУБЪЕКТА ДОСТУПА

Функция CreatePerson

```
GuidResult CreatePerson( Guid sessionId, Person person )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Person person</code> | Данные сотрудника. |

Результат: Возвращает GuidResult.

Описание: Создает сотрудника с указанными данными. В качестве параметра person может быть Person или PersonWithPhoto. Возвращает ключ вновь созданного сотрудника. Если поле ID в структуре person равен Guid.Empty (00000000-0000-0000-0000-000000000000), то ID генерируется автоматически и возвращается. Иначе используется заданное значение ID.

Функция CreateVisitor

```
GuidResult CreateVisitor( Guid sessionId, Person person )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Person person</code> | Данные посетителя. |

Результат: Возвращает GuidResult.

Описание: Создает посетителя с указанными данными. В качестве параметра person может быть Person или PersonWithPhoto. Возвращает ключ вновь созданного посетителя. Если поле ID в структуре person равен Guid.Empty (00000000-0000-0000-0000-000000000000), то ID генерируется автоматически и возвращается. Иначе используется заданное значение ID.

Функция CreateVehicle

```
GuidResult CreateVehicle( Guid sessionId, Person person )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Person person</code> | Данные автомобиля. |

Результат: Возвращает GuidResult.

Описание: Создает автомобиль с указанными данными. В качестве параметра person может быть Person или PersonWithPhoto. Возвращает ключ вновь созданного автомобиля. Если поле ID в структуре person равен Guid.Empty (00000000-0000-0000-0000-000000000000), то ID генерируется автоматически и возвращается. Иначе используется заданное значение ID.

Функция OpenPersonEditingSession

```
GuidResult OpenPersonEditingSession( Guid sessionId, Guid personID )
```

Параметры:

| | |
|-----------------------------|-----------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid personID</code> | Уникальный ключ субъекта доступа. |

Результат: Возвращает `GuidResult`.

Описание: Открывает сессию редактирования субъекта доступа. Возвращает ключ вновь созданной сессии.

Функция `ClosePersonEditingSession`

```
void ClosePersonEditingSession( Guid personEditSessionID )
```

Параметры:

| | |
|---------------------------------------|---|
| <code>Guid personEditSessionID</code> | Уникальный ключ сессии редактирования субъекта доступа. |
|---------------------------------------|---|

Результат: –

Описание: Закрывает сессию редактирования субъекта доступа.

Функция `SavePerson`

```
BaseResult SavePerson( Guid personEditSessionID, BasePerson person )
```

Параметры:

| | |
|---------------------------------------|---|
| <code>Guid personEditSessionID</code> | Уникальный ключ сессии редактирования субъекта доступа. |
| <code>BasePerson person</code> | Данные субъекта доступа. |

Результат: Возвращает `BaseResult`.

Описание: Изменяет данные субъекта доступа. В качестве параметра `person` может служить `BasePerson`, `Person`, `PersonWithPhoto`.

При передаче в запросе в качестве данных субъекта доступа структуры `Person` или `PersonWithPhoto` необходимо в обязательном порядке указывать название класса в атрибуте `xsi:type` элемента `person`.

В случае удачного сохранения сессия редактирования субъекта доступа закрывается.

Функция `SetPersonPhoto`

```
BaseResult SetPersonPhoto( Guid personEditSessionID, byte[] photoByteArray )
```

Параметры:

| | |
|---------------------------------------|---|
| <code>Guid personEditSessionID</code> | Уникальный ключ сессии редактирования субъекта доступа. |
| <code>byte[] photoByteArray</code> | Фото субъекта доступа. |

Результат: Возвращает `BaseResult`.

Описание: Сохраняет фото субъекта доступа. В случае удачного сохранения сессия редактирования субъекта доступа закрывается.

Функция SetPersonOrgUnit

`BaseResult SetPersonOrgUnit(Guid personEditSessionID, Guid orgUnitID)`

Параметры:

| | |
|---|---|
| <code>Guid</code> <code>personEditSessionID</code> | Уникальный ключ сессии редактирования субъекта доступа. |
| <code>Guid</code> <code>orgUnitID</code> | Уникальный ключ подразделения. |

Результат: Возвращает `BaseResult`.

Описание: Устанавливает подразделение субъекта доступа. В случае удачного сохранения сессия редактирования субъекта доступа закрывается.

Функция SetPersonExtraFieldValue

`BaseResult SetPersonExtraFieldValue(Guid personEditSessionID, Guid templateID, object value)`

Параметры:

| | |
|---|---|
| <code>Guid</code> <code>personEditSessionID</code> | Уникальный ключ сессии редактирования субъекта доступа. |
| <code>Guid</code> <code>templateID</code> | Уникальный ключ шаблона дополнительного поля. |
| <code>object</code> <code>value</code> | Значение дополнительного поля. |

Результат: Возвращает `BaseResult`.

Описание: Устанавливает новое значение указанного дополнительного поля субъекта доступа.

В запросе, для элемента *value* необходимо указывать тип значения в атрибуте *xsi:type*. Например, *xsi:type="xsd:string"*.

Пример корректного запроса:

```
<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
      <SetPersonExtraFieldValue
        xmlns="http://parsec.ru/Parsec3IntergationService">
        <personEditSessionID>b2aea303-f736-4a29-b835-
670c2384551f</personEditSessionID>
        <templateID>4d7371ef-e856-4371-88da-
0365277c2032</templateID>
        <value xsi:type="xsd:string">valid value</value>
      </SetPersonExtraFieldValue>
    </soap:Body>
  </soap:Envelope>
```

Функция SetPersonExtraFieldValues

```
BaseResult SetPersonExtraFieldValues( Guid personEditSessionID,  
ExtraFieldValue[] values )
```

Параметры:

| | |
|---|---|
| <code>Guid</code> <code>personEditSessionID</code> | Уникальный ключ сессии редактирования субъекта доступа. |
| <code>ExtraFieldValue[]</code> <code>values</code> | Массив значений дополнительных полей. |

Результат: Возвращает BaseResult.

Описание: Устанавливает значения для дополнительных полей.

В запросе, в каждом элементе массива *values* необходимо указывать тип значения в атрибуте *xsi:type*. Например, *xsi:type="xsd:string"* (см. пример в описании ф-ии [SetPersonExtraFieldValue](#)).

Функция DeletePerson

```
BaseResult DeletePerson( Guid sessionID, Guid personID )
```

Параметры:

| | |
|--|-----------------------------------|
| <code>Guid</code> <code>sessionID</code> | Уникальный ключ сессии. |
| <code>Guid</code> <code>personID</code> | Уникальный ключ субъекта доступа. |

Результат: Возвращает BaseResult.

Описание: Удаляет субъект доступа с указанным ключом.

Функция BlockPerson

```
BaseResult BlockPerson( Guid personEditSessionID, int typeBlock )
```

Параметры:

| | |
|---|--|
| <code>Guid</code> <code>personEditSessionID</code> | Уникальный ключ сессии редактирования данных субъекта доступа. |
| <code>int</code> <code>typeBlock</code> | Тип блокировки. |

Результат: Возвращает BaseResult.

Описание: Блокирует субъект доступа в соответствии с указанным типом блокировки.

Значения типа блокировки описаны в таблице:

| Значение <code>typeBlock</code> | Описание |
|------------------------------------|--|
| 1 | Установка привилегии «Выход запрещен». |
| 2 | Установка привилегии «Вход запрещен». |
| 3 | Установка привилегии «Вход запрещен» и «Выход запрещен». |

В случае удачного выполнения функции сессия редактирования данных субъекта доступа закрывается.

Функция UnblockPerson

```
BaseResult UnblockPerson( Guid personEditSessionID )
```


Параметры:

| | |
|----------------------------------|---------------------------------------|
| <code>Guid</code> | Уникальный ключ сессии редактирования |
| <code>personEditSessionID</code> | данных субъекта доступа. |

Результат: Возвращает `BaseResult`.

Описание: Снимает все блокировки с субъекта доступа, заблокированного ранее.

В случае удачного выполнения функции сессия редактирования субъекта доступа закрывается.

ТОПОЛОГИЯ

Функция `GetRootTerritory`

`Territory` `GetRootTerritory(Guid sessionId)`

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает корневую `Territory`.

Описание: Данная функция предназначена для получения корневого объекта дерева территорий.

Функция `GetTerritoriesHierarchy`

`Territory[]` `GetTerritoriesHierarchy(Guid sessionId)`

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив `Territory`, значения могут быть `TerritoryWithComponent`.

Описание: Возвращает полную иерархию территорий.

Функция `GetTerritorySubItems`

`BaseTerritory[]` `GetTerritorySubItems(Guid sessionId, Guid TerraID)`

Параметры:

| | |
|-----------------------------|-----------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid TerraID</code> | Уникальный ключ территории. |

Результат: Возвращает массив `BaseTerritory`, значения могут быть `TerritoryWithComponent`.

Описание: Возвращает список территорий, принадлежащих территории с указанным ключом.

Функция `GetTerritory`

`Territory` `GetTerritory(Guid sessionId, Guid territoryID)`

Параметры:

| | |
|-------------------------------|----------------------------------|
| <code>Guid sessionId</code> | Уникальный ключ открытой сессии. |
| <code>Guid territoryID</code> | Уникальный ключ территории. |

Результат: Возвращает Territory, значение может быть TerritoryWithComponent.

Описание: Данная функция предназначена для получения описания территории по ее ключу.

ИДЕНТИФИКАТОРЫ И ДОСТУП

Функция GetPersonIdentifiers

```
Identifier[] GetPersonIdentifiers( Guid sessionId, Guid personID )
```

Параметры:

| | |
|----------------|-----------------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Guid personID | Уникальный ключ субъекта доступа. |

Результат: Массив Identifier, элементы массива могут быть Identifier или IdentifierTemp.

Описание: Возвращает массив идентификаторов для указанного субъекта доступа.

Функция DeleteIdentifier

```
BaseResult DeleteIdentifier( Guid sessionId, string Code )
```

Параметры:

| | |
|----------------|-------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| string Code | Код идентификатора. |

Результат: Возвращает BaseResult.

Описание: Удаляет идентификатор с указанным кодом.

Функция AddPersonIdentifier

```
BaseResult AddPersonIdentifier( Guid personEditSessionID,  
BaseIdentifier identifier )
```

Параметры:

| | |
|------------------------------|---|
| Guid personEditSessionID | Уникальный ключ сессии редактирования субъекта доступа. |
| BaseIdentifier identifier | Параметры идентификатора. |

Результат: Возвращает BaseResult.

Описание: Добавляет идентификатор субъекту доступа или изменяет данные уже имеющегося идентификатора. В качестве параметра identifier могут служить BaseIdentifier, Identifier, IdentifierTemp. Элемент IS_PRIMARY используется только в позитивном смысле, т.е. при установке этого признака, соответствующий идентификатор становится первичным, а при снятии этого признака у первичного идентификатора при сохранении в смысле первичности идентификатора ничего не меняется. Элемент PRIVILEGE_MASK игнорируется.

При передаче в запросе в качестве идентификатора структуры Identifier или IdentifierTemp необходимо в обязательном порядке указывать в элементе identifier атрибут *xsi:type="Identifier"* либо *xsi:type="IdentifierTemp"* соответственно.

Пример корректного запроса:

50

```

<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <AddPersonIdentifier xmlns="http://parsec.ru/ParsecIntergationService">
            <personEditSessionID>a0c8c558-0893-4fb6-b2ea-
528d2f08edd2</personEditSessionID>
            <identifier xsi:type="Identifier">
                <CODE>11111112</CODE>
                <PERSON_ID>be4d3ce8-d830-4796-9197-7fa65c78d13f</PERSON_ID>
                <IS_PRIMARY>>false</IS_PRIMARY>
                <ACCGROUP_ID>11111111-2222-3333-4444-
555555555555</ACCGROUP_ID>
            </identifier>
        </AddPersonIdentifier>
    </soap:Body>
</soap:Envelope>

```

Функция ChangePersonIdentifier

```

BaseResult ChangePersonIdentifier( Guid personEditSessionID,
BaseIdentifier identifier )

```

Параметры:

| | |
|------------------------------|---|
| Guid personEditSessionID | Уникальный ключ сессии редактирования субъекта доступа. |
| BaseIdentifier identifier | Параметры идентификатора. |

Результат: Возвращает BaseResult.

Описание: Изменяет параметры существующего идентификатора. В качестве параметра identifier могут служить BaseIdentifier, Identifier, IdentifierTemp. Элемент IS_PRIMARY используется только в позитивном смысле, т.е. при установке этого признака, соответствующий идентификатор становится первичным, а при снятии этого признака у первичного идентификатора при сохранении в смысле первичности идентификатора ничего не меняется. Элемент PRIVILEGE_MASK игнорируется.

При передаче в запросе в качестве идентификатора структуры Identifier или IdentifierTemp необходимо в обязательном порядке указывать в элементе identifier атрибут xsi:type="Identifier" либо xsi:type="IdentifierTemp" соответственно.

Пример корректного запроса:

```

<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>

```

```

    <ChangePersonIdentifier xmlns="http://parsec.ru/ParsecIntergationService">
      <personEditSessionID>a0c8c558-0893-4fb6-b2ea-
528d2f08edd2</personEditSessionID>
      <identifier xsi:type="Identifier">
        <CODE>11111112</CODE>
        <PERSON_ID>be4d3ce8-d830-4796-9197-7fa65c78d13f</PERSON_ID>
        <IS_PRIMARY>false</IS_PRIMARY>
        <ACCGROUP_ID>11111111-2222-3333-4444-
555555555555</ACCGROUP_ID>
      </identifier>
    </ ChangePersonIdentifier >
  </soap:Body>
</soap:Envelope>

```

Функция SetIdentifierPrivileges

```

BaseResult SetIdentifierPrivileges( Guid sessionID, string cardCode,
long privilegesMask )

```

Параметры:

| | |
|----------------------------------|---------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>string cardCode</code> | Код идентификатора. |
| <code>long privilegesMask</code> | Битовая маска привилегий. |

Результат: Возвращает BaseResult.

Описание: Функция устанавливает набор привилегий с помощью битовой маски. Значения битов маски описаны в таблице:

| № бита | Описание |
|--------|--|
| 0 | Выключение звука двери. |
| 1 | Управление охраной. |
| 2 | Проход при блокировке. |
| 3 | Прием тревоги. |
| 4 | Постановка на охрану. |
| 5 | Снятие с охраны. |
| 6 | Проход при антипассбеке. |
| 7 | Гостевая карта. |
| 8 | Карта с привилегиями. |
| 9 | Выход запрещен. |
| 10 | Выход вне временного профиля разрешен. |
| 11 | Управление доступом. |
| 12 | - |
| 13 | Карта владельца. |

| | |
|----|---|
| 14 | Не использовать счетчик проходов. |
| 15 | Вход запрещен. |
| 16 | Проход без сопровождения запрещен. |
| 17 | Строго контролировать время возврата ключа (ключница) |

Пример: для установки идентификатору привилегий «Проход при блокировке» и «Проход при антипассбеке» значение параметра *privilegesMask* должно быть равно $2^2 + 2^6 = 4 + 64 = 68$ $_{10} = 1000100_2$

Функция GetIdentifierExtraData

```
IdentifierExData GetIdentifierExtraData( Guid sessionId, string
cardCode )
```

Параметры:

| | |
|------------------------------|---|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>string cardCode</code> | Код идентификатора карты в 16-ричном формате. |

Результат: Возвращает IdentifierExData.

Описание: Выдает сведения о дополнительных свойствах идентификатора.

Функция SetIdentifierExtraData

```
BaseResult SetIdentifierExtraData( Guid sessionId, string cardCode,
IdentifierExData exData )
```

Параметры:

| | |
|--------------------------------------|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>string cardCode</code> | Код идентификатора карты в 16-ричном формате. |
| <code>IdentifierExData exData</code> | Параметры дополнительных свойств идентификатора. |

Результат: Возвращает BaseResult.

Описание: Устанавливает значения дополнительных свойств идентификатора.

Функция GetPassageRoles

```
PassageRole[]GetPassageRoles( Guid sessionId )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив PassageRole.

Описание: Выдает список ролей группового прохода.

Функция CreatePassageRole

```
GuidResult CreatePassageRole( Guid sessionId, PassageRole role )
```

Параметры:

| | |
|-------------------------------|-----------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>PassageRole role</code> | Параметры создаваемой роли. |

Результат: Возвращает `GuidResult`.

Описание: Создает роль группового прохода с указанными параметрами. Возвращает ключ созданной роли. Если поле ID в структуре `PassageRole` равен `Guid.Empty` (00000000-0000-0000-0000-000000000000), то ID генерируется автоматически и возвращается. Иначе используется заданное значение ID.

Функция **SavePassageRole**

```
BaseResult SavePassageRole( Guid sessionID, PassageRole role )
```

Параметры:

| | |
|-------------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>PassageRole role</code> | Параметры роли. |

Результат: Возвращает `BaseResult`.

Описание: Изменяет параметры роли.

Функция **DeletePassageRole**

```
BaseResult DeletePassageRole( Guid sessionID, Guid roleID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid roleID</code> | Уникальный ключ роли. |

Результат: Возвращает `BaseResult`.

Описание: Удаляет роль группового прохода с указанным ключом.

Функция **GetUnique4bCardCode**

```
StringResult GetUnique4bCardCode ( Guid sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает `StringResult`.

Описание: Возвращает уникальный неиспользуемый код идентификатора длиной 4 байта.

Функция **GetCardCodeFromUID**

```
StringResult GetCardCodeFromUID ( Guid sessionID, string UID, bool reverseByteOrder )
```

Параметры:

| | |
|------------------------------------|---------------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>string UID</code> | UID карты в 16-ричном формате. |
| <code>bool reverseByteOrder</code> | Признак обратного порядка байт в UID. |

Результат: Возвращает StringResult.

Описание: Возвращает код карты Mifare на основе UID. Возвращаемый код зависит от значений параметров, задаваемых в разделе «Настройка настольных считывателей».

Функция GenerateParsecQRCode

```
StringResult GenerateParsecQRCode ( Guid sessionId, string cardCode )
```

Параметры:

| | |
|-----------------|-------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| string cardCode | Код идентификатора. |

Результат: Возвращает StringResult.

Описание: Возвращает строку, содержащую зашифрованный код идентификатора, предназначенную для генерации QR-кода. Сгенерированный QR-код может использоваться считывателями Parsec.

Функция GetAdvancedQRGroups

```
QRAdvancedGroup[] GetAdvancedQRGroups( Guid sessionId )
```

Параметры:

| | |
|----------------|-------------------------|
| Guid sessionId | Уникальный ключ сессии. |
|----------------|-------------------------|

Результат: Возвращает массив QRAdvancedGroup.

Описание: Возвращает набор групп контроллеров расширенных QR-кодов.

Функция GenerateAdvancedParsecQRCode

```
StringResult GenerateAdvancedParsecQRCode( Guid sessionId,  
QRAdvancedData qrData )
```

Параметры:

| | |
|-----------------------|--------------------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| QRAdvancedData qrData | Права доступа генерируемого QR-кода. |

Результат: Возвращает StringResult.

Описание: Возвращает строку, предназначенную для генерации QR-кода. В строке в зашифрованном виде содержатся заданные права доступа. Сгенерированный расширенный QR-код может использоваться считывателями Parsec.

РАСПИСАНИЯ И ГРУППЫ ДОСТУПА

Функция GetAccessSchedules

```
Schedule[] GetAccessSchedules( Guid sessionId )
```

Параметры:

| | |
|----------------|-------------------------|
| Guid sessionId | Уникальный ключ сессии. |
|----------------|-------------------------|

Результат: Возвращает массив Schedule, значение может быть AccessSchedule.

Описание: Выдает набор расписаний доступа системы

Функция GetWorktimeSchedules

```
Schedule[] GetWorktimeSchedules( Guid sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив `Schedule`, значение может быть `WorktimeSchedule`.

Описание: Выдает набор расписаний рабочего времени.

Функция GetScheduleIntervals

```
TimeInterval[] GetScheduleIntervals( Guid sessionID, Guid scheduleID,  
DateTime from, DateTime to )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

| | |
|------------------------------|-----------------------------|
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
|------------------------------|-----------------------------|

| | |
|----------------------------|-----------------------------|
| <code>DateTime from</code> | Начальная дата для анализа. |
|----------------------------|-----------------------------|

| | |
|--------------------------|----------------------------|
| <code>DateTime to</code> | Конечная дата для анализа. |
|--------------------------|----------------------------|

Результат: Возвращает массив `TimeInterval`, значения могут быть `WorktimeInterval`.

Описание: Выдает набор интервалов указанного расписания, ограниченный начальной и конечной датой. Функцией принимается во внимание только DATE часть параметров.

Функция CreateAccessSchedule

```
GuidResult CreateAccessSchedule( Guid sessionID, AccessSchedule  
schedule, ScheduleDay[] days )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

| | |
|--|-----------------------|
| <code>AccessSchedule schedule</code> | Параметры расписания. |
|--|-----------------------|

| | |
|-------------------------------------|--|
| <code>ScheduleDay[] days</code> | Массив шаблонов дней цикла расписания. |
|-------------------------------------|--|

Результат: Возвращает `GuidResult`.

Описание: Создает новое расписание доступа в системе с указанными параметрами и заданным циклом. Возвращает ключ вновь созданного расписания доступа.

При создании расписания необходимо строго соблюдать следующие условия:

- Для недельного расписания доступа обязателен тип применения праздников «Применять с заменой».
- Для недельного расписания рабочего времени не может быть установлен тип применения праздников «Применять со вставкой».
- Для недельных расписаний массив шаблонов дней `days` должен содержать 7 элементов.

- Элементы массива *days* должны иметь индексы дня в цикле расписания. Индекс первого дня в цикле имеет значение «1». Все элементы цикла должны иметь одинаковую дату.
- Для недельных расписаний дата начала цикла должна приходиться на понедельник.
- В цикле недельного расписания доступа не должно быть более трех уникальных шаблонов дней, один из которых – выходной (не содержит временных интервалов).

Функция CreateWorktimeSchedule

```
GuidResult CreateWorktimeSchedule( Guid sessionId, WorktimeSchedule
schedule, ScheduleDay[] days )
```

Параметры:

| | |
|--|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>WorktimeSchedule schedule</code> | Параметры расписания. |
| <code>ScheduleDay[] days</code> | Массив шаблонов дней цикла расписания. |

Результат: Возвращает GuidResult.

Описание: Создает новое расписание рабочего времени в системе с указанными параметрами и заданным циклом. Возвращает ключ вновь созданного расписания рабочего времени. При создании необходимо соблюдать правила, описанные в разделе функции [CreateAccessSchedule](#).

Функция GetSchedule

```
Schedule GetSchedule( Guid sessionId, Guid scheduleID )
```

Параметры:

| | |
|------------------------------|-----------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |

Результат: Возвращает Schedule, результат может быть AccessSchedule или WorktimeSchedule.

Описание: Возвращает информацию о расписании с указанным ключом.

Функция SaveSchedule

```
BaseResult SaveSchedule( Guid sessionId, Schedule schedule )
```

Параметры:

| | |
|--------------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Schedule schedule</code> | Параметры расписания. |

Результат: Возвращает BaseResult.

Описание: Изменяет параметры расписания. Параметр schedule может быть AccessSchedule или WorktimeSchedule.

При передаче в запросе в качестве данных расписания структуры AccessSchedule или WorktimeSchedule необходимо в обязательном порядке указывать название класса в атрибуте *xsi:type* элемента *schedule*.

Функция DeleteSchedule

```
BaseResult DeleteSchedule( Guid sessionId, Guid scheduleID )
```

Параметры:

| | |
|------------------------------|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания, которое нужно удалить. |

Результат: Возвращает BaseResult.

Описание: Удаляет расписание с указанным ключом.

Функция GetScheduleDetails

```
ScheduleDay[] GetScheduleDetails( Guid sessionId, Guid scheduleID )
```

Параметры:

| | |
|------------------------------|-----------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |

Результат: Возвращает массив ScheduleDay. Массив может содержать ScheduleFix.

Описание: Возвращает информацию о всех шаблонах дней и днях-поправках, содержащихся в расписании с указанным ключом.

Функция SetScheduleDays

```
BaseResult SetScheduleDays( Guid sessionId, Guid scheduleID,  
ScheduleDay[] days )
```

Параметры:

| | |
|---------------------------------|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>ScheduleDay[] days</code> | Массив шаблонов дней цикла расписания. |

Результат: Возвращает BaseResult.

Описание: Добавляет новый или изменяет существующий цикл (при совпадении даты начала цикла) расписания с указанным ключом. Параметр days должен соответствовать правилам, описанным в разделе функции [CreateAccessSchedule](#).

Функция SetScheduleFix

```
BaseResult SetScheduleFix( Guid sessionId, Guid scheduleID,  
ScheduleFix[] fixes )
```

Параметры:

| | |
|----------------------------------|----------------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>ScheduleFix[] fixes</code> | Массив дней-поправок расписания. |

Результат: Возвращает BaseResult.

Описание: Добавляет или изменяет существующие дни-поправки (при совпадении даты) в расписании с указанным ключом. Не может быть применена для недельного расписания доступа.

Функция DeleteScheduleDays

```
BaseResult DeleteScheduleDays( Guid sessionId, Guid scheduleID,
DateTime[] days )
```

Параметры:

| | |
|-----------------|-----------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Guid scheduleID | Уникальный ключ расписания. |
| DateTime[] days | Массив дат. |

Результат: Возвращает BaseResult.

Описание: Удаляет цикл расписания и дни-поправки, если дата в массиве days совпадает с датой начала цикла или датой поправки. Недопустимо удаление последнего цикла в расписании.

Функция GetHolidays

```
Holiday[] GetHolidays( Guid sessionId )
```

Параметры:

| | |
|----------------|-------------------------|
| Guid sessionId | Уникальный ключ сессии. |
|----------------|-------------------------|

Результат: Возвращает массив holiday.

Описание: Выдает список праздников, существующих в системе.

Функция SetHolidays

```
BaseResult SetHolidays( Guid sessionId, Holiday[] holidays )
```

Параметры:

| | |
|--------------------|--------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Holiday[] holidays | Массив праздничных дней. |

Результат: Возвращает BaseResult.

Описание: Добавляет или изменяет существующие (при совпадении дат) праздничные дни.

Функция DeleteHolidays

```
BaseResult DeleteHolidays ( Guid sessionId, Holiday[] holidays )
```

Параметры:

| | |
|--------------------|---------------------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Holiday[] holidays | Массив праздничных дней для удаления. |

Результат: Возвращает BaseResult.

Описание: Удаляет праздничные дни с указанными параметрами.

Функция **GetAccessGroups**

```
AccessGroup[] GetAccessGroups( Guid sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив `AccessGroup`.

Описание: Возвращает массив доступных групп доступа.

Функция **CreateTempAccessGroup**

```
GuidResult CreateTempAccessGroup( Guid sessionID, Guid scheduleID,  
Guid[] territories )
```

Параметры:

| | |
|---------------------------------|--------------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>Guid[] territories</code> | Массив уникальных ключей территорий. |

Результат: Возвращает `GuidResult`.

Описание: Создает временную группу доступа типа «Подсистема доступа «Parsec» по заданному расписанию для доступа на заданные территории. Группа является временной и имеет ограниченное «время жизни». Группа недействительна без привязки к идентификатору сотрудника.

Функция **CreateAccessGroup**

```
GuidResult CreateAccessGroup( Guid sessionID, string groupName, Guid  
scheduleID, Guid[] territories )
```

Параметры:

| | |
|---------------------------------|--|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>string groupName</code> | Наименование группы доступа. Должно быть уникальным. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>Guid[] territories</code> | Массив ключей территорий. |

Результат: Возвращает `GuidResult`.

Описание: Создает группу доступа типа «Подсистема доступа «Parsec» по указанному расписанию для доступа на выбранные территории. Группа доступа будет иметь указанное в функции имя.

Функция **CreateVehicleTempAccessGroup**

```
GuidResult CreateVehicleTempAccessGroup( Guid sessionID, Guid  
scheduleID, Guid[] territories )
```

Параметры:

| | |
|---------------------------------|--------------------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>Guid[] territories</code> | Массив уникальных ключей территорий. |

Результат: Возвращает GuidResult.

Описание: Создает временную группу доступа типа «Автомобильный номер» по заданному расписанию для доступа на заданные территории. Группа является временной и имеет ограниченное «время жизни». Группа недействительна без привязки к идентификатору сотрудника.

Функция CreateVehicleAccessGroup

```
GuidResult CreateVehicleAccessGroup( Guid sessionId, string groupName, Guid scheduleID, Guid[] territories )
```

Параметры:

| | |
|---------------------------------|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>string groupName</code> | Наименование группы доступа. Должно быть уникальным. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>Guid[] territories</code> | Массив ключей территорий. |

Результат: Возвращает GuidResult.

Описание: Создает группу доступа типа «Автомобильный номер» по указанному расписанию для доступа на выбранные территории. Группа доступа будет иметь указанное в функции имя.

Функция DeleteAccessGroup

```
BaseResult DeleteAccessGroup( Guid sessionId, Guid accessGroupID )
```

Параметры:

| | |
|---------------------------------|---------------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid accessGroupID</code> | Уникальный ключ группы доступа. |

Результат: Возвращает BaseResult.

Описание: Удаляет группу доступа с указанным ключом.

Для удаления доступны только группы доступа типов «Подсистема доступа «Parsec» и «Автомобильный номер».

Функция AddSubAccessGroup

```
GuidResult AddSubAccessGroup( Guid sessionId, Guid accessGroupID, Guid scheduleID, Guid[] territories )
```

Параметры:

| | |
|---------------------------------|---------------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid accessGroupID</code> | Уникальный ключ группы доступа. |
| <code>Guid scheduleID</code> | Уникальный ключ расписания. |
| <code>Guid[] territories</code> | Массив ключей территорий. |

Результат: Возвращает GuidResult.

Описание: Добавляет к существующей группе доступа группу компонент с указанным расписанием для доступа на выбранные территории.

Если указанная территория уже используется в какой-либо группе компонент (в пределах одной группы доступа), то она будет удалена из этой группы компонент.

Для изменения доступны только группы доступа типов «Подсистема доступа «Parsec» и «Автомобильный номер».

Функция DeleteSubAccessGroup

```
BaseResult DeleteSubAccessGroup( Guid sessionId, Guid accessGroupID, Guid subGroupID )
```

Параметры:

| | |
|--------------------|-----------------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Guid accessGroupID | Уникальный ключ группы доступа. |
| Guid subGroupID | Уникальный ключ группы компонент. |

Результат: Возвращает BaseResult.

Описание: Удаляет группу компонент с указанным ключом.

Для изменения доступны только группы доступа типов «Подсистема доступа «Parsec» и «Автомобильный номер».

Функция GetSubAccessGroups

```
SubAccessGroup[] GetSubAccessGroups( Guid sessionId, Guid accessGroupID )
```

Параметры:

| | |
|--------------------|---------------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Guid accessGroupID | Уникальный ключ группы доступа. |

Результат: Возвращает массив SubAccessGroup.

Описание: Выдает список групп компонент (подгрупп) для указанной группы доступа.

Функция GetInheritedAccessGroups

```
Guid[] GetInheritedAccessGroups( Guid sessionId, Guid accessGroupID )
```

Параметры:

| | |
|--------------------|---------------------------------|
| Guid sessionId | Уникальный ключ сессии. |
| Guid accessGroupID | Уникальный ключ группы доступа. |

Результат: Возвращает массив ключей групп доступа.

Описание: Выдает список ключей групп доступа, вложенных для указанной группы доступа. Ключи вложенных групп возвращаются в порядке убывания приоритета наследования.

Функция SetInheritedAccessGroups

```
BaseResult SetInheritedAccessGroup( Guid sessionId, Guid accessGroupID, Guid[] inheritedGroups )
```

Параметры:

| | |
|-------------------------------------|---------------------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid accessGroupID</code> | Уникальный ключ группы доступа. |
| <code>Guid[] inheritedGroups</code> | Массив ключей вложенных групп доступа |

Результат: Возвращает `BaseResult`.

Описание: Задает вложенные группы доступа для указанной группы. Приоритет наследования определяется порядком элементов в массиве.

Для отмены наследования необходимо передать пустой массив.

Для изменения доступны только группы доступа типов «Подсистема доступа «Parsec» и «Автомобильный номер».

РАБОТА С ЗАЯВКАМИ БЮРО ПРОПУСКОВ

Функция `GetAcceptedVisitorRequests`

```
VisitorRequest[] GetAcceptedVisitorRequests( Guid sessionId )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив `VisitorRequest`.

Описание: Выдает набор заявок бюро пропусков со статусом «Одобрены».

Функция `FindVisitorRequest`

```
VisitorRequest FindVisitorRequest( Guid sessionId, int requestNumber )
```

Параметры:

| | |
|--------------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>int requestNumber</code> | Номер заявки. |

Результат: Возвращает `VisitorRequest`, либо `null`, если заявка с указанным номером не найдена.

Описание: Выполняет поиск заявки с указанным номером среди заявок со статусом «Одобрена».

Функция `ActivateVisitorRequest`

```
BaseResult ActivateVisitorRequest( Guid sessionId, Guid requestID, string cardCode )
```

Параметры:

| | |
|------------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid requestID</code> | Уникальный ключ заявки. |
| <code>String cardCode</code> | Код идентификатора. |

Результат: Возвращает `BaseResult`.

Описание: Выдает посетителю идентификатор с указанным кодом из пула. Данная функция применима только к заявкам со статусом «Одобрена». Переводит заявку в состояние «Активная».

Функция CreateVisitorRequest

```
VisitorRequest CreateVisitorRequest( Guid sessionId, VisitorRequest request )
```

Параметры:

| | |
|-------------------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>VisitorRequest request</code> | Параметры заявки. |

Результат: Возвращает VisitorRequest.

Описание: Создает заявку на посетителя и возвращает информацию о ней. В параметре обязательно должны быть заполнены PERSON_ID, ORGUNIT_ID, ADMIT_START, ADMIT_END. Все остальные параметры необязательны. Заявка создается со статусом «Ожидание согласования».

Функция GetVisitorRequest

```
VisitorRequest GetVisitorRequest( Guid sessionId, Guid requestId )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid requestId</code> | Уникальный ключ заявки. |

Результат: Возвращает VisitorRequest.

Описание: Возвращает информацию о затребованной заявке.

Функция SaveVisitorRequest

```
BaseResult SaveVisitorRequest( Guid sessionId, VisitorRequest request )
```

Параметры:

| | |
|-------------------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>VisitorRequest request</code> | Параметры заявки. |

Результат: Возвращает BaseResult.

Описание: Изменяет параметры в существующей заявке.

Функция DeleteIssuedVisitorRequest

```
BaseResult DeleteIssuedVisitorRequest( Guid sessionId, Guid requestId )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid requestId</code> | Уникальный ключ заявки. |

Результат: Возвращает BaseResult.

Описание: Удаляет заявку на посетителя. Может быть удалена только заявка со статусом «Ожидание согласования».

Функция GetIssuedVisitorRequests

```
VisitorRequest[] GetIssuedVisitorRequests( Guid sessionID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает массив VisitorRequest.

Описание: Возвращает массив заявок, имеющих статус «Ожидание согласования». Если заявок не найдено, то возвращается пустой массив.

Функция GetVisitorRequests

```
VisitorRequest[] GetVisitorRequests ( Guid sessionID, Guid orgUnitID, DateTime from, bool issued, bool accepted, bool declined, bool active, bool completed )
```

Параметры:

| | |
|-----------------------------|---|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid orgUnitID</code> | Уникальный ключ подразделения. |
| <code>DateTime from</code> | Начальная дата для отбора заявок. |
| <code>bool issued</code> | Заявки со статусом «Ожидание согласования». |
| <code>bool accepted</code> | Заявки со статусом «Одобрена». |
| <code>bool declined</code> | Заявки со статусом «Отклонена». |
| <code>bool active</code> | Заявки со статусом «Выдан пропуск». |
| <code>bool completed</code> | Заявки со статусом «Закрыта». |

Результат: Массив VisitorRequest.

Описание: Выдает все заявки в указанном подразделении, отобранные по дате и статусам. Функцией принимается во внимание только DATE часть параметра from.

Функция CloseAllActiveVisitorRequests

```
BaseResult CloseAllActiveVisitorRequests( Guid sessionID, Guid visitorID )
```

Параметры:

| | |
|-----------------------------|-----------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid visitorID</code> | Уникальный ключ посетителя. |

Результат: Возвращает BaseResult.

Описание: Ищет и закрывает все активные заявки, связанные с указанным посетителем, идентификатор после закрытия заявки отвязывается от посетителя и возвращается в пул карт бюро пропусков.

Функция CloseVisitorRequest

```
BaseResult CloseVisitorRequest( Guid sessionID, Guid requestID )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid requestID</code> | Уникальный ключ заявки. |

Результат: Возвращает `BaseResult`.

Описание: Закрывает заявку с указанным кодом. Заккрытие выполняется только для активных заявок (статус «Пропуск выдан»). Связанный идентификатор после закрытия заявки отвязывается от посетителя и возвращается в пул карт бюро пропусков.

Функция `GetPersonVisitorRequests`

```
VisitorRequest[] GetPersonVisitorRequests( Guid sessionID,  
Guid visitorID, bool issued, bool accepted, bool declined,  
bool active, bool completed )
```

Параметры:

| | |
|-----------------------------|--|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid visitorID</code> | Уникальный ключ посетителя. |
| <code>bool issued</code> | Заявки со статусом «Ожидание согласования» |
| <code>bool accepted</code> | Заявки со статусом «Одобрена» |
| <code>bool declined</code> | Заявки со статусом «Отклонена» |
| <code>bool active</code> | Заявки со статусом «Выдан пропуск» |
| <code>bool completed</code> | Заявки со статусом «Закрыта» |

Результат: Возвращает массив `VisitorRequest`.

Описание: Возвращает массив всех заявок, доступных по области видимости, оформлявшихся на указанного посетителя, с возможностью отбора/фильтрации по всем типам статусов заявок. Если заявок не найдено, то возвращается пустой массив.

СОБЫТИЯ СИСТЕМЫ

Функция `GetEvents`

```
EventsHistoryResult GetEvents( Guid sessionID, Guid TerritoryID, Guid  
PersNodeID, DateTime dtFrom, DateTime dtTo )
```

Параметры:

| | |
|-------------------------------|--|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid TerritoryID</code> | Уникальный ключ территории. |
| <code>Guid PersNodeID</code> | Уникальный ключ персонала (подразделения или сотрудника). |
| <code>DateTime dtFrom</code> | Дата с... |
| <code>DateTime dtTo</code> | Дата по... |

Результат: Возвращает `EventsHistoryResult`.

Описание: Возвращает историю событий для указанных территорий и персонала за указанный период времени. Для ключа территории и ключа

персонала можно использовать специальное значение – Guid.Empty (00000000-0000-0000-0000-000000000000), при указании которого в качестве критерия будет взят корень соответствующей иерархии.

Функция *GetEvents* устарела. Вместо неё рекомендуется использовать пакет функций [OpenEventHistorySession](#) (см. ниже).

Функция OpenEventHistorySession

```
GuidResult OpenEventHistorySession( Guid sessionId,  
EventHistoryQueryParams parameters )
```

Параметры:

| | |
|---|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>EventHistoryQueryParams parameters</code> | Параметры формирования\фильтрации событий. |

Результат: Возвращает GuidResult.

Описание: Открывает сессию отчета по событиям. Возвращает ключ для дальнейшего получения событий.

Функция CloseEventHistorySession

```
void CloseEventHistorySession( Guid sessionId, Guid  
eventHistorySessionID)
```

Параметры:

| | |
|---|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid eventHistorySessionID</code> | Уникальный ключ сессии отчета по событиям. |

Результат: –

Описание: Закрывает сессию отчета по событиям.

Функция GetEventHistoryResultCount

```
int GetEventHistoryResultCount( Guid sessionId, Guid  
eventHistorySessionID )
```

Параметры:

| | |
|---|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>Guid eventHistorySessionID</code> | Уникальный ключ сессии отчета по событиям. |

Результат: Возвращает int.

Описание: Возвращает количество событий в открытой сессии.

Функция GetEventHistoryResult

```
EventObject[] GetEventHistoryResult( Guid sessionId, Guid  
eventHistorySessionID, Guid[] fields, int offset, int count )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

| | |
|---|---|
| <code>Guid</code> <code>eventHistorySessionID</code> | Уникальный ключ сессии отчета по событиям. |
| <code>Guid[] fields</code> | Массив уникальных ключей полей, значения которых должны быть в результате. |
| <code>int offset</code> | Количество событий с начала, которое должно быть проигнорировано в этом результате. |
| <code>int count</code> | Количество событий, которое должно быть в результате. |

Результат: Возвращает массив [EventObject](#).

Описание: Возвращает массив событий ([EventObject](#)) со значениями затребованных полей. Список затребованных полей (`fields`) состоит из одного или нескольких ключей из приведенной ниже таблицы. В списке могут присутствовать ключи дополнительных полей.

| Guid | Описание |
|--------------------------------------|---|
| 71B03D7B-2E11-47CD-BF47-ADAF320AEB10 | Дата события |
| C7AD4F51-D8AF-4944-BF92-23714715147E | Время события |
| 2C5EE108-28E3-4DCC-8C95-7F3222D8E67F | Дата/время события |
| 633904B5-971B-4751-96A0-92DC03D5F616 | Источник события (наименование территории или оператора) |
| 42DAB9C6-5D30-4030-8CCD-2CAD6FCBC5F2 | Источник события (массив идентификаторов территорий) |
| D1847AFF-11AA-4EF2-AAAA-795CEEFE5F9F | Тип события (наименование) |
| 9F7A30E6-C9ED-4E62-83E3-59032A0F8D27 | Идентификатор события (Guid) |
| C4AE9465-8375-4169-BA61-EB7E365A7352 | Тип события (код в 16-ричной сист. счисления) |
| 57CA38E4-ED6F-4D12-ADCB-2FAA16F950D7 | Тип события (код в 10-чной сист. счисления) |
| 68EF9FD3-A72D-4520-9C63-5C37B0AE8539 | Субъект (ФИО) – из dictionary |
| 7C6D82A0-C8C8-495B-9728-357807193D23 | Идентификатор субъекта (PERS_ID – Guid) |
| 66C5B505-C3A7-4227-AAD3-6B7BA3D8E612 | Идентификатор субъекта (PERS_ID – Guid) в ParsecNET Office |
| 4C5807CB-2C06-4725-9243-747E40C41D6C | Область (название) |
| 2AB38696-1E30-4E04-A956-B951CB7C2033 | Деталь (название) |
| 89C9D5AC-6E13-4715-A524-7C3B34931385 | Рабочая станция |
| FEA92E1C-E07D-4932-A6A1-E8C53E3087D9 | Оператор |
| 03CEB65F-DCAD-4B56-94B8-BE9FDB463988 | Подробности |
| 763C4AD9-2D42-4A04-95B9-881E6B7FE7D3 | Температура человека |

| Guid | Описание |
|--------------------------------------|--------------------------------------|
| D396E37B-9CDF-48DC-93E0-A1FAB140C8D3 | Уровень алкоголя |
| E5AC823F-C4F6-48E7-BEBE-E6D44C57C7AD | Сводка |
| 66AA3A39-C866-4F34-9E99-E75F9918EAE7 | Комментарии оператора |
| 99914915-C882-4D11-80FF-57ACDC6CC015 | Заголовок |
| 2F4A647E-4D9E-48AD-BF11-B1E49FFEAC7F | Сообщение |
| 1BF8A893-7D21-4C0C-9A2D-2E333A2D769D | Полное имя субъекта |
| 0DE358E0-C91B-4333-B902-000000000003 | Фамилия / Номер авто |
| 0DE358E0-C91B-4333-B902-000000000001 | Имя / Модель авто |
| 0DE358E0-C91B-4333-B902-000000000002 | Отчество / Цвет авто |
| 0DE358E0-C91B-4333-B902-000000000006 | Табель |
| 0DE358E0-C91B-4333-B902-000000000009 | Организация |
| 0DE358E0-C91B-4333-B902-000000000004 | Подразделение |
| 0A679144-D5CE-476D-A56E-0A696F079B71 | Описание подразделения |
| 0DE358E0-C91B-4333-B902-00000000000A | Группа доступа |
| 0DE358E0-C91B-4333-B902-000000000005 | Код карты |
| 0DE358E0-C91B-4333-B902-000000000007 | Время действия с |
| 0DE358E0-C91B-4333-B902-000000000008 | Время действия по |
| 3AD06D24-43F6-45E0-8164-A98B4DA955DC | Фото |
| 6FCFA1BB-9624-4248-A2D5-AA84901C53C8 | Субъект в чёрном списке |
| 68D13785-C708-4418-8683-678A3F74957B | Изображения, прикрепленные к событию |

Функция GetHardwareEvents

`string[] GetHardwareEvents(Guid sessionId, int fromIndex)`

Параметры:

| | |
|-----------------------------|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>int fromIndex</code> | События с заданного значения индекса включительно. |

Результат: Возвращает набор событий.

Описание: Возвращает набор из заданного количества оперативных событий. Данные возвращаются в виде JSON-документа.

Пример объекта Event, полученного в результате выполнения функции

```
{
  "ID": "e299479e-596d-4be0-9688-950918621783",
  "Index": 1,
  "ParentID": "00000000-0000-0000-0000-000000000000",
  "Code": 590193,
```

```

"Component": "770ca0ee-4ecb-4648-bd3f-02b47140a670",
"Territories": [
  "ee7b6630-bcee-4137-b759-61c90889bf39"
],
>Date": "2015-06-10T11:43:30Z",
"Items": [
  {
    "Instance": 0,
    "Key": 83886592,
    "Type": 5,
    "Value": "0084C471"
  },
  {
    "Instance": 0,
    "Key": 16778496,
    "Type": 1,
    "Value": 83886081
  },
  {
    "Instance": 0,
    "Key": 67110912,
    "Type": 4,
    "Value": "eec6409a-c092-46b7-9d8f-d311fc45c66d"
  },
  {
    "Instance": 0,
    "Key": 67110144,
    "Type": 4,
    "Value": "770ca0ee-4ecb-4648-bd3f-02b47140a670"
  },
  {
    "Instance": 0,
    "Key": 50334464,
    "Type": 3,
    "Value": 65600
  },
  {
    "Instance": 0,
    "Key": 67109376,
    "Type": 4,
    "Value": "039b2bf8-5a66-4b22-be4d-fb3d42af14d5"
  }
]

```

```

    },
    {
        "Instance": 0,
        "Key": 201329152,
        "Type": 4,
        "Value": "a16486f7-8356-4fa9-8a4a-79e5659de052"
    }
]

```

Структура объекта Event

Таблица 1.

| Tag | Физический/ Логический тип | Описание |
|-------------|-------------------------------|---|
| ID | string/Guid | Идентификатор события |
| Index | int/int | Индекс события. Отсчитывается с момента логина оператора в интеграционный сервис. |
| ParentID | string/Guid | Идентификатор родительского события, которое выступало «инициирующим» |
| Code | int/int | Код типа события |
| Component | string/Guid | Идентификатор компонента-источника события |
| Territories | string[]/Guid[] | Массив идентификаторов территорий, ссылающихся на компонент-источник события |
| Date | string/DateTime | Дата и время события |
| Items | ItemData[] | Массив служебных данных типа ItemData (см. Таблица 2.) |

Структура объекта ItemData

Таблица 2.

| Tag | Физический/ логический тип | Описание |
|----------|-------------------------------|---|
| Instance | int/int | Индекс структуры данных в массиве данных события, сгруппированных по идентичности значений тэгов «Key» и «Type» |
| Key | int/int | Служебные данные. Содержат класс данных тэга «Value» в терминах системы ParsecNET. Класс данных вычленяется выполнением операции ((Key >> 8) & 0xffff). (см. Таблица 3.) |
| Type | int/int | Логический тип данных тэга «Value» в терминах системы ParsecNET (см. Таблица 4.) |

Возможные значения класса данных в тэге «Key» структуры ItemData

Таблица 3.

| Значение | Описание |
|----------|-------------------------|
| 0 | Неопределенное значение |
| 2 | Пользователь |
| 4 | Команда |
| 5 | Компонент |
| 6 | Состояние |
| 7 | Деталь |

| | |
|----|---------------------------------------|
| 8 | Устройство |
| 9 | Получатель быстрого транспорта |
| 10 | Отправитель быстрого транспорта |
| 11 | Маска событий |
| 12 | Рабочая станция |
| 13 | Оператор |
| 14 | Объект системы ParsecNET |
| 15 | Область видимости |
| 16 | Задание автоматизации |
| 17 | Интерпретатор события |
| 18 | Сервер |
| 19 | Дата/Время |
| 20 | Изображение |
| 21 | Автомобильный номер |
| 22 | Область изображения |
| 23 | Комментарии оператора |
| 24 | Канал |
| 25 | Звук |
| 26 | Связанное событие |
| 27 | Текстовое сообщение |
| 28 | Адрес получателя текстового сообщения |

Возможные значения тэга «Type» структуры ItemData

Таблица 4.

| Значение | Описание |
|----------|--|
| 0 | Неопределенное значение |
| 1 | DWORD. 4-байтовое целое число |
| 2 | double. 8-байтовое число с плавающей точкой |
| 3 | long. 8-байтовое целое число |
| 4 | guid. 16-байтовый глобальный уникальный идентификатор |
| 5 | char[]. Строковое значение с максимальной длиной в 16 символов |
| 6 | datetime. Дата-Время |
| 7 | byte[]. Массив данных с максимальной длиной массива 16 байт |
| 8 | guid. Ссылка на бинарные данные большого размера |
| 9 | struct { int, int, int, int }. Структура-описатель прямоугольной области |

Функция GetHardwareEventsResolved

```
string[] GetHardwareEventsResolved( Guid sessionId, int fromIndex )
```

Параметры:

| | |
|-----------------------------|--|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>int fromIndex</code> | События с заданного значения индекса включительно. |

Результат: Возвращает набор событий.

Описание: Возвращает набор из заданного количества оперативных событий. Данные представляются в удобном для пользователя виде.

Пример объекта EventResolved, полученного в результате выполнения функции (для того же события, что и в примере к предыдущей функции):


```

{
  "ID": "e299479e-596d-4be0-9688-950918621783",
  "Index": 1,
  "ParentID": "00000000-0000-0000-0000-000000000000",
  "Code": {
    "Raw": 590193,
    "Resolved": "Нет входа - нет карты в БД"
  },
  "Classes": [
    {
      "Raw": 64,
      "Resolved": "Отказ в доступе"
    },
    {
      "Raw": 65536,
      "Resolved": "Отказ в доступе на вход"
    }
  ],
  "Component": {
    "Raw": "770ca0ee-4ecb-4648-bd3f-02b47140a670",
    "Resolved": "Шлагбаум въезд всех и выезд посетителей (NC 5K)"
  },
  "Territories": [
    {
      "Raw": "ee7b6630-bcee-4137-b759-61c90889bf39",
      "Resolved": "Шлагбаум въезд всех и выезд посетителей (NC 5K)"
    }
  ],
  "Date": "2015-06-10T11:43:30Z",
  "CardCode": "0084C471",
  "User": {
    "Raw": "039b2bf8-5a66-4b22-be4d-fb3d42af14d5",
    "Resolved": "Иванов Иван Иванович"
  },
  "Operator": null,
  "Workstation": null,
  "Part": null,
  "Details": null
}

```

Структура объекта EventResolved

| Tag | Физический/ логический тип | Описание |
|-----|-------------------------------|----------|
|-----|-------------------------------|----------|

| | | |
|-------------|-----------------|--|
| ID | string/Guid | Идентификатор события |
| Index | int/int | Индекс события. Отсчитывается с момента логина оператора в интеграционный сервис. |
| ParentID | string/Guid | Идентификатор родительского события, которое выступало «инициирующим» |
| Code | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий тип события |
| Classes | ResolvedValue[] | Массив объектов типа ResolvedValue, описывающих категории, к которым относится событие |
| Component | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий компонент-источник события |
| Territories | ResolvedValue[] | Массив объектов типа ResolvedValue, описывающих территории, которые ссылаются на компонент-источник события |
| Date | string/DateTime | Дата и время события |
| CardCode | string/string | Код карты пользователя системы ParsecNet (если применимо к конкретному типу события) |
| User | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий пользователя системы ParsecNet (если применимо к конкретному типу события) |
| Operator | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий оператора системы ParsecNet (если применимо к конкретному типу события) |
| Workstation | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий рабочую станцию системы ParsecNet (если применимо к конкретному типу события) |
| Part | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий деталь системы ParsecNet (если применимо к конкретному типу события) |
| Details | ResolvedValue | Экземпляр объекта типа ResolvedValue, описывающий данные объекта системы ParsecNet, связанного с конкретным событием (если применимо к конкретному типу события) |

Структура объекта ResolvedValue

| Tag | Физический/ логический тип | Описание |
|----------|-------------------------------|---|
| Raw | Any/Any | Данные, подходящие для компьютерной обработки |
| Resolved | string/string | Данные, подходящие для прочтения человеком |

Функция EventsSubscribe

`GuidResult EventsSubscribe(Guid sessionId, EventFilter eventFilter, int eventFormat, int httpMethod, string urlDestination)`

Параметры:

`Guid sessionId`

Уникальный ключ сессии.

`EventFilter eventFilter`

Параметры фильтрации событий.
В качестве данного параметра можно использовать `AdvancedEventFilter`.

| | |
|------------------------------------|---|
| <code>int eventFormat</code> | Выбор формата предоставления данных события: 0 – JSON (см. <code>GetHardwareEvents</code>); 1 – Resolved JSON (см. <code>GetHardwareEventsResolved</code>). |
| <code>int httpMethod</code> | Метод получения данных: 0 – GET; 1 – POST. |
| <code>string urlDestination</code> | URL, который будет вызван при возникновении событий, удовлетворяющих заданным фильтрам. |

Результат: Возвращает `GuidResult`.

Описание: Функция позволяет подписаться на определенные события, указав условия фильтрации, которым они должны соответствовать, и URL, который будет вызываться по заданным параметрам. Возвращает ключ созданной подписки.

Для получения данных по методу GET (`httpMethod=0`) параметр `urlDestination` должен содержать набор ключей полей события, значения которых должны быть в вызываемом URL. Список ключей полей события описаны в функции [GetEventHistoryResult](#).

Например (цветом выделены ключи полей события):
`http://<url>:<port>/?eventDate=2c5ee108-28e3-4dcc-8c95-7f3222d8e67f`
`&code=57ca38e4-ed6f-4d12-adcb-2faa16f950d7`
`&component=633904b5-971b-4751-96a0-92dc03d5f616`
`&user=68ef9fd3-a72d-4520-9c63-5c37b0ae8539`

Подписки будут продолжать функционировать даже после автоматического завершения сессии (по умолчанию через 5 минут). Если закрыть сессию принудительно, вызвав функцию [CloseSession](#), то все подписки, созданные в этой сессии, будут удалены.

Функция `EventsSubscribeCheck`

`BaseResult EventsSubscribeCheck(Guid sessionID, Guid subscriptionID)`

Параметры:

| | |
|----------------------------------|---------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid subscriptionID</code> | Уникальный ключ подписки. |

Результат: Возвращает `BaseResult`.

Описание: Проверяет наличие подписки с указанным ключом.

Функция `EventsUnsubscribe`

`BaseResult EventsUnsubscribe(Guid sessionID, Guid subscriptionID)`

Параметры:

| | |
|----------------------------------|---------------------------|
| <code>Guid sessionID</code> | Уникальный ключ сессии. |
| <code>Guid subscriptionID</code> | Уникальный ключ подписки. |

Результат: Возвращает `BaseResult`.

Описание: Функция удаляет подписку с указанным ключом.

Функция EventsUnsubscribeUrl

```
BaseResult EventsUnsubscribeUrl( Guid sessionId, string urlDestination)
```

Параметры:

| | |
|------------------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>string urlDestination</code> | URL подписки. |

Результат: Возвращает BaseResult.

Описание: Функция удаляет все подписки, начинающиеся с указанного URL.

Функция GetTransactionClasses

```
TransactionClass[] GetTransactionClasses( Guid sessionId )
```

Параметры:

| | |
|-----------------------------|-------------------------|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
|-----------------------------|-------------------------|

Результат: Возвращает набор категорий событий.

Описание: Возвращает набор всех категорий возникающих в системе событий. Событие в системе ParsecNET может принадлежать к одной и более категории одновременно. Каждая категория в возвращаемом наборе соответствует 1 биту битовой маски, которая назначается событиям.

Функция GetTransactionTypes

```
TransactionType[] GetTransactionTypes( Guid sessionId, long classMask )
```

Параметры:

| | |
|-----------------------------|---|
| <code>Guid sessionId</code> | Уникальный ключ сессии. |
| <code>long classMask</code> | Битовая маска, задающая категории событий. Если определенный бит установлен в 1, то будут возвращены события, относящиеся к данной категории. |

Результат: Возвращает набор типов событий.

Описание: Возвращает набор событий с помощью битовой маски категорий.

3. ИСТОРИЯ ИЗМЕНЕНИЙ

Версия 3.13.113 (январь 2023)

Добавлены объекты: QRAdvancedGroup, QRAdvancedData.

Добавлены функции: EventsSubscribeCheck, GetAdvancedQRGroups, GenerateAdvancedParsecQRCode.

Формат данных, передаваемых при вызове функции подписки по методу POST, изменился на Content-Type: application/json.

Версия 3.12.323.44 (август 2022)

Изменилось поведение функций GetTerritoriesHierarchy, GetTerritorySubItems, GetPersonExtraFieldValues, GetHolidays, GetDomains, GetEventHistoryResult, GetHardwareState, GetHardwareEvents, GetHardwareEventsResolved, FindPeople, FindVisitors, FindVehicle, FindInBlackList – теперь при возникновении ошибки возвращается null, а не пустой массив объектов.

Версия 3.12.323.38 (июль 2022)

Добавлены объекты: EventFilter и AdvancedEventFilter.

Добавлены функции: EventsSubscribe, EventsUnsubscribe и EventsUnsubscribeUrl.

Версия 3.12.323.17 (апрель 2022)

Добавлена возможность работать с группой доступа типа «Автомобильный номер».

Добавлены функции: CreateVehicleAccessGroup, CreateVehicleTempAccessGroup, GetBlackList, FindInBlackList, AddToBlackList, RemoveFromBlackList.

Добавился новый параметр UseLocalTime в объекте EventHistoryQueryParams.

Версия 3.11.629.15 (июль 2021)

Добавлены функции: GetUnique4bCardCode, GetCardCodeFromUID, GenerateParsecQRCode.

Версия 3.11.127.25 (март 2021)

Добавлены функции: GetObjectName, SendVerificationCommand, GetMultiplePersons, PersonSearch

Дополнены роли, используемые в функции CheckRole.

Версия 3.10.1111.12 (декабрь 2020)

Добавлены объекты PassageRole и IdentifierExData.

Добавлены функции: GetPassageRoles, CreatePassageRole, SavePassageRole, DeletePassageRole, GetIdentifierExtraData, SetIdentifierExtraData, GetPersonWorktimeSchedule, SetPersonWorktimeSchedule.

Версия 3.10.1111.6 (ноябрь 2020)

Расширена функция CreateOrgUnit.

В объект Identifier добавлено новое поле NAME.

Версия 3.10.325.23 (август 2020)

Добавлены описания для функций GetTransactionClasses, GetTransactionTypes и описания классов TransactionClass и TransactionType.

Расписан нюанс с необходимостью указания в запросах атрибута xsi: Type в соответствующих методах.

Версия 3.8 (октябрь 2018)

Добавлена функция GetPersonsChangedAfter

Исправления в целях упрощения понимания.

Версии, начиная с 3.7.1107.65 (август 2018)

Функция GetEventHistoryResult

Добавлена строка

| | |
|--------------------------------------|--|
| 42DAB9C6-5D30-4030-8CCD-2CAD6FCBC5F2 | Источник события (массив идентификаторов территорий) |
|--------------------------------------|--|

Версия 3.7.1107 (ноябрь 2017)

Добавлена строка «Элемент PRIVILEGE_MASK игнорируется» в описание функций:

AddPersonIdentifier

ChangePersonIdentifier

Строка «Если поле ID в структуре person равен null, то ID генерируется автоматически и возвращается.» заменена строкой «Если поле ID в структуре person равен Guid.Empty (00000000-0000-0000-0000-000000000000), то ID генерируется автоматически и возвращается.» в описании функций:

CreatePerson

CreateVisitor

Версия 3.7.407 (март 2017)

Добавлены функции работы с вложенными группами доступа через веб-сервис:

GetInheritedAccessGroups

SetInheritedAccessGroups

Реализована возможность иметь несколько идентификаторов с одинаковым кодом у субъекта доступа.

Версия 3.6.1018.18 (от 21/11/2016)

Добавлена функция GetVisitorExtraFieldTemplates, которая возвращает шаблоны дополнительных полей Посетителей.

Внесены изменения в связи с добавлением отдельного права "Удаление персонала".

Теперь посредством интеграционного сервиса можно получить прикрепленное к событию изображение с камеры.

Исправлена ошибка в интеграционном сервисе и в Web-консоли заявок Бюро пропусков: нельзя было изменить заявку со статусом "Согласована".

Менеджер сессий интеграционного сервиса оптимизирован.

Версия 3.5.818

Расширен функционал API ParsecNET 3. Теперь разработчикам доступно создание произвольных расписаний и групп доступа! Также расширены возможности получения списка заявок бюро пропусков с фильтрацией на стороне сервера. Полная свобода действий и неограниченные возможности интеграции!

Удалена привилегия canBlocked.

Добавлены аудитные события на открытие и закрытие сессии интеграционного сервиса.

Добавлена новая функция GetVisitorRequests

Добавлены новые функции для работы с расписаниями:

CreateAccessSchedule

CreateWorktimeSchedule

GetSchedule

SaveSchedule

DeleteSchedule

GetScheduleDetails

SetScheduleDays

SetScheduleFix

DeleteScheduleDays

GetHolidays

SetHolidays

DeleteHolidays

Добавлена новая функция для получения списка подгрупп компонент группы доступа GetSubAccessGroups , а также новый класс SubAccessGroup .

Версия 3.4.755 (от 17/07/2015)

Добавлены функции для работы с группами доступа:

CreateAccessGroup

AddSubAccessGroup

DeleteSubAccessGroup

DeleteAccessGroup

Добавлена возможность получения статусов оборудования и отправки команд прямого управления посредством интеграционного сервиса:

GetHardwareState

SendHardwareCommand

Добавлены оперативные события оборудования в интеграционном сервисе

GetHardwareEvents

GetHardwareEventsResolved

Добавлен новый метод: SetIdentifierPrivileges

Версия 3.3.742 (от 02/03/2015)

Добавлена функция CheckRole для проверки ролей оператора.

Добавлено поле EVENT_ID, чтобы посредством интеграционного сервиса можно было получать связанные транзакции.

Исправлен список ключей дополнительных полей событий для GetEventHistoryResult, необходимых для работы со связанными запросами.

Исправлена ошибка в функции CreateTempAccessGroup.

Исправлены ошибки в функциях GetRootTerritory и GetRootOrgUnit.

Исправлены ошибки в функции удаления заявки
DeleteIssuedVisitorRequest.

Версия 3.3.702 (от 18/06/2014)

Для совместимости интеграционного сервиса (SDK) с Delphi в функцию OpenEventHistorySession внесено изменение, интерпретирующее EventHistoryQueryParams.ParentEventId=Guid.Empty как аналог значения null.

Версия 3.3.682 (от 02/12/2013)

Исправлена ошибка переподключения интеграционного сервиса к серверу приложений.

Устранена ошибка, приводящая к зависанию консоли и интеграционного сервиса при большой нагрузке на систему.

В интеграционный сервис добавлено 3 новых функции для работы с заявками бюро пропусков:

CloseAllActiveVisitorRequests

CloseVisitorRequest

GetPersonVisitorRequests

Версия 3.3.673 (от 12/09/2013)

Исправлена ошибка в интеграционном сервисе в функции GetPersonIdentifiers (в клиентских приложениях, работающих через интеграционный сервис и использующих данную функцию, необходима регенерация проху-классов).

Версия 3.3.670 (от 19/08/2013)

Устранена проблема с падением интеграционного сервиса при работе с web-консолью заявок бюро пропусков под нагрузкой.

Исправлена ошибка с падением службы "ParsecNET 3 Server" при интенсивном создании сессий EventReport'ов через интеграционный сервис.

Версия 3.3.659 (от 01/06/2013)

Добавлены две новые функции для предоставления/отмены привилегий "Запрет выхода" (NC-100K-IP, NC-8000, NC-8000-D), "Запрет входа" (NC-8000, NC-8000-D):

BlockPerson

UnblockPerson

Версия 3.3.644 (от 25/02/2013)

Добавлены новые функции для работы с посетителями:

CreateVisitor

FindVisitors

GetOrgUnitSubItemsHierarchyWithVisitors

GetOrgUnitsHierarchyWithVisitors

4. АЛФАВИТНЫЙ УКАЗАТЕЛЬ

| | | | |
|-------------------------------------|----|----------------------------------|----|
| AccessGroup | 24 | DeleteHolidays | 59 |
| AccessSchedule | 18 | DeleteIdentifier | 50 |
| ActivateVisitorRequest | 63 | DeleteIssuedVisitorRequest | 64 |
| AddPersonIdentifier | 50 | DeleteOrgUnit | 37 |
| AddPersonScheduleFix | 42 | DeletePassgeRole | 54 |
| AddSubAccessGroup | 61 | DeletePerson | 48 |
| AddToBlackList | 44 | DeletePersonScheduleFix | 43 |
| AdvancedEventFilter | 27 | DeleteSchedule | 58 |
| BaseIdentifier | 20 | DeleteScheduleDays | 59 |
| BaseObject | 15 | DeleteSubAccessGroup | 62 |
| BaseOrgUnit | 15 | Domain | 25 |
| BasePerson | 16 | Event | 24 |
| BaseResult | 14 | EventFilter | 27 |
| BaseTerritory | 22 | EventHistoryQueryParams | 25 |
| BlockPerson | 48 | EventObject | 25 |
| ChangePersonIdentifier | 51 | EventsHistory | 24 |
| CheckRole | 30 | EventsHistoryResult | 15 |
| CloseAllActiveVisitorRequests | 65 | EventsSubscribe | 74 |
| CloseEventHistorySession | 67 | EventsSubscribeCheck | 75 |
| CloseOrgUnitEditingSession | 36 | EventsUnsubscribe | 75 |
| ClosePersonEditingSession | 46 | EventsUnsubscribeUrl | 76 |
| CloseSession | 30 | ExtraFieldValue | 17 |
| CloseVisitorRequest | 65 | FindInBlackList | 44 |
| ContinueSession | 30 | FindPeople | 38 |
| CreateAccessGroup | 60 | FindPersonByIdentifier | 39 |
| CreateAccessSchedule | 56 | FindVehicle | 38 |
| CreateOrgUnit | 36 | FindVisitorRequest | 63 |
| CreatePassageRole | 53 | FindVisitors | 38 |
| CreatePerson | 45 | GenerateAdvancedParsecQRCode ... | 55 |
| CreateTempAccessGroup | 60 | GenerateParsecQRCode | 55 |
| CreateVehicle | 45 | GetAcceptedVisitorRequests | 63 |
| CreateVehicleAccessGroup | 61 | GetAccessGroups | 60 |
| CreateVehicleTempAccessGroup | 60 | GetAccessSchedules | 55 |
| CreateVisitor | 45 | GetAdvancedQRGroups | 55 |
| CreateVisitorRequest | 64 | GetBlackList | 44 |
| CreateWorktimeSchedule | 57 | GetCardCodeFromUID | 54 |
| DeleteAccessGroup | 61 | GetDomains | 29 |

| | | | |
|--|----|-------------------------------------|----|
| GetEventHistoryResult | 67 | GetSubAccessGroups | 62 |
| GetEventHistoryResultCount | 67 | GetTerritoriesHierarhy | 49 |
| GetEvents | 66 | GetTerritory | 49 |
| GetHardwareEvents | 69 | GetTerritorySubItems | 49 |
| GetHardwareEventsResolved | 72 | GetTransactionClasses | 76 |
| GetHardwareState | 32 | GetTransactionTypes | 76 |
| GetHolidays | 59 | GetUnique4bCardCode | 54 |
| GetIdentifierExtraData | 53 | GetVehicleExtraFieldTemplates | 38 |
| GetInheritedAccessGroups | 62 | GetVersion | 29 |
| GetIssuedVisitorRequests | 65 | GetVisitorExtraFieldTemplates | 37 |
| GetMultiplePersons | 41 | GetVisitorRequest | 64 |
| GetObjectName | 31 | GetVisitorRequests | 65 |
| GetOrgUnit | 36 | GetWorktimeSchedules | 56 |
| GetOrgUnitsHierarhy | 34 | GuidResult | 14 |
| GetOrgUnitsHierarhyWithPersons | 34 | HardwareState | 27 |
| GetOrgUnitsHierarhyWithVehicle | 34 | Holiday | 20 |
| GetOrgUnitsHierarhyWithVisitors | 34 | Identifier | 21 |
| GetOrgUnitSubItems | 35 | IdentifierExData | 22 |
| GetOrgUnitSubItemsHierarhyWithPersons | 35 | IdentifierTemp | 21 |
| GetOrgUnitSubItemsHierarhyWithVehicle | 35 | ObjectResult | 15 |
| GetOrgUnitSubItemsHierarhyWithVisitors | 35 | OpenEventHistorySession | 67 |
| GetPassageRoles | 53 | OpenOrgUnitEditingSession | 36 |
| GetPerson | 40 | OpenPersonEditingSession | 45 |
| GetPersonExtraFieldTemplates | 37 | OpenSession | 29 |
| GetPersonExtraFieldValue | 41 | OpenSessionWithInLocale | 29 |
| GetPersonExtraFieldValues | 41 | OrgUnit | 16 |
| GetPersonExtraFieldValueString | 42 | PassageRole | 22 |
| GetPersonIdentifiers | 50 | Person | 16 |
| GetPersonsChangedAfter | 41 | PersonExtraFieldTemplate | 17 |
| GetPersonScheduleFixes | 42 | PersonScheduleFix | 17 |
| GetPersonVisitorRequests | 66 | PersonSearch | 39 |
| GetPersonWorktimeSchedule | 43 | PersonWithPhoto | 16 |
| GetRootOrgUnit | 34 | QRAdvancedData | 28 |
| GetRootTerritory | 49 | QRAdvancedGroup | 28 |
| GetSchedule | 57 | RemoveFromBlackList | 44 |
| GetScheduleDetails | 58 | SaveOrgUnit | 37 |
| GetScheduleIntervals | 56 | SavePassageRole | 54 |
| | | SavePerson | 46 |
| | | SavePersonScheduleFix | 43 |

| | | | |
|--------------------------------|----|---------------------------------|----|
| SaveSchedule..... | 57 | SetPersonWorktimeSchedule | 43 |
| SaveVisitorRequest | 64 | SetScheduleDays | 58 |
| Schedule | 18 | SetScheduleFix | 58 |
| ScheduleDay | 19 | StockIdentifier | 21 |
| ScheduleFix | 19 | StringResult | 14 |
| SendHardwareCommand | 31 | SubAccessGroup | 24 |
| SendVerificationCommand | 32 | Territory | 23 |
| Session | 15 | TerritoryWithComponent | 23 |
| SessionResult | 14 | TimeInterval..... | 20 |
| SetHolidays..... | 59 | TransactionClass | 27 |
| SetIdentifierExtraData | 53 | TransactionType..... | 28 |
| SetIdentifierPrivileges | 52 | UnblockPerson | 48 |
| SetInheritedAccessGroups | 62 | ValidateExtraFieldValue | 42 |
| SetPersonExtraFieldValue | 47 | VisitorRequest | 18 |
| SetPersonExtraFieldValues..... | 48 | WorktimeInterval | 20 |
| SetPersonOrgUnit..... | 47 | WorktimeSchedule..... | 19 |
| SetPersonPhoto | 46 | | |