

# **SOAP сервисы APS**

## **USER MANUAL**

**© 2017 APS Solutions**  
**APS Solutions**

<b>1. Описание</b>	<b>6</b>
<b>2. Список методов</b>	<b>8</b>
<b>2.1 Справочные методы</b>	<b>9</b>
2.1.1 GetVersion	9
2.1.2 BudgetItemFind	9
2.1.3 CompanyFind	9
2.1.4 DepFind	10
2.1.5 UnitsFind	11
2.1.6 GetServersStats	11
2.1.7 GetBudgetItems	11
2.1.8 GetCompanyDeps	12
2.1.9 NMCDetailFind	12
2.1.10 CheckAuth	13
2.1.11 UserIdFind	13
2.1.12 GetCompanies	14
2.1.13 GetNMCDetails	14
<b>2.2 Категории и номенклатуры</b>	<b>15</b>
2.2.1 IndustryFind	15
2.2.2 NMCAppend	15
2.2.3 NMCUpdate	16
2.2.4 NMCSetRest	17
2.2.5 NMCDelete	18
2.2.6 IndustryGetCount	19
2.2.7 IndustryGetByN	19
2.2.8 NMCGetCount	19
2.2.9 NMCGetByN	20
2.2.10 NMCFind	20
2.2.11 GetIndustries	21
2.2.12 NMCGetNewCount	21
2.2.13 NMCGetNext	22
2.2.14 NMCSetRests	22
2.2.15 NMCArrayAppend	23
<b>2.3 Заявки</b>	<b>24</b>
2.3.1 RQAppend	24
2.3.2 RQItemAppend	25
2.3.3 RQAppendWithItems	26
2.3.4 RQGetNewCount	27
2.3.5 RQGetNext	28
2.3.6 RQItemGetByN	28
2.3.7 RQGetByN	29
2.3.8 RQGetReturnedCount	29
2.3.9 RQReturnedGetNext	29

2.3.10	ClosedRQItemGetByN .....	30
<b>2.4</b>	<b>Договоры и контрагенты .....</b>	<b>30</b>
2.4.1	SupplierGet .....	30
2.4.2	SupplierGetOtherUsers .....	31
2.4.3	ContractsGetNewCount .....	31
2.4.4	ContractsGetNext .....	32
2.4.5	ContractAppend .....	32
2.4.6	ContractApprove .....	34
2.4.7	ContractsGetByContragent .....	34
2.4.8	ContractsGetByN .....	35
2.4.9	SuppliersGetNewCount .....	35
2.4.10	SuppliersGetNext .....	36
2.4.11	SuppliersGetByN .....	37
<b>2.5</b>	<b>Заказы .....</b>	<b>37</b>
2.5.1	OrdersGetNewCount .....	37
2.5.2	OrdersGetNext .....	38
2.5.3	OrderItemGetByN .....	38
2.5.4	OrderItemSetAccepted .....	39
2.5.5	OrdersGetNextWithItems .....	40
2.5.6	OrderAppend .....	40
2.5.7	OrderAppendWithItems .....	41
2.5.8	OrderClose .....	43
2.5.9	OrdersGetByN .....	43
2.5.10	OrdersGetByNWithItems .....	44
<b>2.6</b>	<b>Платежи .....</b>	<b>44</b>
2.6.1	PaymentInsert .....	44
<b>2.7</b>	<b>Компании .....</b>	<b>45</b>
2.7.1	CompanyAppend .....	45
2.7.2	CompanyUpdate .....	46
<b>2.8</b>	<b>Служебные методы .....</b>	<b>47</b>
2.8.1	EntitySetSynced .....	47
<b>3.</b>	<b>Типы данных .....</b>	<b>49</b>
<b>3.1</b>	<b>AuthStruct .....</b>	<b>50</b>
<b>3.2</b>	<b>IDStruct .....</b>	<b>50</b>
<b>3.3</b>	<b>NMCStruct .....</b>	<b>50</b>
<b>3.4</b>	<b>NMCRestStruct .....</b>	<b>50</b>
<b>3.5</b>	<b>RQStruct .....</b>	<b>50</b>
<b>3.6</b>	<b>RQStructWithItems .....</b>	<b>51</b>
<b>3.7</b>	<b>RQItemStructArray .....</b>	<b>51</b>
<b>3.8</b>	<b>RQItemStruct .....</b>	<b>51</b>
<b>3.9</b>	<b>SupplierDataStruct .....</b>	<b>52</b>

3.10	SupplierData .....	52
3.11	SupplierUserStruct .....	52
3.12	SupplierUser .....	52
3.13	ContractInfoStruct .....	53
3.14	ContractInfo .....	53
3.15	OrderInfoStruct .....	53
3.16	OrderInfo .....	54
3.17	OrderItemStruct .....	54
3.18	Industry .....	54
3.19	RQInfoStruct .....	54
3.20	NMCInfoStruct .....	55
3.21	IndustryInfoStruct .....	55
3.22	RequestItemStruct .....	55
3.23	CodeNamePairArrayStruct .....	55
3.24	CodeNamePairArray .....	55
3.25	CodeNamePair .....	56
3.26	OrderItemDataStruct .....	56
3.27	OrderInfoStructWithItems .....	56
3.28	OrderInfoWithItems .....	56
3.29	PaymentInfo .....	57
3.30	NMCRestStructArray .....	57
3.31	NMCStructArray .....	57
3.32	ContractInfoArray .....	57
3.33	ContractInfoArrayStruct .....	57
3.34	CompanyData .....	57
3.35	CompanyDataStruct .....	58
3.36	ClosedRQItemStruct .....	58
3.37	ClosedRequestItemStruct .....	58
<b>4.</b>	<b>Статусы объектов</b>	<b>59</b>
4.1	Заявки .....	60
4.2	Договоры .....	60
4.3	Заказы .....	60
4.4	Закрытые позиции заявок .....	60
<b>5.</b>	<b>Коды ошибок</b>	<b>61</b>
<b>6.</b>	<b>Правила автораспределения заявки</b>	<b>63</b>

<b>Index</b>	<b>0</b>
--------------	----------

## Описание

## 1 Описание

SOAP сервисы APS предназначены для интеграции системы APS Tender с внешними системами клиента.

# Список методов



## 2 Список методов

### 2.1 Справочные методы

#### 2.1.1 GetVersion

**Возвращает текущую версию модуля.**

Входные параметры:

- нет

Возвращаемый тип: *string*

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetVersion soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
  </soapenv:Body>
</soapenv:Envelope>
```

#### 2.1.2 BudgetItemFind

**Поиск кода статьи бюджета**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- budgetItemName – название статьи бюджета (тип данных *string*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

**Особенности:**

Поиск регистронезависим. При успешном поиске в поле ErrorCode результата содержится 0, в поле Result\_id - код найденной статьи бюджета.

**Возможные коды ошибок:**

- -1 - статья бюджета не найдена.
- 22331 - недопустимое название статьи бюджета.

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="
  <soapenv:Header/>
  <soapenv:Body>
    <urn:BudgetItemFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <budgetItemName xsi:type="xsd:string">quae divum incedo</budgetItemName>
    </urn:BudgetItemFind>
  </soapenv:Body>
</soapenv:Envelope>
```

#### 2.1.3 CompanyFind

**Поиск кода компании/филиала**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

- `companyName` – название компании/филиала (тип данных *string*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Поиск регистронезависим. При успешном поиске в поле `ErrorCode` результата содержится 0, в поле `Result_id` – код найденного филиала/подразделения

#### Возможные коды ошибок:

- -1 – компания/филиал не найден(а).
- 22241 – недопустимое название компании/филиала.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:CompanyFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">username</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <companyName xsi:type="xsd:string">432</companyName>
    </urn:CompanyFind>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.1.4 DepFind

### Поиск кода подразделения

Входные параметры:

- `auth` – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- `depName` – название подразделения (тип данных *string*).
- `companyId` – код компании (тип данных *long*) – необязательный параметр.

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Поиск регистронезависим. При успешном поиске в поле `ErrorCode` результата содержится 0, в поле `Result_id` – код найденного подразделения. При положительном значении параметра `companyId` поиск происходит только по подразделениям указанной компании.

#### Возможные коды ошибок:

- -1 – подразделение не найдено.
- 22441 – недопустимое название подразделения.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:DepFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
```

```

        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
    </auth>
    <depName xsi:type="xsd:string">quae divum incedo</depName>
    <companyId xsi:type="xsd:long">10</companyId>
</urn:DepFind>
</soapenv:Body>
</soapenv:Envelope>

```

### 2.1.5 UnitsFind

#### Поиск кода единицы измерения

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- unitName – название единицы измерения (тип данных *string*).
- insertNew – если значение 1, то при отсутствии искомого значения оно добавляется (тип данных *short*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Поиск регистрозависим(!). При успешном поиске в поле ErrorCode результата содержится 0, в поле Result\_id – код найденной единицы измерения

#### Возможные коды ошибок:

- -1 – единица измерения не найдена.
- 22231 – недопустимое название единицы измерения.

#### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:UnitsFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <unitName xsi:type="xsd:string">quae divum incedo</unitName>
      <insertNew xsi:type="xsd:short">1</insertNew>
    </urn:UnitsFind>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.1.6 GetServersStats

#### Получение диагностической информации

#### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetServersStats soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.1.7 GetBudgetItems

#### Получение списка доступных статей бюджета

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [CodeNamePairArrayStruct](#)<sup>[55]</sup>, в поле Code каждого элемента - ID статьи бюджета, в поле Name - название.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetBudgetItems soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </Auth>
    </urn:GetBudgetItems>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.1.8 GetCompanyDeps

#### Получение списка подразделений компании

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- companyId - код компании (тип данных *long*)

Возвращается структура типа [CodeNamePairArrayStruct](#)<sup>[55]</sup>, в поле Code каждого элемента - ID подразделения, в поле Name - название.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetCompanyDeps soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </Auth>
      <CompanyId xsi:type="xsd:long">4</CompanyId>
    </urn:GetCompanyDeps>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.1.9 NMCDetailFind

#### Поиск кода места поставки

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- nmcDetailName – название места поставки (тип данных *string*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Поиск регистронезависим. При успешном поиске в поле ErrorCode результата содержится 0, в поле Result\_id - код найденного места поставки

#### Возможные коды ошибок:

- -1 - место поставки не найдено.
- 22251 - недопустимое название места поставки.

**Пример:**

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCDetailFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <nmcDetailName xsi:type="xsd:string">Склад 3</nmcDetailName>
    </urn:NMCDetailFind>
  </soapenv:Body>
</soapenv:Envelope>

```

**2.1.10 CheckAuth****Проверка авторизации**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>**Особенности:**

При успешной авторизации в поле ErrorCode результата содержится 0, в поле Result\_id - код пользователя

**Возможные коды ошибок:**

- -11 - недопустимый пользователь
- -2 - пустой пароль
- -7 - дублирование пользователя
- -4 - неактивный пользователь
- -5 - неверный пароль

**Пример:**

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:CheckAuth soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:CheckAuth>
  </soapenv:Body>
</soapenv:Envelope>

```

**2.1.11 UserIdFind****Выдача кода пользователя**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- userLogin – логин пользователя (тип данных *string*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>**Особенности:**

Поиск регистронезависим. При успешном поиске в поле ErrorCode результата содержится 0, в поле Result\_id - код найденного пользователя

#### Возможные коды ошибок:

- -1 - пользователь не найден.
- 25331 - недопустимый логин.

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:UserIdFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <userLogin xsi:type="xsd:string">user2</userLogin>
    </urn:UserIdFind>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.1.12 GetCompanies

#### Выдача списка компаний/филиалов

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>501</sup>)

Возвращается структура типа [CodeNamePairArrayStruct](#)<sup>551</sup>

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetCompanies soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </Auth>
    </urn:GetCompanies>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.1.13 GetNMCDetails

#### Выдача списка складов

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>501</sup>)

Возвращается структура типа [CodeNamePairArrayStruct](#)<sup>551</sup>

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetNMCDetails soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </Auth>
    </urn:GetNMCDetails>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        </urn:GetNMCDetails>
    </soapenv:Body>
</soapenv:Envelope>

```

## 2.2 Категории и номенклатуры

Enter topic text here.

### 2.2.1 IndustryFind

#### Поиск кода категории

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- industryName – название категории (тип данных *string*).
- insertNew - если значение 1, то при отсутствии искомого значения оно добавляется (тип данных *short*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Поиск регистронезависим. При успешном поиске в поле ErrorCode результата содержится 0, в поле Result\_id - код найденной категории

#### Возможные коды ошибок:

- -1 - категория не найдена.
- 22221 - недопустимое название категории.

#### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:IndustryFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <industryName xsi:type="xsd:string">quae divum incedo</industryName>
      <insertNew xsi:type="xsd:short">1</insertNew>
    </urn:IndustryFind>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.2.2 NMCAppend

#### Добавление номенклатуры в базу

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- NMC – структура номенклатуры (тип данных [NMCStruct](#)<sup>[50]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном добавлении в поле ErrorCode результата содержится 0 или 1013, в поле Result\_id - код добавленной номенклатуры. При необходимости добавить группу номенклатур верхнего уровня тэг PARENT\_ID пропускается, или в него вносится 0 или отрицательное число.  
Обязательные поля:

- Name – Название номенклатуры (тип данных *string*)
- Industry\_Id – Код родительской категории (тип данных *long*)
- UnitsName – Название единицы измерения (тип данных *string*)

Необязательные поля:

- External\_ID – код номенклатуры заказчика (тип данных *string*)
- Parent\_Id – код родительской номенклатуры (тип данных *long*)
- Item\_id – код номенклатуры (тип данных *long*)

**Внимание! Во избежание ошибок рекомендуется выбрать один из вариантов добавления номенклатур - или с указанием Item\_ID или без него!**

#### Возможные коды ошибок:

- 1008 - недопустимое значение кода категории.
- 1009 - недопустимое значение единицы измерения.
- 1010 - недопустимое значение названия номенклатуры.
- 1011 - недопустимое значение родительской номенклатуры.
- 1012 - номенклатура уже существует.
- 1013 - номенклатура восстановлена из удаленных.
- 1023 - номенклатура с укзанным кодом уже существует.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <NMC xsi:type="urn:NMCStruct" xmlns:urn="urn:APS_Remote_Types">
        <External_ID xsi:type="xsd:string">quae divum incedo</External_ID>
        <Name xsi:type="xsd:string">verrantque per auras</Name>
        <Parent_ID xsi:type="xsd:long">10</Parent_ID>
        <Industry_Id xsi:type="xsd:long">10</Industry_Id>
        <UnitsName xsi:type="xsd:string">per auras</UnitsName>
      </NMC>
    </urn:NMCAppend>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.2.3 NMCUpdate

#### Изменение номенклатуры в базе

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- nmcId – код номенклатуры для изменения (тип данных *long*).
- NMC – структура номенклатуры (тип данных [NMCStruct](#)<sup>[50]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном изменении в поле ErrorCode результата содержится 0 или 1013, в поле Result\_id – код измененной номенклатуры.

Поле Industry\_id входящей структуры игнорируются. Изменение Parent\_id возможно, если у номенклатуры отсутствуют дочерние элементы.

Обязательные поля:

- Name – Название номенклатуры (тип данных *string*)
- Industry\_Id – Код родительской категории (тип данных *long*)



- UnitsName – Название единицы измерения (тип данных *string*)
- Необязательные поля:
- External\_ID – код номенклатуры заказчика (тип данных *string*)
  - Parent\_Id – Код родительской номенклатуры (тип данных *long*)

#### Возможные коды ошибок:

- 1009 - недопустимое значение единицы измерения.
- 1010 - недопустимое значение названия номенклатуры.
- 1012 - номенклатура уже существует.
- 1013 - номенклатура восстановлена из удаленных.
- 1014 - недопустимое значение кода номенклатуры.
- 1015 - номенклатура не найдена.
- 1016 - номенклатура с входящими значениями найдена среди удаленных с другим кодом.
- 1017 - номенклатура с входящими значениями найдена с другим кодом.
- 1018 - попытка изменить Parent\_id у номенклатуры с дочерними элементами.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCUpdate soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </Auth>
      <nmcId xsi:type="xsd:long">10</nmcId>
      <NMC xsi:type="urn:NMCStruct" xmlns:urn="urn:APS_Remote_Types">
        <External_ID xsi:type="xsd:string">quae divum incedo</External_ID>
        <Name xsi:type="xsd:string">verrantque per auras</Name>
        <Parent_ID xsi:type="xsd:long">10</Parent_ID>
        <UnitsName xsi:type="xsd:string">per auras</UnitsName>
      </NMC>
    </urn:NMCUpdate>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.2.4 NMCSetRest

### Занесение остатка по номенклатуре в базу

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- NMCRest – структура остатка по номенклатуре (тип данных [NMCRestStruct](#)<sup>[50]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном внесении остатка в поле ErrorCode результата содержится 0, в поле Result\_id – код записи в базе

Обязательные поля:

- Company\_Id – код компании/филиала (тип данных *long*)
- NMC\_Item\_ID – код номенклатуры (тип данных *long*)
- Item\_Name – название номенклатуры (тип данных *string*)
- Is\_Unique\_Name – признак уникальности названия (тип данных *short*)
- Rest – остаток (тип данных *double*)

**Возможные коды ошибок:**

- 21111 - недопустимое значение кода компании/филиала.
- 21112 - недопустимое значение кода номенклатуры.
- 21113 - в базе существует несколько записей по коду номенклатуры с разными названиями, необходимо указать название.

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCSetRest soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <NMCRest xsi:type="urn:NMCRestStruct" xmlns:urn="urn:APS_Remote_Types">
        <Company_Id xsi:type="xsd:long">10</Company_Id>
        <NMC_Item_ID xsi:type="xsd:long">10</NMC_Item_ID>
        <Item_Name xsi:type="xsd:string">quae divum incedo</Item_Name>
        <Is_Unique_Name xsi:type="xsd:short">1</Is_Unique_Name>
        <Rest xsi:type="xsd:double">1.05</Rest>
      </NMCRest>
    </urn:NMCSetRest>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.2.5 NMCDelete****Удаление номенклатуры из базы**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- nmcId – код номенклатуры для удаления (тип данных *long*).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

**Особенности:**

При успешном удалении в поле ErrorCode результата содержится 0, в поле Result\_id - код удаленной номенклатуры

**Возможные коды ошибок:**

- -1 - номенклатура не найдена
- -2 - у номенклатуры присутствуют неудаленные дочерние элементы

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCDelete soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <nmcId xsi:type="xsd:long">10</nmcId>
    </urn:NMCDelete>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.2.6 IndustryGetCount

### Получение количества категорий.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество категорий.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:IndustryGetCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:IndustryGetCount>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.2.7 IndustryGetByN

### Получение данных n-ой категории из списка.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- N - порядковый номер в списке (тип данных *int*) (от 1 до количества возвращенного методом [IndustryGetCount](#)<sup>[19]</sup>)

Возвращается структура типа [IndustryInfoStruct](#)<sup>[55]</sup>

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:IndustryGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">3</N>
    </urn:IndustryGetByN>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.2.8 NMCGetCount

### Получение количества номенклатур.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество номенклатур.

#### Пример:

```
<soapenv:Envelope
```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
<soapenv:Header/>
<soapenv:Body>
  <urn:NMCGetCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
      <UserLogin xsi:type="xsd:string">user</UserLogin>
      <UserPassword xsi:type="xsd:string">md5</UserPassword>
    </auth>
  </urn:NMCGetCount>
</soapenv:Body>
</soapenv:Envelope>

```

## 2.2.9 NMCGetByN

### Получение данных n-ой номенклатуры из списка.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- N - порядковый номер в списке (тип данных *int*) (от 1 до количества возвращенного методом [NMCGetCount](#)<sup>[19]</sup>)

Возвращается структура типа [NMCInfoStruct](#)<sup>[55]</sup>

**Внимание! Во избежание нагрузки не рекомендуется часто производить выгрузку номенклатур!**

### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">3</N>
    </urn:NMCGetByN>
  </soapenv:Body>
</soapenv:Envelope>

```

## 2.2.10 NMCFind

### Поиск кода номенклатуры/группы номенклатур

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- NMCName – название номенклатуры/группы номенклатур (тип данных *string*).
- industryId - код категории (тип данных *long*) - необязательный параметр.
- unitsName - название единицы измерения (тип данных *string*) - необязательный параметр.
- externalId - внешний код номенклатуры/группы номенклатур (тип данных *long*) - необязательный параметр.

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

### Особенности:

Поиск регистрозависим (!). При успешном поиске в поле *ErrorCode* результата содержится 0, в поле *Result\_id* - код найденной номенклатуры/группы номенклатур.

**Возможные коды ошибок:**

- -1 - номенклатура не найдена.
- 22541 - недопустимое название номенклатуры.
- 22542 - недопустимый код категории.
- 22543 - недопустимое название единицы измерения.
- 22544 - найдено несколько номенклатур с заданными параметрами, необходимо добавить поля для поиска.

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCFind soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <NMCName xsi:type="xsd:string">verrantque per auras</NMCName>
      <industryId xsi:type="xsd:long">1</industryId>
      <unitsName xsi:type="xsd:string">шт</unitsName>
      <externalId xsi:type="xsd:string">nan</externalId>
    </urn:NMCFind>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.2.11 GetIndustries****Получение списка доступных категорий**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [CodeNamePairArrayStruct](#)<sup>[55]</sup>, в поле Code каждого элемента - ID категории, в поле Name - название.

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetIndustries soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </Auth>
    </urn:GetIndustries>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.2.12 NMCGetNewCount****Получение количества номенклатур, не считанных после последнего изменения данным пользователем.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество номенклатур.

**Пример:**

```
<soapenv:Envelope
```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
<soapenv:Header/>
<soapenv:Body>
  <urn:NMCGetNewCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
      <UserLogin xsi:type="xsd:string">user</UserLogin>
      <UserPassword xsi:type="xsd:string">md5</UserPassword>
    </auth>
  </urn:NMCGetNewCount>
</soapenv:Body>
</soapenv:Envelope>

```

### 2.2.13 NMCGetNext

**Получение данных очередной номенклатуры и проставление в базе АПС отметки, что данный пользователь считал эту номенклатуру.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [NMCInfoStruct](#)<sup>[55]</sup>

#### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCGetNext soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:NMCGetNext>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.2.14 NMCSetRests

**Занесение массива остатков по номенклатуре в базу**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- NMCRests – массив структур остатков по номенклатуре (тип данных [NMCRestStructArray](#)<sup>[57]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном внесении остатка в поле ErrorCode результата содержится 0, в поле Result\_id - размер массива. Если занесение хотя бы одного остатка произошло с ошибкой, занесение всего массива отменяется.

#### Возможные коды ошибок:

- 21111 - недопустимое значение кода компании/филиала.
- 21112 - недопустимое значение кода номенклатуры.
- 21113 - в базе существует несколько записей по коду номенклатуры с разными названиями, необходимо указать название.

#### Пример:

```

<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"

```

```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
<soapenv:Header/>
<soapenv:Body>
  <urn:NMCSetRests soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
      <UserLogin xsi:type="xsd:string">user4</UserLogin>
      <UserPassword xsi:type="xsd:string">password</UserPassword>
    </auth>
    <NMCRests xsi:type="urn:NMCRestStructArray" soapenc:arrayType="urn:NMCRestStruct[]" xmlns:urn="urn:APS_Remote_Types">
      <item>
        <Company_Id xsi:type="xsd:long">1</Company_Id>
        <NMC_Item_ID xsi:type="xsd:long">1</NMC_Item_ID>
        <Rest xsi:type="xsd:double">117.05</Rest>
      </item>
      <item>
        <Company_Id xsi:type="xsd:long">1</Company_Id>
        <NMC_Item_ID xsi:type="xsd:long">2</NMC_Item_ID>
        <Rest xsi:type="xsd:double">13.05</Rest>
      </item>
    </NMCRests>
  </urn:NMCSetRests>
</soapenv:Body>
</soapenv:Envelope>

```

### 2.2.15 NMCArrayAppend

#### Добавление массива номенклатур в базу

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- NMCArray – массив структур номенклатуры (тип данных [NMCStructArray](#)<sup>[57]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном добавлении в поле ErrorCode результата содержится 0 или 1013, в поле Result\_id – размер массива. При необходимости добавить группу номенклатур верхнего уровня тэг PARENT\_ID пропускается, или в него вносится 0 или отрицательное число. Если добавление хотя бы одной номенклатуры произошло с ошибкой, добавление всего массива отменяется.

#### Возможные коды ошибок:

- 1008 – недопустимое значение кода категории.
- 1009 – недопустимое значение единицы измерения.
- 1010 – недопустимое значение названия номенклатуры.
- 1011 – недопустимое значение родительской номенклатуры.
- 1012 – номенклатура уже существует.
- 1013 – номенклатура восстановлена из удаленных.

#### Пример:

```

<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:NMCArrayAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>

```

```

<NMCArray xsi:type="urn:NMCStructArray" soapenc:arrayType="urn:NMCStruct[]" xmlns:urn="urn:APS_Remote_Types">
  <item>
    <Name xsi:type="xsd:string">Второе комплектующее</Name>
    <Parent_ID xsi:type="xsd:long">1</Parent_ID>
    <Industry_Id xsi:type="xsd:long">1</Industry_Id>
    <UnitsName xsi:type="xsd:string">шт.</UnitsName>
  </item>
  <item>
    <Name xsi:type="xsd:string">Третье комплектующее</Name>
    <Parent_ID xsi:type="xsd:long">1</Parent_ID>
    <Industry_Id xsi:type="xsd:long">1</Industry_Id>
    <UnitsName xsi:type="xsd:string">шт.</UnitsName>
  </item>
</NMCArray>
</urn:NMCArrayAppend>
</soapenv:Body>
</soapenv:Envelope>

```

## 2.3 Заявки

### 2.3.1 RQAppend

#### Добавление заявки в базу (на этап исполнение).

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- RQ – структура заявки (тип данных [RQStruct](#)<sup>[50]</sup>).
- autoDistribute – признак автораспределения заявки (тип данных *short*)  
- необязательный параметр.

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Заявка должна содержать одну позицию. При отсутствии в позиции поля UnitsName значение единицы измерения подставляется из справочника номенклатур, аналогично для ItemName.

При отсутствии поля в структуре заявки поля Author\_Id автор заявки определяется исходя из данных авторизации.

При отсутствии в структуре заявки поля Executor\_Id и значении поля autoDistribute заявка распределяется на исполнителя по [правилам](#)<sup>[64]</sup>.

Обязательные поля в структуре заявки:

- Dep\_Id – код подразделения (тип данных *long*)
- Author\_Id – код автора заявки (тип данных *long*)
- BudgetItem\_Id – код статьи бюджета (тип данных *long*)
- Note – описание заявки (тип данных *string*)

Необязательные поля:

- Executor\_Id – код исполнителя заявки (тип данных *long*)
- DateCreate – дата создания (тип данных *dateTime*)
- DateChange – дата согласования (тип данных *dateTime*)
- IsEmergency – признак аварийности (тип данных *int*)
- External\_ID – код заявки заказчика (тип данных *long*)
- DeadLine – желаемая дата поставки (тип данных *dateTime*)

#### Возможные коды ошибок:

- 1001 – недопустимое значение кода филиала/подразделения
- 1002 – недопустимое значение кода автора заявки
- 1003 – недопустимое значение кода статьи бюджета
- 1004 – недопустимое значение описания заявки



**Пример:**

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <RQ xsi:type="urn:RQStruct" xmlns:urn="urn:APS_Remote_Types">
        <External_ID xsi:type="xsd:long">10</External_ID>
        <DateCreate xsi:type="xsd:dateTime">2004-02-14T20:44:14</DateCreate>
        <DateChange xsi:type="xsd:dateTime">2018-11-01T07:36:46+02:00</DateChange>
        <DeadLine xsi:type="xsd:date">2013-05-22+03:00</DeadLine>
        <Dep_Id xsi:type="xsd:long">10</Dep_Id>
        <Author_Id xsi:type="xsd:long">10</Author_Id>
        <BudgetItem_Id xsi:type="xsd:long">10</BudgetItem_Id>
        <IsEmergency xsi:type="xsd:int">3</IsEmergency>
        <Note xsi:type="xsd:string">fremunt celsa</Note>
        <Executor_Id xsi:type="xsd:long">10</Executor_Id>
        <RQItem xsi:type="urn:RQItemStruct">
          <NMC_Item_ID xsi:type="xsd:long">10</NMC_Item_ID>
          <ItemName xsi:type="xsd:string">ferant rapidi</ItemName>
          <Qty xsi:type="xsd:double">1.051732E7</Qty>
          <Price xsi:type="xsd:double">1.051732E7</Price>
          <Note xsi:type="xsd:string">sceptra tenens</Note>
          <UnitsName xsi:type="xsd:string">turbine corripuit scopuloque</UnitsName>
        </RQItem>
      </RQ>
    </urn:RQAppend>
  </soapenv:Body>
</soapenv:Envelope>

```

**2.3.2 RQItemAppend****Добавление позиции к заявке.****Входные параметры:**

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- RQid – код заявки (тип данных *long*).
- RQ – структура позиции в заявке (тип данных [RQItemStruct](#)<sup>[51]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

**Особенности:**

При отсутствии в позиции поля UnitsName значение единицы измерения подставляется из справочника номенклатур, аналогично для ItemName.

Обязательные поля:

- NMC\_Item\_ID – код номенклатуры (тип данных *long*)
- Qty – количество (тип данных *double*)

Необязательные поля:

- DetailId - код места поставки (тип данных *long*)
- ItemName – название позиции в заявке (тип данных *string*)
- Price – цена (тип данных *double*)
- Note – описание (тип данных *string*)
- UnitsName – единица измерения (тип данных *string*)
- CPV - идентификатор CPV (тип данных *string*)
- AddInfo - дополнительная информация (например, состав) (тип данных *string*)

**Возможные коды ошибок:**

- 1006 - недопустимое значение кода заявки
- 1007 - заявка в недопустимом состоянии

- 1008 - недопустимое значение номенклатуры позиции
- 1003 - недопустимое значение количества

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQItemAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <RQid xsi:type="xsd:long">10</RQid>
      <RQItem xsi:type="urn:RQItemStruct" xmlns:urn="urn:APS_Remote_Types">
        <NMC_Item_ID xsi:type="xsd:long">10</NMC_Item_ID>
        <ItemName xsi:type="xsd:string">quae divum incedo</ItemName>
        <Qty xsi:type="xsd:double">1.051732E7</Qty>
        <Price xsi:type="xsd:double">1.051732E7</Price>
        <Note xsi:type="xsd:string">verrantque per auras</Note>
        <UnitsName xsi:type="xsd:string">per auras</UnitsName>
      </RQItem>
    </urn:RQItemAppend>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.3.3 RQAppendWithItems****Добавление заявки в базу (на этап исполнение).**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- RQ – структура заявки (тип данных [RQStructWithItems](#)<sup>[51]</sup>).
- autoDistribute - признак автораспределения заявки (тип данных *short*)  
- необязательный параметр.

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

**Особенности:**

Заявка должна содержать не менее одной позиции. При отсутствии в позиции поля UnitsName значение единицы измерения подставляется из справочника номенклатур, аналогично для ItemName.

При отсутствии поля в структуре заявки поля Author\_Id автор заявки определяется исходя из данных авторизации.

При отсутствии в структуре заявки поля Executor\_Id и значении поля autoDistribute заявка распределяется на исполнителя по [правилам](#)<sup>[64]</sup>.

Обязательные поля в структуре заявки:

- Dep\_Id – код подразделения (тип данных *long*)
- Author\_Id – код автора заявки (тип данных *long*)
- BudgetItem\_Id – код статьи бюджета (тип данных *long*)
- Note – описание заявки (тип данных *string*)

Необязательные поля:

- Executor\_Id – код исполнителя заявки (тип данных *long*)
- DateCreate – дата создания (тип данных *dateTime*)
- DateChange – дата согласования (тип данных *dateTime*)
- IsEmergency – признак аварийности (тип данных *int*)
- External\_ID – код заявки заказчика (тип данных *long*)
- DeadLine – желаемая дата поставки (тип данных *dateTime*)

**Возможные коды ошибок:**

- 1001 - недопустимое значение кода филиала/подразделения
- 1002 - недопустимое значение кода автора заявки
- 1003 - недопустимое значение кода статьи бюджета
- 1004 - недопустимое значение описания заявки

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQAppendWithItems soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <RQ xsi:type="urn:RQStructWithItems" xmlns:urn="urn:APS_Remote_Types">
        <External_ID xsi:type="xsd:long">10</External_ID>
        <DateCreate xsi:type="xsd:dateTime">2004-02-14T20:44:14</DateCreate>
        <DateChange xsi:type="xsd:dateTime">2018-11-01T07:36:46+02:00</DateChange>
        <DeadLine xsi:type="xsd:date">2013-05-22+03:00</DeadLine>
        <Dep_Id xsi:type="xsd:long">10</Dep_Id>
        <Author_Id xsi:type="xsd:long">10</Author_Id>
        <BudgetItem_Id xsi:type="xsd:long">10</BudgetItem_Id>
        <IsEmergency xsi:type="xsd:int">3</IsEmergency>
        <Note xsi:type="xsd:string">fremunt celsa</Note>
        <Executor_Id xsi:type="xsd:long">10</Executor_Id>
        <RQItems xsi:type="urn:RQItemStructArray">
          <item>
            <NMC_Item_ID>2</NMC_Item_ID>
            <ItemName>Какие-то Биндери 25 мм, кольор хром, в пластиковом тубусі (за 48 шт.)</ItemName>
            <Qty>75.6</Qty>
            <Price>15.56</Price>
            <Note>Наименование какое-то</Note>
          </item>
          <item>
            <NMC_Item_ID>3</NMC_Item_ID>
            <ItemName>Розрахункова квитанція Ф РК-1</ItemName>
            <Qty>15</Qty>
            <Note>Зачем-то еще этот текст</Note>
          </item>
        </RQItems>
      </RQ>
    </urn:RQAppendWithItems>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.3.4 RQGetNewCount**

**Получение количества заявок, находящихся на стадии "Инициирование" и выше, не считанных на текущей стадии данным пользователем.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество заявок.

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQGetNewCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
```

```

        <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
            <UserLogin xsi:type="xsd:string">user</UserLogin>
            <UserPassword xsi:type="xsd:string">md5</UserPassword>
        </auth>
    </urn:RQGetNewCount>
</soapenv:Body>
</soapenv:Envelope>

```

### 2.3.5 RQGetNext

**Получение данных очередной заявки из находящихся на стадии "Инициирование" и выше.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- readOnly - признак проставления отметки о прочтении (тип данных *boolean*) - необязательный параметр

Возвращается структура типа [RQInfoStruct](#)<sup>[54]</sup>

#### Особенности:

При отсутствии параметра readOnly или если параметр readOnly равен false, происходит проставление в базе АПС отметки, что данный пользователь считал эту заявку на этой стадии.

#### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQGetNext soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <readOnly xsi:type="xsd:boolean">true</readOnly>
    </urn:RQGetNext>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.3.6 RQItemGetByN

**Получение данных N-ой позиции из заявки.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- RQId – код заявки (тип данных *long*) (поле RQId структуры RQ)
- N – порядковый номер в списке (тип данных *int*) (от 1 до количества возвращенного в поле ItemCount структуры [RQStruct](#)<sup>[50]</sup>)

Возвращается структура типа [RequestItemStruct](#)<sup>[55]</sup>

#### Пример:

```

<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQItemGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <RQId xsi:type="xsd:long">675</RQId>
      <N xsi:type="xsd:int">1</N>
    </urn:RQItemGetByN>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.3.7 RQGetByN

**Получение данных N-ой заявки из находящихся на стадии "Инициирование" и выше.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- N - порядковый номер заявки среди несчитанных

Возвращается структура типа [RQInfoStruct](#)<sup>[54]</sup>

#### Особенности:

Проставление в базе АПС отметки о прочтении не происходит.

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">60</N>
    </urn:RQGetByN>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.3.8 RQGetReturnedCount

**Получение количества возвращенных во внешнюю систему заявок, не считанных данным пользователем.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество заявок.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQGetReturnedCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:RQGetReturnedCount>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.3.9 RQReturnedGetNext

**Получение данных очередной заявки из возвращенных во внешнюю систему.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- readOnly – признак проставления отметки о прочтении (тип данных *boolean*) – необязательный параметр

Возвращается структура типа [RQInfoStruct](#)<sup>[54]</sup>

#### Особенности:

При отсутствии параметра readOnly или если параметр readOnly равен false, происходит проставление в базе АПС отметки, что данный пользователь считал эту заявку.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:RQReturnedGetNext soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <readOnly xsi:type="xsd:boolean">true</readOnly>
    </urn:RQReturnedGetNext>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.3.10 ClosedRQItemGetByN

#### Получение данных N-ой позиции из закрытой заявки.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- RQId – код заявки (тип данных *long*) (поле RQId структуры RQ)
- N – порядковый номер в списке (тип данных *int*) (от 1 до количества возвращенного в поле ItemCount структуры [RQStruct](#)<sup>[50]</sup>)

Возвращается структура типа [ClosedRequestItemStruct](#)<sup>[58]</sup>

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ClosedRQItemGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <RQId xsi:type="xsd:long">675</RQId>
      <N xsi:type="xsd:int">1</N>
    </urn:ClosedRQItemGetByN>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.4 Договоры и контрагенты

### 2.4.1 SupplierGet

#### Получение данных контрагента.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Edrpou – код ЕДРПОУ (тип данных *string*).

Возвращается структура типа [SupplierDataStruct](#)<sup>[52]</sup>

#### Возможные коды ошибок:

- -1 - значение не найдено

#### Особенности:

Выдаются данные, заполненные закупщиком в поля "Реквизиты контрагента для добавления в текст Договора, Спецификации и Доп.соглашения"

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:SupplierGet soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <Edrpou xsi:type="xsd:string">88888888</Edrpou>
    </urn:SupplierGet>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.4.2 SupplierGetOtherUsers

### Получение данных неосновного пользователя контрагента.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Edrpou – код ЕДРПОУ (тип данных *string*).
- index – индекс (тип данных *int*).

Возвращается структура типа [SupplierUserStruct](#)<sup>[52]</sup>

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:SupplierGetOtherUser soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user5</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <Edrpou xsi:type="xsd:string">88888888</Edrpou>
      <index xsi:type="xsd:int">1</index>
    </urn:SupplierGetOtherUser>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.4.3 ContractsGetNewCount

### Получение количества договоров, находящихся на стадии "Исполнение" и "Закрит", не считанных на текущей стадии данным пользователем.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество договоров.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ContractsGetNewCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:ContractsGetNewCount>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.4.4 ContractsGetNext

**Получение данных очередного договора/спецификации/доп.соглашения из находящихся на стадии "Исполнение" и "Закрит".**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- readOnly - признак проставления отметки о прочтении (тип данных *boolean*) - необязательный параметр

Возвращается структура типа [ContractInfoStruct](#)<sup>[53]</sup>

#### Особенности:

При отсутствии параметра readOnly или если параметр readOnly равен false, происходит проставление в базе АПС отметки, что данный пользователь считал этот договор на этой стадии.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ContractsGetNext soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <readOnly xsi:type="xsd:boolean">true</readOnly>
    </urn:ContractsGetNext>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.4.5 ContractAppend

**Добавление договора в базу (на этап исполнение).**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Contract – структура договора (тип данных [ContractInfo](#)<sup>[53]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - код



договора

#### Особенности:

При отсутствии поля в структуре договора поля Author\_Id автор определяется исходя из данных авторизации.

Обязательные поля:

- AuthorId – ID автора договора (тип данных *long*)
- CompanyId – ID группы компаний/компании заказчика (тип данных *long*)
- Name – наименование договора (тип данных *string*)
- Edrpou – код ЕДРПОУ контрагента (тип данных *string*)
- StartDate – дата договора (тип данных *dateTime*)
- ContrSum – сумма договора (тип данных *float*)

Обязательные поля в зависимости от настроек:

- SubCompId – ID предприятия заказчика (тип данных *long*)
- DepId – код подразделения (тип данных *long*)
- IndustryId – ID категории (тип данных *long*)
- BudgetItem\_Id – ID статьи бюджета (тип данных *long*)

Необязательные поля:

- RegNum – внутренний номер (тип данных *string*)
- Descr – описание (тип данных *string*)
- EndDate – дата окончания договора, если нет автопрологации (тип данных *dateTime*)
- Prolongation – статус "с автопрологацией", если 1 (тип данных *int*)

#### Возможные коды ошибок:

- 7002 - недопустимое значение кода автора договора
- 7004 - недопустимое значение номера договора
- 7005 - недопустимое значение названия договора
- 7006 - недопустимое значение кода группы компаний
- 7007 - недопустимое значение кода компании
- 7008 - недопустимое значение кода филиала/подразделения
- 7009 - недопустимое значение кода категории
- 7010 - недопустимое значение кода статьи бюджета
- 7011 - недопустимое значение кода контрагента
- 7012 - недопустимое значение даты начала
- 7013 - недопустимое значение даты конца
- 7014 - недопустимое значение суммы договора

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ContractAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <Contract xsi:type="urn:ContractInfo" xmlns:urn="urn:APS_Remote_Types">
        <Name xsi:type="xsd:string">quae divum incedo</Name>
        <ContractId xsi:type="xsd:long">10</ContractId>
        <ContrSpecId xsi:type="xsd:long">10</ContrSpecId>
        <ProcessId xsi:type="xsd:short">1</ProcessId>
        <Edrpou xsi:type="xsd:string">verrantque per auras</Edrpou>
        <CompanyId xsi:type="xsd:long">10</CompanyId>
        <StartDate xsi:type="xsd:dateTime">2008-11-15T18:52:58</StartDate>
        <Descr xsi:type="xsd:string">regemque dedit</Descr>
        <RegNum xsi:type="xsd:string">fremunt celsa</RegNum>
      </Contract>
    </urn:ContractAppend>
  </soapenv:Body>
</soapenv:Envelope>
```

```

<TypeDoc xsi:type="xsd:short">1</TypeDoc>
<EndDate xsi:type="xsd:dateTime">2012-09-13T16:00:34+03:00</EndDate>
<Prolongation xsi:type="xsd:short">1</Prolongation>
<CreateDate xsi:type="xsd:dateTime">2009-10-14T10:16:36</CreateDate>
<DepId xsi:type="xsd:long">10</DepId>
<AuthorId xsi:type="xsd:long">10</AuthorId>
<ContrSum xsi:type="xsd:float">1.5E2</ContrSum>
<IndustryId xsi:type="xsd:long">10</IndustryId>
<BudgetItem_Id xsi:type="xsd:long">10</BudgetItem_Id>
<SubCompId xsi:type="xsd:long">10</SubCompId>
</Contract>
</urn:ContractAppend>
</soapenv:Body>
</soapenv:Envelope>

```

## 2.4.6 ContractApprove

### Перевод договора на этап "исполнение".

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- contractId – код документа (тип данных *long*)
- typeDoc – тип документа: -1-договор 0-спецификация 1-дополнительное соглашение (тип данных *int*)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id – код документа

### Особенности:

Перевод договора на этап "Исполнение" возможен с этапов "Согласование с контрагентом" и "Утверждение контракта".

### Возможные коды ошибок:

- -1 – документ не найден
- -3 – недопустимое значение типа документа
- -4 – неверный статус договора
- -5 – в настройках системы статус спецификаций и доп.соглашений изменяется вместе со статусом договора.

### Пример:

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ContractApprove soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <contractId xsi:type="xsd:long">76</contractId>
      <typeDoc xsi:type="xsd:short">-1</typeDoc>
    </urn:ContractApprove>
  </soapenv:Body>
</soapenv:Envelope>

```

## 2.4.7 ContractsGetByContragent

### Получение массива данных договоров/спецификаций/доп.соглашений, заключенных с контрагентом.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- edrpou – ЕДРПОУ контрагента (тип данных *string*)

Возвращается структура типа [ContractInfoArrayStruct](#)<sup>[57]</sup>

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ContractsGetByContragent soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </Auth>
      <edrpou xsi:type="xsd:string">1111111111</edrpou>
    </urn:ContractsGetByContragent>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.4.8 ContractsGetByN**

**Получение данных N-го договора/спецификации/доп.соглашения из находящихся на стадии "Исполнение" и "Закрыт".**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- N - порядковый номер договора среди нескитанных

Возвращается структура типа [ContractInfoStruct](#)<sup>[53]</sup>

**Особенности:**

Проставление в базе АПС отметки о прочтении не происходит.

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:ContractsGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">60</N>
    </urn:ContractsGetByN>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.4.9 SuppliersGetNewCount**

**Получение количества контрагентов, не считанных данным пользователем.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество контрагентов.

**Особенности:**

Признак прочтения контрагента снимается при изменении закупщиком следующих полей в блоке "Реквизиты контрагента для добавления в текст Договора, Спецификации и Доп.соглашения":

- ЕДРПОУ

- Название
- Банк
- Номер счета
- МФО
- Номер плательщика НДС
- ИНН
- Страна
- Форма собственности

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:SuppliersGetNewCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:SuppliersGetNewCount>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.4.10 SuppliersGetNext****Получение данных очередного непрочитанного ранее контрагента.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>50)</sup>)
- readOnly - признак проставления отметки о прочтении (тип данных *boolean*) - необязательный параметр

Возвращается структура типа [SupplierDataStruct](#)<sup>52)</sup>

**Особенности:**

При отсутствии параметра readOnly или если параметр readOnly равен false, происходит проставление в базе АПС отметки, что данный пользователь считал этого контрагента.

Выдаются данные, заполненные закупщиком в поля "Реквизиты контрагента для добавления в текст Договора, Спецификации и Доп.соглашения".

Признак прочтения контрагента снимается при изменении закупщиком следующих полей в блоке "Реквизиты контрагента для добавления в текст Договора, Спецификации и Доп.соглашения":

- ЕДРПОУ
- Название
- Банк
- Номер счета
- МФО
- Номер плательщика НДС
- ИНН
- Страна
- Форма собственности

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
<soapenv:Header/>
<soapenv:Body>
  <urn:SuppliersGetNext soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
      <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
      <UserPassword xsi:type="xsd:string">md5</UserPassword>
    </auth>
    <readOnly xsi:type="xsd:boolean">true</readOnly>
  </urn:SuppliersGetNext>
</soapenv:Body>
</soapenv:Envelope>

```

### 2.4.11 SuppliersGetByN

**Получение данных N-го неп прочитанного контрагента.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- N - порядковый номер контрагента среди несчитанных

Возвращается структура типа [SupplierDataStruct](#)<sup>[52]</sup>

#### Особенности:

Проставление в базе АПС отметки о прочтении не происходит.

#### Пример:

```

<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:SuppliersGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">60</N>
    </urn:SuppliersGetByN>
  </soapenv:Body>
</soapenv:Envelope>

```

## 2.5 Заказы

### 2.5.1 OrdersGetNewCount

**Получение количества заказов на поставку, находящихся на стадии "На исполнении" и "Закрыт", не считанных на текущей стадии данным пользователем.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- companyId - код компании/филиала (тип данных *long*) - необязательный параметр

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - количество заказов.

#### Особенности:

При отсутствии параметра companyId или в случае companyId=0 учитываются заказы всех компаний/филиалов.

**Пример:**

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrdersGetNewCount soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
    </urn:OrdersGetNewCount>
  </soapenv:Body>
</soapenv:Envelope>

```

**2.5.2 OrdersGetNext****Получение данных очередного заказа из находящихся на стадии "На исполнении" и "Закрит".**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- readOnly - признак проставления отметки о прочтении (тип данных *boolean*) - необязательный параметр
- companyId - код компании/филиала (тип данных *long*) - необязательный параметр

Возвращается структура типа [OrderInfoStruct](#)<sup>[53]</sup>

**Особенности:**

При отсутствии параметра readOnly или если параметр readOnly равен false, происходит проставление в базе АПС отметки, что данный пользователь считал этот заказ на этой стадии.

При отсутствии поля companyId или в случае companyId=0 выдаются заказы всех компаний/филиалов.

**Пример:**

```

<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrdersGetNext soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <readOnly xsi:type="xsd:boolean">true</readOnly>
    </urn:OrdersGetNext>
  </soapenv:Body>
</soapenv:Envelope>

```

**2.5.3 OrderItemGetByN****Получение данных N-ой позиции из заказа на поставку.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- OrderId – код заказа (тип данных *long*) (поле OrderId структуры OrderInfo)
- N – порядковый номер в списке (тип данных *int*) (от 1 до количества возвращенного в поле ItemCount структуры [OrderInfo](#)<sup>[54]</sup>)

Возвращается структура типа [OrderItemStruct](#)<sup>[54]</sup>

#### Особенности:

В поле RQId структуры OrderItem возвращается код заявки, из которой создана позиция заказа.

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrderItemGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <OrderId xsi:type="xsd:long">675</OrderId>
      <N xsi:type="xsd:int">1</N>
    </urn:OrderItemGetByN>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.5.4 OrderItemSetAccepted

### Занесение данных по поставке позиции заказа в базу.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Contract – структура позиции (тип данных [OrderItemDataStruct](#)<sup>[56]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id – код позиции

#### Особенности:

В данный момент цена принятия игнорируется. История поставок не ведется, то есть передаваться должно не оприходованное в данный момент времени количество, а итоговая сумма, принятая кладовщиком.

Обязательные поля:

- OrderId – код заказа в базе (тип данных *long*)
- OrderItemId – код позиции заказа в базе (тип данных *long*)
- AcceptedQty – принятое количество (тип данных *double*)

Необязательные поля:

- AcceptedPrice – цена принятия (тип данных *double*)

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrderItemSetAccepted soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <OrderItemAccepted xsi:type="urn:OrderItemDataStruct" xmlns:urn="urn:APS_Remote_Types">
        <OrderId xsi:type="xsd:long">424</OrderId>
        <OrderItemId xsi:type="xsd:long">629</OrderItemId>
        <AcceptedQty xsi:type="xsd:double">9</AcceptedQty>
        <AcceptedPrice xsi:type="xsd:double">12.2</AcceptedPrice>
      </OrderItemAccepted>
    </urn:OrderItemSetAccepted>
  </soapenv:Body>
```

```
</soapenv:Envelope>
```

### 2.5.5 OrdersGetNextWithItems

**Получение данных очередного заказа из находящихся на стадии "На исполнении" и "Закрит" (вместе с позициями).**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- readOnly - признак проставления отметки о прочтении(тип данных *boolean*) - необязательный параметр
- companyId - код компании/филиала (тип данных *long*) - необязательный параметр

Возвращается структура типа [OrderInfoStructWithItems](#)<sup>[56]</sup>

#### Особенности:

При отсутствии параметра readOnly или если параметр readOnly равен false, происходит проставление в базе АПС отметки, что данный пользователь считал этот заказ на этой стадии.

При отсутствии поля companyId или в случае companyId=0 выдаются заказы всех компаний/филиалов.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrdersGetNextWithItems soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">no_aps_user</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <readOnly xsi:type="xsd:boolean">true</readOnly>
    </urn:OrdersGetNextWithItems>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.5.6 OrderAppend

**Добавление заказа в базу (на этап исполнение).**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Order – структура заказа (тип данных [OrderInfo](#)<sup>[54]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

Заказ должен содержать одну позицию. При отсутствии в позиции поля UnitsName значение единицы измерения подставляется из справочника номенклатур, аналогично для ItemName.

Обязательные поля:

- OrderName - название заказа (тип данных *string*)
- BudgetItemId - статья бюджета (тип данных *long*)
- DepId - ID подразделения (тип данных *long*)
- AuthorId - ID автора (тип данных *long*)
- Edrpou – код ЕДРПОУ контрагента (тип данных *string*)
- DeadLine - дата поставки (тип данных *date*)
- DateCreate – дата создания заказа (тип данных *dateTime*)



Необязательные поля:

- CompanyId – ID предприятия заказчика (тип данных *long*)
- DestinationId – ID получателя (тип данных *long*)
- SubCompId – ID компании заказчика (тип данных *long*)
- DetailId – код склада (тип данных *long*)

#### Возможные коды ошибок:

- 2001 – недопустимое значение кода филиала/подразделения или получателя
- 2002 – недопустимое значение кода автора заказа
- 2003 – недопустимое значение кода статьи бюджета
- 2004 – недопустимое значение названия заказа
- 2006 – недопустимое значение группы компаний
- 2006 – недопустимое значение заказчика
- 2011 – недопустимое значение контрагента
- 2012 – недопустимое значение даты создания
- 2013 – недопустимое значение даты поставки
- 2014 – недопустимое значение кода склада

#### Пример:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrderAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <Order xsi:type="urn:OrderInfo" xmlns:urn="urn:APS_Remote_Types">
        <Edrpou xsi:type="xsd:string">88888888</Edrpou>
        <CompanyId xsi:type="xsd:long">1</CompanyId>
        <DateCreate xsi:type="xsd:dateTime">2008-11-15T18:52:58</DateCreate>
        <ItemCount xsi:type="xsd:int">3</ItemCount>
        <SubCompId xsi:type="xsd:long">1</SubCompId>
        <OrderType xsi:type="xsd:short">1</OrderType>
        <OrderName xsi:type="xsd:string">regemque dedit</OrderName>
        <BudgetItemId xsi:type="xsd:long">1</BudgetItemId>
        <DepId xsi:type="xsd:long">1</DepId>
        <DestinationId xsi:type="xsd:long">1</DestinationId>
        <DeadLine xsi:type="xsd:date">2002-11-05</DeadLine>
        <DeliveryType xsi:type="xsd:int">0</DeliveryType>
        <AuthorId xsi:type="xsd:long">1</AuthorId>
        <DetailId>4</DetailId>
        <OrderItem xsi:type="urn:RQItemStruct">
          <NMC_Item_ID xsi:type="xsd:long">10</NMC_Item_ID>
          <ItemName xsi:type="xsd:string">profundum quippe ferant</ItemName>
          <Qty xsi:type="xsd:double">1.051732</Qty>
          <Price xsi:type="xsd:double">128</Price>
          <Note xsi:type="xsd:string">et carcere</Note>
          <UnitsName xsi:type="xsd:string">iovis rapidum iaculata</UnitsName>
          <RQId xsi:type="xsd:long">10</RQId>
        </OrderItem>
      </Order>
    </urn:OrderAppend>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.5.7 OrderAppendWithItems

#### Добавление заказа в базу (на этап исполнение).

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Order – структура заказа (тип данных [OrderInfo](#)<sup>[54]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

**Особенности:**

Заказ должен содержать не менее одной позиции. При отсутствии в позиции поля UnitsName значение единицы измерения подставляется из справочника номенклатур, аналогично для ItemName.

Обязательные поля:

- OrderName - название заказа (тип данных *string*)
- BudgetItemId - статья бюджета (тип данных *long*)
- DepId - ID подразделения (тип данных *long*)
- AuthorId - ID автора (тип данных *long*)
- Edrpou - код ЕДРПОУ контрагента (тип данных *string*)
- DeadLine - дата поставки (тип данных *date*)
- DateCreate - дата создания заказа (тип данных *dateTime*)

Необязательные поля:

- CompanyId - ID предприятия заказчика (тип данных *long*)
- DestinationId - ID получателя (тип данных *long*)
- SubCompId - ID компании заказчика (тип данных *long*)
- DetailId - код склада (тип данных *long*)

**Возможные коды ошибок:**

- 2001 - недопустимое значение кода филиала/подразделения или получателя
- 2002 - недопустимое значение кода автора заказа
- 2003 - недопустимое значение кода статьи бюджета
- 2004 - недопустимое значение названия заказа
- 2006 - недопустимое значение группы компаний
- 2006 - недопустимое значение заказчика
- 2011 - недопустимое значение контрагента
- 2012 - недопустимое значение даты создания
- 2013 - недопустимое значение даты поставки
- 2014 - недопустимое значение кода склада

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrderAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <Order xsi:type="urn:OrderInfo" xmlns:urn="urn:APS_Remote_Types">
        <Edrpou xsi:type="xsd:string">88888888</Edrpou>
        <CompanyId xsi:type="xsd:long">1</CompanyId>
        <DateCreate xsi:type="xsd:dateTime">2008-11-15T18:52:58</DateCreate>
        <ItemCount xsi:type="xsd:int">3</ItemCount>
        <SubCompId xsi:type="xsd:long">1</SubCompId>
        <OrderType xsi:type="xsd:short">1</OrderType>
        <OrderName xsi:type="xsd:string">regemque deditt</OrderName>
        <BudgetItemId xsi:type="xsd:long">1</BudgetItemId>
        <DepId xsi:type="xsd:long">1</DepId>
        <DestinationId xsi:type="xsd:long">1</DestinationId>
        <DeadLine xsi:type="xsd:date">2002-11-05</DeadLine>
        <DeliveryType xsi:type="xsd:int">0</DeliveryType>
        <AuthorId xsi:type="xsd:long">1</AuthorId>
        <DetailId>4</DetailId>
        <OrderItems xsi:type="urn:RQItemStructArray" soapenc:arrayType="urn:RQItemStruct[]">
          <item>
            <NMC_Item_ID xsi:type="xsd:long">10</NMC_Item_ID>
            <ItemName xsi:type="xsd:string">profundum quippe ferant</ItemName>
            <Qty xsi:type="xsd:double">1.051732</Qty>
```

```

        <Price xsi:type="xsd:double">128</Price>
        <Note xsi:type="xsd:string">et carcere</Note>
        <UnitsName xsi:type="xsd:string">iovis rapidum iaculata</UnitsName>
    </item>
    <item>
        <NMC_Item_ID xsi:type="xsd:long">11</NMC_Item_ID>
        <ItemName xsi:type="xsd:string"></ItemName>
        <Qty xsi:type="xsd:double">11</Qty>
        <Price xsi:type="xsd:double">128.15</Price>
        <Note xsi:type="xsd:string">carcere</Note>
        <UnitsName xsi:type="xsd:string">iaculata</UnitsName>
    </item>
</OrderItems>
</Order>
</urn:OrderAppend>
</soapenv:Body>
</soapenv:Envelope>

```

## 2.5.8 OrderClose

### Заккрытие заказа.

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- orderId – код заказа (тип данных *long*)
- acceptItems - признак принятия позиций (тип данных boolean)

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - код заказа

### Особенности:

Если acceptItems=true, проставляется признак принятия позиций на склад

### Возможные коды ошибок:

- -1 - документ не найден

### Пример:

```

<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrderClose soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user4</UserLogin>
        <UserPassword xsi:type="xsd:string">password</UserPassword>
      </auth>
      <orderId xsi:type="xsd:long">424</orderId>
      <acceptItems xsi:type="xsd:boolean">true</acceptItems>
    </urn:OrderClose>
  </soapenv:Body>
</soapenv:Envelope>

```

## 2.5.9 OrdersGetByN

### Получение данных N-го непрочитанного заказа из находящихся на стадии "Исполнение" и "Закрыт".

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- N - порядковый номер заказа среди нечитанных
- companyId - код компании/филиала(тип данных *long*) - необязательный параметр

Возвращается структура типа [OrderInfoStruct](#)<sup>[53]</sup>

### Особенности:

Проставление в базе АПС отметки о прочтении не происходит.  
При отсутствии поля `companyId` или в случае `companyId=0`  
выдаются заказы всех компаний/филиалов.

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrdersGetByN soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">60</N>
    </urn:OrdersGetByN>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.5.10 OrdersGetByNWithItems**

**Получение данных N-го непрочитанного заказа из находящихся на стадии "Исполнение" и "Закрыт" (вместе с позициями).**

Входные параметры:

- `auth` – данные авторизации (тип данных [AuthStruct](#)<sup>501</sup>)
- `N` - порядковый номер заказа среди несчитанных
- `companyId` - код компании/филиала (тип данных *long*) - необязательный параметр

Возвращается структура типа [OrderInfoStructWithItems](#)<sup>561</sup>

**Особенности:**

Проставление в базе АПС отметки о прочтении не происходит.  
При отсутствии поля `companyId` или в случае `companyId=0`  
выдаются заказы всех компаний/филиалов.

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:OrdersGetByNWithItems soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <N xsi:type="xsd:int">60</N>
    </urn:OrdersGetByNWithItems>
  </soapenv:Body>
</soapenv:Envelope>
```

**2.6 Платежи****2.6.1 PaymentInsert**

**Занесение данных по платежу в базу.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Payment – структура договора (тип данных [PaymentInfo](#)<sup>[57]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id - код платежа

#### Особенности:

Обязательные поля:

- PaymentSum - сумма платежа (тип данных *decimal*)
- PaymentDate - дата платежа (тип данных *dateTime*)

Необязательные поля:

- OrderId – ID заказа на поставку (тип данных *long*)
- ContractId - ID договора (тип данных *long*)
- ContrSpecId - ID спецификации (тип данных *long*)

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:PaymentInsert soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </auth>
      <Payment xsi:type="urn:PaymentInfo" xmlns:urn="urn:APS_Remote_Types">
        <OrderId xsi:type="xsd:long">56</OrderId>
        <ContractId xsi:type="xsd:long">56</ContractId>
        <ContractSpecId xsi:type="xsd:long">66</ContractSpecId>
        <PaymentSum xsi:type="xsd:decimal">10.5600</PaymentSum>
        <PaymentDate xsi:type="xsd:dateTime">2015-02-12T20:49:14</PaymentDate>
      </Payment>
    </urn:PaymentInsert>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.7 Компании

### 2.7.1 CompanyAppend

#### Добавление компании заказчика в базу

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- Company – структура компании (тип данных [CompanyData](#)<sup>[57]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном добавлении в поле ErrorCode результата содержится 0 или 91013, в поле Result\_id - код добавленной компании. При необходимости добавить компанию верхнего уровня тэг ParentId пропускается, или в него вносится 0 или отрицательное число.

Обязательные поля:

- CompanyName – Название компании (тип данных *string*)

Необязательные поля:

- ParentId – id родительской компании (тип данных *long*)
- FullName – полное название компании (тип данных *string*)

- Edrpou – код ЕДРПОУ (тип данных *string*)
- DirectorName – ФИО директора (тип данных *string*)
- AccountantName – ФИО бухгалтера (тип данных *string*)
- Note – примечание (тип данных *string*)

#### Возможные коды ошибок:

- 91010 - недопустимое значение названия компании.
- 91011 - недопустимое значение родительской компании.
- 91012 - компания с таким именем уже существует.
- 91013 - компания восстановлена из удаленных.

#### Пример:

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:CompanyAppend soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </auth>
      <Company xsi:type="urn:CompanyData" xmlns:urn="urn:APS_Remote_Types">
        <CompanyName xsi:type="xsd:string">Ромашка</CompanyName>
        <ParentId xsi:type="xsd:long">1</ParentId>
        <FullName xsi:type="xsd:string">ЗАО "Ромашка Продакшнз"</FullName>
        <Edrpou xsi:type="xsd:string">09090909</Edrpou>
        <DirectorName xsi:type="xsd:string">Билл Гейтс</DirectorName>
        <AccountantName xsi:type="xsd:string">Сергей Брин</AccountantName>
        <Note xsi:type="xsd:string">Примечание</Note>
      </Company>
    </urn:CompanyAppend>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.7.2 CompanyUpdate

### Изменение компании заказчика в базе

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- companyId – код номенклатуры для изменения (тип данных *long*).
- Company – структура номенклатуры (тип данных [CompanyData](#)<sup>[51]</sup>).

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>

#### Особенности:

При успешном изменении в поле ErrorCode результата содержится 0, в поле Result\_id - код измененной компании.

Обязательные поля:

- CompanyName – Название компании (тип данных *string*)

Необязательные поля:

- ParentId – id родительской компании (тип данных *long*)
- FullName – полное название компании (тип данных *string*)
- Edrpou – код ЕДРПОУ (тип данных *string*)
- DirectorName – ФИО директора (тип данных *string*)
- AccountantName – ФИО бухгалтера (тип данных *string*)
- Note – примечание (тип данных *string*)
- Deleted - признак удаления (тип данных *short*)

#### Возможные коды ошибок:

- 91010 - недопустимое значение названия компании.

- 91011 - недопустимое значение родительской компании.
- 91012 - компания с таким именем уже существует.
- 91014 - недопустимое значение кода компании.
- 91015 - компания не найдена.
- 91016 - компания с таким именем найдена среди удаленных с другим кодом.

**Пример:**

```
<soapenv:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:CompanyUpdate soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">gero et</UserLogin>
        <UserPassword xsi:type="xsd:string">sonoras imperio</UserPassword>
      </Auth>
      <companyId xsi:type="xsd:long">10</nmId>
      <Company xsi:type="urn:CompanyData" xmlns:urn="urn:APS_Remote_Types">
        <CompanyName xsi:type="xsd:string">Ромашка</CompanyName>
        <ParentId xsi:type="xsd:long">1</ParentId>
        <FullName xsi:type="xsd:string">ЗАО "Ромашка Продакшнз"</FullName>
        <Edrpou xsi:type="xsd:string">09090909</Edrpou>
        <DirectorName xsi:type="xsd:string">Билл Гейтс</DirectorName>
        <AccountantName xsi:type="xsd:string">Сергей Брин</AccountantName>
        <Note xsi:type="xsd:string">Примечание</Note>
        <Deleted xsi:type="xsd:short">0</Note>
      </Company>
    </urn:CompanyUpdate>
  </soapenv:Body>
</soapenv:Envelope>
```

## 2.8 Служебные методы

### 2.8.1 EntitySetSynced

**Проставление признака чтения сущности данным пользователем в журнал.**

Входные параметры:

- auth – данные авторизации (тип данных [AuthStruct](#)<sup>[50]</sup>)
- entityType – тип сущности (тип данных *int*)
- entityId – код сущности
- processId – этап, на котором находится сущность.

Возвращается структура типа [IDStruct](#)<sup>[50]</sup>, в поле Result\_id – код сущности

**Особенности:**

Тип сущности может принимать следующие значения:

- 2 – заявка, возвращенная во внешнюю систему.
- 1 – контрагент
- 0 – номенклатура
- 2 – заявка
- 4 – заказ на поставку
- 5 – договор
- 6 – спецификация
- 7 – доп. соглашение

Для неположительных значений entityType поле processId игнорируется.

**Возможные коды ошибок:**

- -1 - недопустимый код сущности
- -2 - недопустимый тип сущности
- -3 - неверный этап

**Пример:**

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:APS_RemoteIntf-IAPS_Remote">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:EntitySetSynced soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <Auth xsi:type="urn:AuthStruct" xmlns:urn="urn:APS_Remote_Types">
        <UserLogin xsi:type="xsd:string">user2</UserLogin>
        <UserPassword xsi:type="xsd:string">md5</UserPassword>
      </Auth>
      <entityType xsi:type="xsd:int">-2</entityType>
      <entityId xsi:type="xsd:long">389</entityId>
      <processId xsi:type="xsd:int">4</processId>
    </urn:EntitySetSynced>
  </soapenv:Body>
</soapenv:Envelope>
```



# Типы данных

## 3 Типы данных

### 3.1 AuthStruct

#### структура данных авторизации

Состоит из следующих полей:

- UserLogin – имя пользователя (тип данных *string*)
- UserPassword – MD5 хэш пароля или пароль (в зависимости от настроек, тип данных *string*)

### 3.2 IDStruct

#### универсальная структура для возврата ID

Состоит из следующих полей:

- Result\_id – возвращаемый код (тип данных *long*)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

### 3.3 NMCStruct

#### структура данных номенклатуры

Состоит из следующих полей:

- External\_ID – код номенклатуры заказчика (тип данных *string*)
- Name – Название номенклатуры (тип данных *string*)
- Parent\_Id – Код родительской номенклатуры (тип данных *long*)
- Industry\_Id – Код родительской категории (тип данных *long*)
- UnitsName – Название единицы измерения (тип данных *string*)
- Item\_ID – Код номенклатуры в системе APS (тип данных *long*)
- UnitsId – Код единицы измерения (тип данных *long*)
- Deleted – Признак удаления номенклатуры (тип данных *short*)

### 3.4 NMCRestStruct

#### структура остатка по номенклатуре

Состоит из следующих полей:

- Company\_Id – код компании/филиала (тип данных *long*)
- NMC\_Item\_ID – код номенклатуры (тип данных *long*)
- Item\_Name – название номенклатуры (тип данных *string*)
- Is\_Unique\_Name – признак уникальности названия (тип данных *short*)
- Rest – остаток (тип данных *double*)

### 3.5 RQStruct

#### структура заявки

Состоит из следующих полей:

- External\_ID – код заявки заказчика (тип данных *long*)
- DateCreate – дата создания (тип данных *dateTime*)
- DateChange – дата согласования (тип данных *dateTime*)

- **Deadline** – желаемая дата поставки (тип данных *dateTime*)
- **Dep\_Id** – код подразделения (тип данных *long*)
- **Author\_Id** – код автора заявки (тип данных *long*)
- **BudgetItem\_Id** – код статьи бюджета (тип данных *long*)
- **IsEmergency** – признак аварийности (тип данных *int*)
- **Note** – описание заявки (тип данных *string*)
- **Executor\_Id** – код исполнителя заявки (тип данных *long*)
- **RQItem** – позиция в заявке (тип данных - [RQItemStruct](#)<sup>[51]</sup>)
- **RQId** - ID заявки (тип данных *long*)
- **ItemCount** - количество позиций (тип данных *int*)
- **ProcessId** - статус заявки (тип данных *short*)
- **LastComment** - последний комментарий (тип данных *string*)
- **ExternalN** - номер заявки заказчика (тип данных *string*)

## 3.6 RQStructWithItems

### структура заявки

Состоит из следующих полей:

- **External\_ID** – код заявки заказчика (тип данных *long*)
- **DateCreate** – дата создания (тип данных *dateTime*)
- **DateChange** – дата согласования (тип данных *dateTime*)
- **Deadline** – желаемая дата поставки (тип данных *dateTime*)
- **Dep\_Id** – код подразделения (тип данных *long*)
- **Author\_Id** – код автора заявки (тип данных *long*)
- **BudgetItem\_Id** – код статьи бюджета (тип данных *long*)
- **IsEmergency** – признак аварийности (тип данных *int*)
- **Note** – описание заявки (тип данных *string*)
- **Executor\_Id** – код исполнителя заявки (тип данных *long*)
- **RQItems** – позиции в заявке (тип данных - [RQItemStructArray](#)<sup>[51]</sup>)
- **RQId** - ID заявки (тип данных *long*)
- **ItemCount** - количество позиций (тип данных *int*)
- **ProcessId** - статус заявки (тип данных *short*)
- **LastComment** - последний комментарий (тип данных *string*)
- **ExternalN** - номер заявки заказчика (тип данных *string*)

## 3.7 RQItemStructArray

### массив позиций заявки/заказа

Состоит из объектов типа [RQItemStruct](#)<sup>[51]</sup>

## 3.8 RQItemStruct

### структура позиции в заявке/заказе

Состоит из следующих полей:

- **NMC\_Item\_ID** – код номенклатуры (тип данных *long*)
- **ItemName** – название позиции в заявке (тип данных *string*)
- **Qty** – количество (тип данных *double*)
- **Price** – цена (тип данных *double*)
- **Note** – описание (тип данных *string*)
- **UnitsName** – единица измерения (тип данных *string*)
- **RQItemId** - код позиции заявки в базе (тип данных *long*)
- **RQId** - код заявки в базе (тип данных *long*)

- DetailId - код места поставки (тип данных *long*)
- CPV - идентификатор CPV (тип данных *string*)
- AddInfo - дополнительная информация (например, состав) (тип данных *string*)

## 3.9 SupplierDataStruct

### структура для возврата контрагента

Состоит из следующих полей:

- SupplierData – данные контрагента (тип данных [SupplierData](#)<sup>[52]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.10 SupplierData

### данные контрагента

Состоит из следующих полей:

- SupplierName – название контрагента (тип данных *string*)
- Edrpou – код ЕДРПОУ (тип данных *string*)
- Ownership – организационная форма (тип данных *string*)
- Country – страна (тип данных *string*)
- Inn – ИНН (тип данных *string*)
- NdsNumber – № свидетельства плательщика НДС (тип данных *string*)
- Bank\_Name – название банка (тип данных *string*)
- Bank\_Acc – номер счета в банке (тип данных *string*)
- Bank\_MFO – МФО банка (тип данных *string*)
- ContactUser – контактный пользователь (тип данных [SupplierUser](#)<sup>[52]</sup>)
- OtherUserCount - количество других пользователей (тип данных *int*)
- SupplierId - код контрагента в системе АПС (тип данных *long*)

## 3.11 SupplierUserStruct

### структура для возврата пользователя контрагента

Состоит из следующих полей:

- SupplierUser – данные пользователя (тип данных [SupplierUser](#)<sup>[52]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.12 SupplierUser

### пользователь контрагента

Состоит из следующих полей:

- EMail – E-Mail пользователя (тип данных *string*)
- Position – должность (тип данных *string*)
- Phone – номер телефона (тип данных *string*)
- FIO – ФИО (тип данных *string*)

## 3.13 ContractInfoStruct

### структура для возврата данных договора

Состоит из следующих полей:

- ContractInfo – данные договора (тип данных [ContractInfo](#)<sup>[53]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.14 ContractInfo

### данные договора

Состоит из следующих полей:

- Name – наименование договора (тип данных *string*)
- ContractId – ID договора (тип данных *long*)
- ContrSpecId – ID допсоглашения/спецификации, если документ является допсоглашением, спецификацией (тип данных *long*)
- ProcessId – статус договора (тип данных *int*)
- Edrpou – код ЕДРПОУ контрагента (тип данных *string*)
- CompanyId – ID группы компаний/компании заказчика (тип данных *long*)
- StartDate – дата договора (тип данных *dateTime*)
- Descr – описание (тип данных *string*)
- RegNum – внутренний номер (тип данных *string*)
- TypeDoc – тип документа: -1-договор 0-спецификация 1-дополнительное соглашение (тип данных *int*)
- EndDate – дата окончания договора, если нет автопродолжения (тип данных *dateTime*)
- Prolongation – статус "с автопродолжением", если 1 (тип данных *int*)
- CreateDate – дата создания договора (тип данных *dateTime*)
- DepId – код подразделения (тип данных *long*)
- AuthorId – ID автора договора (тип данных *long*)
- ContrSum – сумма договора (тип данных *float*)
- IndustryId – ID категории (тип данных *long*)
- BudgetItem\_Id – ID статьи бюджета (тип данных *long*)
- SubCompId – ID предприятия заказчика (тип данных *long*)

## 3.15 OrderInfoStruct

### структура для возврата данных заказа на поставку

Состоит из следующих полей:

- OrderInfo – данные договора (тип данных [OrderInfo](#)<sup>[53]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.16 OrderInfo

### данные заказа на поставку

Состоит из следующих полей:

- Edrpou – код ЕДРПОУ контрагента (тип данных *string*)
- CompanyId – ID предприятия заказчика (тип данных *long*)
- OrderId – ID заказа на поставку (тип данных *long*)
- Number – номер заказа (тип данных *string*)
- DateCreate – дата создания заказа (тип данных *dateTime*)
- ItemCount – количество позиций в заказе (тип данных *int*)
- ContractId – ID договора (тип данных *long*)
- ContrSpecId – ID спецификации (тип данных *long*)
- ProcessId – статус заказа (тип данных *int*)
- SubCompId – ID компании заказчика (тип данных *long*)
- OrderType – тип заказа (тип данных *int*)
- OrderName – название заказа (тип данных *string*)
- BudgetItemId – статья бюджета (тип данных *long*)
- DepId – ID подразделения (тип данных *long*)
- DestinationId – ID получателя (тип данных *long*)
- DeadLine – дата поставки (тип данных *dateTime*)
- DeliveryType – условия поставки (тип данных *int*)
- AuthorId – ID автора (тип данных *long*)
- OrderItem – первая позиция в заказе (тип данных [RQItemStruct](#)<sup>[51]</sup>)
- DetailId – код склада (тип данных *long*)

## 3.17 OrderItemStruct

### структура для возврата данных позиции в заказе на поставку

Состоит из следующих полей:

- OrderItem – данные позиции (тип данных [RQItemStruct](#)<sup>[51]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.18 Industry

### Категория

Состоит из следующих полей:

- IndustryId – ID категории (тип данных *long*)
- Name – наименование категории (тип данных *string*)

## 3.19 RQInfoStruct

### структура для возврата данных заявки

Состоит из следующих полей:

- RQ – данные договора (тип данных [RQStruct](#)<sup>[50]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.20 NMCIInfoStruct

### структура для возврата номенклатуры

Состоит из следующих полей:

- NMC – данные номенклатуры (тип данных [NMCStruct](#)<sup>[50]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.21 IndustryInfoStruct

### структура для возврата категории

Состоит из следующих полей:

- Industry – данные категории (тип данных [Industry](#)<sup>[54]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.22 RequestItemStruct

### структура для возврата данных позиции в заявке

Состоит из следующих полей:

- RQItem – данные позиции (тип данных [RQItemStruct](#)<sup>[51]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.23 CodeNamePairArrayStruct

### Структура для возврата массива пар типа "код-название"

Состоит из следующих полей:

- CodeNameArray – массив пар (тип данных [CodeNamePairArray](#)<sup>[55]</sup>)
- Count – Количество элементов в массиве (тип данных *int*)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.24 CodeNamePairArray

### массив пар типа "код-название"

Состоит из объектов типа [CodeNamePair](#)<sup>[56]</sup>

## 3.25 CodeNamePair

### пара типа "код-название"

Состоит из следующих полей:

- Code – возвращаемый код (тип данных *long*)
- Name – возвращаемое название (тип данных *string*)

## 3.26 OrderItemDataStruct

### структура данных по поставке позиции в заказе

Состоит из следующих полей:

- OrderId - код заказа в базе (тип данных *long*)
- OrderItemId - код позиции заказа в базе (тип данных *long*)
- AcceptedQty – принятое количество (тип данных *double*)
- AcceptedPrice – цена принятия (тип данных *double*)

## 3.27 OrderInfoStructWithItems

### структура для возврата данных заказа на поставку (вместе с позициями)

Состоит из следующих полей:

- OrderInfo – данные договора с позициями (тип данных [OrderInfoWithItems](#)<sup>[56]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.28 OrderInfoWithItems

### данные заказа на поставку (с позициями)

Состоит из следующих полей:

- Edrpou – код ЕДРПОУ контрагента (тип данных *string*)
- CompanyId – ID предприятия заказчика (тип данных *long*)
- OrderId – ID заказа на поставку (тип данных *long*)
- Number – номер заказа (тип данных *string*)
- DateCreate – дата создания заказа (тип данных *dateTime*)
- ItemCount – количество позиций в заказе (тип данных *int*)
- ContractId - ID договора (тип данных *long*)
- ContrSpecId - ID спецификации (тип данных *long*)
- ProcessId – статус заказа (тип данных *int*)
- OrderItems – позиции в заказе (тип данных - [RQItemStructArray](#)<sup>[57]</sup>)
- SubCompId - ID компании заказчика (тип данных *long*)
- OrderType - тип заказа (тип данных *int*)
- OrderName - название заказа (тип данных *string*)
- BudgetItemId - статья бюджета (тип данных *long*)
- DepId - ID подразделения (тип данных *long*)
- DestinationId - ID получателя (тип данных *long*)
- DeadLine - дата поставки (тип данных *dateTime*)
- DeliveryType - условия поставки (тип данных *int*)
- AuthorId - ID автора (тип данных *long*)
- DetailId - код склада (тип данных *long*)



## 3.29 PaymentInfo

### данные платежа

Состоит из следующих полей:

- OrderId – ID заказа на поставку (тип данных *long*)
- ContractId – ID договора (тип данных *long*)
- ContrSpecId – ID спецификации (тип данных *long*)
- PaymentSum – сумма платежа (тип данных *decimal*)
- PaymentDate – дата платежа (тип данных *dateTime*)

## 3.30 NMCRestStructArray

### массив остатков по номенклатурам

Состоит из объектов типа [NMCRestStruct](#)<sup>[50]</sup>

## 3.31 NMCStructArray

### массив данных номенклатур

Состоит из объектов типа [NMCStruct](#)<sup>[50]</sup>

## 3.32 ContractInfoArray

### массив данных договоров

Состоит из объектов типа [ContractInfo](#)<sup>[53]</sup>

## 3.33 ContractInfoArrayStruct

### Структура для возврата массива данных договоров

Состоит из следующих полей:

- ContractInfoArray – массив данных договоров (тип данных [ContractInfoArray](#)<sup>[57]</sup>)
- Count – Количество элементов в массиве (тип данных *int*)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.34 CompanyData

### данные компании заказчика

Состоит из следующих полей:

- CompanyId – id компании заказчика (тип данных *long*)
- CompanyName – название компании заказчика (тип данных *string*)
- ParentId – id родительской компании (тип данных *long*)
- FullName – полное название компании (тип данных *string*)
- Edrpou – код ЕДРПОУ (тип данных *string*)
- DirectorName – ФИО директора (тип данных *string*)
- AccountantName – ФИО бухгалтера (тип данных *string*)
- Note – примечание (тип данных *string*)
- Deleted – признак удаления (тип данных *short*)

## 3.35 CompanyDataStruct

### структура для возврата компании заказчика

Состоит из следующих полей:

- CompanyData – данные контрагента (тип данных [CompanyData](#)<sup>[57]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## 3.36 ClosedRQItemStruct

### структура позиции в закрытой заявке

Состоит из следующих полей:

- NMC\_Item\_ID – код номенклатуры (тип данных *long*)
- ItemName – название позиции в заявке (тип данных *string*)
- Qty – количество (тип данных *double*)
- Price – цена (тип данных *double*)
- Note – описание (тип данных *string*)
- UnitsName – единица измерения (тип данных *string*)
- RQItemId – код позиции заявки в базе (тип данных *long*)
- RQId – код заявки в базе (тип данных *long*)
- DetailId – код места поставки (тип данных *long*)
- CPV – идентификатор CPV (тип данных *string*)
- AddInfo – дополнительная информация (например, состав) (тип данных *string*)
- Status – [статус закрытия](#)<sup>[60]</sup> позиции (тип данных *int*)
- ContrId – id контракта, в спецификацию которого ушла позиция (тип данных *long*)
- ContrSpecId – id спецификации, в которую ушла позиция (тип данных *long*)
- TndId – id тендера, в который ушла позиция (тип данных *long*)
- OrderId – id заказа, в который ушла позиция (тип данных *long*)
- Edrpou – ЕДРПОУ контрагента (тип данных *string*)
- FinalPrice – финальная цена (тип данных *double*)

## 3.37 ClosedRequestItemStruct

### структура для возврата данных позиции в закрытой заявке

Состоит из следующих полей:

- RQItem – данные позиции (тип данных [ClosedRQItemStruct](#)<sup>[58]</sup>)
- IsError – Признак ошибки (тип данных *boolean*)
- ErrorCode – Код ошибки (тип данных *short*, при успешном выполнении 0)
- ErrorStr – описание ошибки (тип данных *string*)

## Статусы объектов

## 4 Статусы объектов

### 4.1 Заявки

---

- 2 - Инициирование
- 3 - Согласование
- 4 - Распределение
- 5 - В работе
- 6 - Завершено

### 4.2 Договоры

---

- 4 - Исполнение
- 5 - Закрѳт

### 4.3 Заказы

---

- 5 - Исполнение
- 6 - Приемка
- 7 - Подтверждение выполнения
- 8 - Завершен

### 4.4 Закрѳтые позиции заявок

---

- 0 - отменена
- 1 - выполнена: закрѳта тендером, контрактом и т.д.
- 2 - закрѳта с контрагентом после инициирования
- 3 - закрѳта с контрагентом после согласования

## Коды ошибок

## 5 Коды ошибок

Прочие возможные коды ошибок:

- 99 - отсутствуют данные авторизации
- 198 - отсутствуют данные для добавления
- 197 - в заявке отсутствуют позиции
- 179 - нет прав на редактирование номенклатур
- 189 - нет прав на создание заявок
- 188 - нет прав на добавление позиций к заявке
- 181 - нет доступа к редактированию справочников
- 180 - нет доступа к справочникам
- 170 - нет доступа к договорам
- 160 - нет доступа к заявкам
- 150 - нет доступа к контрагентам
- 210 - нет доступа к заказам
- 217 - в заказе отсутствуют позиции
- 219 - нет прав на создание заказа
- -11 - недопустимый пользователь
- -2 - пустой пароль
- -7 - дублирование пользователя
- -4 - неактивный пользователь
- -5 - неверный пароль

# **Правила автораспределения заявки**

## 6 Правила автораспределения заявки

Правила для пользователя, на которого распределяется заявка:

Общие условия:

- он принадлежит тому же предприятию и подразделению, по которому создана заявка.

Алгоритм выбора:

- среди пользователей, отвечающих 3-м вышеуказанным условиям, выбирается пользователь со статусом "Руководитель". Если таких пользователей несколько, выбирается пользователь с наименьшим по модулю user\_id.
- если среди пользователей, отвечающим 3-м вышеуказанным условиям, нет "Руководителей", то выбирается пользователь с наименьшим по модулю user\_id.