



Банк России

№ 87  
88

# ВЕСТНИК БАНКА РОССИИ

Нормативные акты  
и оперативная информация

11 ноября 2020



## **ВЕСТНИК БАНКА РОССИИ**

**№ 87–88 (2223–2224)**

**11 ноября 2020**

### **Редакционный совет Банка России:**

#### **Председатель совета**

К.В. Юдаева

#### **Заместитель председателя совета**

Д.В. Тулин

#### **Члены совета:**

О.Н. Скоробогатова, С.А. Швецов, Р.Н. Вестеровский,  
А.Б. Заботкин, О.В. Полякова, Д.Г. Скобелкин,  
Н.Ю. Иванова, А.Г. Гузнов, К.В. Тремасов,  
Е.В. Прокунина, А.Г. Морозов, М.В. Рыклина,  
Т.А. Забродина, Е.Б. Федорова, О.В. Кувшинова

#### **Ответственный секретарь совета**

Е.Ю. Ключева

### **Учредитель**

Центральный банк Российской Федерации  
107016, Москва, ул. Неглинная, 12  
Адрес официального сайта Банка России:  
[www.cbr.ru](http://www.cbr.ru)  
Тел. 8 (495) 771-43-73,  
e-mail: [mvg@cbr.ru](mailto:mvg@cbr.ru)

Издатель и распространитель: АО «АЭИ «ПРАЙМ»  
119021, Москва, Зубовский бул., 4  
Тел. 8 (495) 645-37-00,  
факс 8 (495) 637-45-60,  
e-mail: [sales01@1prime.ru](mailto:sales01@1prime.ru), [www.1prime.ru](http://www.1prime.ru)

Отпечатано в ООО «Типография Возрождение»  
117105, Москва, Варшавское ш., 37а, стр. 2

Издание зарегистрировано Федеральной службой по надзору  
в сфере связи, информационных технологий и массовых  
коммуникаций.

Регистрационный номер ПИ № ФС77-47238

© **Центральный банк Российской Федерации, 1994**



СТАНДАРТЫ БАНКА РОССИИ

**ОТКРЫТЫЕ  
БАНКОВСКИЕ  
ИНТЕРФЕЙСЫ**

## Содержание

Стандарт Банка России от 23.10.2020 “Открытые банковские интерфейсы. Общие положения” .....	3
Стандарт Банка России от 23.10.2020 “Открытые банковские интерфейсы. Инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации” .....	19
Стандарт Банка России от 23.10.2020 “Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной” .....	89



СТАНДАРТ БАНКА РОССИИ

# ОТКРЫТЫЕ БАНКОВСКИЕ ИНТЕРФЕЙСЫ

Общие положения

Дата введения: 2020-10-23

Москва  
2020

## Оглавление

<b>1. История изменений</b> .....	<b>5</b>
<b>2. Введение</b> .....	<b>6</b>
<b>3. Область применения</b> .....	<b>6</b>
<b>4. Термины и определения</b> .....	<b>7</b>
<b>5. Общие положения</b> .....	<b>8</b>
5.1. Структура комплекса стандартов .....	8
5.2. Принципы архитектуры .....	8
5.3. Использование международного стандарта ISO 20022 .....	8
5.4. Управление Стандартами Открытых банковских интерфейсов .....	8
5.5. Изменения Стандартов Открытых банковских интерфейсов .....	9
5.6. Независимость Открытых банковских интерфейсов от способа платежа .....	9
5.7. Физический уровень описания сообщений .....	9
5.8. Защита прав потребителей .....	9
<b>6. Роли и участники процесса</b> .....	<b>11</b>
<b>7. Принципы построения взаимодействия</b> .....	<b>12</b>
7.1. Предоставление Пользователем долгосрочного согласия .....	12
7.2. Предоставление Пользователем краткосрочного согласия .....	13
7.3. Обмен данными между Сторонним поставщиком и ППУ .....	13
<b>8. Взаимодействие участников процесса предоставления данных</b> .....	<b>14</b>
8.1. Инициирование одиночного перевода денежных средств .....	15
8.2. Создание долгосрочного согласия на получение информации о счете .....	16
8.3. Получение списка счетов .....	17
8.4. Получение детальной информации о счете .....	17
8.5. Получение баланса по счету .....	18
8.6. Получение списка транзакций по счету .....	18

## 1. История изменений

Версия	Дата	Автор	Комментарий
1.0.0	16.12.2019	АФТ, Направление открытых API	Создана первая версия документа
1.0.1	22.01.2020	АФТ, Направление открытых API	Внесены минорные изменения
1.0.2	10.02.2020	АФТ, Направление открытых API	Внесены минорные изменения по замечаниям Центрального банка Российской Федерации
1.1.0	17.04.2020	АФТ, Направление открытых API	Добавлен раздел “5.8. Защита прав потребителей” по предложению Службы по защите прав потребителей и обеспечению доступности финансовых услуг Банка России
1.2.1	17.06.2020	АФТ, Направление открытых API	Внесены изменения по замечаниям рабочей группы ТК122 ПКЗ

## 2. Введение

Настоящий Стандарт содержит принципы и рекомендации по обмену данными для осуществления взаимодействия через Открытые банковские интерфейсы.

## 3. Область применения

Настоящий Стандарт рекомендован к использованию организациями при обмене финансовыми сообщениями, связанными с:

- получением информации о банковском счете;
- переводом денежных средств в валюте Российской Федерации.

Настоящий Стандарт предназначен для:

- участников получения информации о банковском счете (банки и их клиенты, а также Сторонние поставщики<sup>1, 2</sup>);
- участников перевода денежных средств (банки и их клиенты, а также Сторонние поставщики<sup>3</sup>);
- разработчиков информационного и программного обеспечения.

Положения настоящего Стандарта применяются на добровольной основе.

По предложениям участников среды Открытых банковских интерфейсов настоящий Стандарт может дополняться принятыми в международной практике ролями и сценариями.

<sup>1</sup> Определение понятия вводится в 6-м разделе текущего документа.

<sup>2</sup> Получают доступ к информации с согласия владельца счета.

<sup>3</sup> Получают доступ к счету с согласия владельца счета.

## 4. Термины и определения

В настоящем документе применяются следующие термины и определения:

Наименование	Описание
API (Application Programming Interface)	Набор процедур, протоколов и инструментов для создания программных приложений. API определяет, как программные компоненты должны взаимодействовать
Открытые банковские интерфейсы	Общедоступные интерфейсы прикладного программирования (API), которые предоставляют разработчикам программный доступ к финансовым данным в финансовых сервисах
Пользователь	Физическое или юридическое лицо, являющееся плательщиком или получателем средств
Среда Открытых банковских интерфейсов, Среда	Комплекс стандартов Открытых банковских интерфейсов, управление, системы, процессы, безопасность и процедуры, используемые для поддержки участников
Участники среды Открытых банковских интерфейсов	Пользователи, банки и прочие организации, которые участвуют в создании и развитии среды Открытых банковских интерфейсов
Плательщик	Пользователь, осуществляющий перевод денежных средств (либо от имени которого осуществляется перевод денежных средств)
Получатель средств	Пользователь, в пользу которого осуществляется перевод денежных средств
Стандарт ISO 20022 <sup>4</sup>	Международный стандарт обмена электронными сообщениями между организациями финансовой отрасли
Многофакторная аутентификация Пользователя	Аутентификация, которая основана на использовании двух или более элементов, классифицированных как знания, владение и неотъемлемость. Эти пункты являются независимыми, поскольку нарушение одного не угрожает надежности других
YAML	Формат сериализации данных, концептуально близкий к языкам разметки, но ориентированный на удобство ввода-вывода типичных структур данных многих языков программирования

Таблица 1. Термины и определения

<sup>4</sup> Детальное описание можно посмотреть на сайте: <https://www.iso20022.org/>.

## 5. Общие положения

Информационный обмен между банками и Сторонними поставщиками осуществляется посредством электронных сообщений, формируемых на стороне банков и на стороне Сторонних поставщиков посредством Открытых банковских интерфейсов.

### 5.1. Структура комплекса стандартов

Начальная версия комплекса стандартов по описанию Среды Открытых банковских интерфейсов состоит из следующих документов:

1. **“Открытые банковские интерфейсы. Общие положения”** — текущий документ, описывающий общие правила построения среды Открытых банковских интерфейсов, участников и их роли, принципы построения взаимодействия и сценарии использования.
2. **“Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”** — документ, описывающий технические детали протокола аутентификации и обеспечения информационной безопасности взаимодействия через интерфейсы Среды.
3. **“Открытые банковские интерфейсы. Инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации”** — документ, описывающий технические детали взаимодействия через интерфейсы Среды для Стороннего поставщика с ролью СППУ<sup>5</sup>.
4. **“Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной”** — документ, описывающий технические детали взаимодействия через интерфейсы Среды для Стороннего поставщика с ролью СПИУ<sup>6</sup>.

Комплекс стандартов Открытых банковских интерфейсов используется ППУ<sup>7</sup> для разработки интерфейсов, которые предоставляют Сторонним поставщикам (СППУ и СПИУ) доступ к счетам, управляемым ППУ, с согласия Пользователя.

### 5.2. Принципы архитектуры

Архитектура среды Открытых банковских интерфейсов соответствует концепции RESTful API<sup>8</sup>.

Данная концепция была выбрана на основании отзывов участников рынка, а также согласно опыту мировых практик.

### 5.3. Использование международного стандарта ISO 20022

Принципы, которые применялись при использовании элементов и компонентов сообщений из стандарта ISO 20022:

- следование семантике сообщений из ISO 20022;
- уменьшение иерархии схем сообщений API для удобства участников среды Открытых банковских интерфейсов;
- адаптация наименований и состава элементов схем сообщений API для удобства участников среды Открытых банковских интерфейсов;
- использование типов данных элементов сообщений из ISO 20022.

### 5.4. Управление Стандартами Открытых банковских интерфейсов

Управление Стандартами Открытых банковских интерфейсов осуществляется Банком России совместно с Ассоциацией развития финансовых технологий (далее — АФТ).

Согласование, утверждение и публикация контрольных экземпляров Стандартов Открытых банковских интерфейсов осуществляется Банком России.

АФТ взаимодействует с подразделениями Банка России, федеральными органами исполнительной власти Российской Федерации, профессиональными объединениями участников рынка, национальными и международными комитетами по стандартизации, центральными (национальными) банками

<sup>5</sup> Определение роли вводится в 6-м разделе текущего документа.

<sup>6</sup> Определение роли вводится в 6-м разделе текущего документа.

<sup>7</sup> Определение роли вводится в 6-м разделе текущего документа.

<sup>8</sup> Детальное описание подхода можно посмотреть на сайте: <https://restfulapi.net/>.

государств — участников интеграционных объединений с участием Российской Федерации, Евразийской экономической комиссией, а также другими организациями.

АФТ обладает специализированной автоматизированной системой с необходимым набором функций для осуществления разработки, ведения и публикации Стандартов Открытых банковских интерфейсов, тестовой площадкой для апробации вносимых в них изменений, а также обеспечивает сопровождение специализированного портала по Стандартам Открытых банковских интерфейсов в сети Интернет.

## 5.5. Изменения Стандартов Открытых банковских интерфейсов

Изменения Стандартов Открытых банковских интерфейсов могут происходить в зависимости от:

- потребностей участников рынка;
- изменений законодательства Российской Федерации (например, изменений в федеральных законах и (или) нормативных актах Банка России).

Изменения Стандартов Открытых банковских интерфейсов происходят следующим образом:

- изменения выносятся на рассмотрение членами АФТ;
- при положительном решении о принятии изменений на площадке АФТ изменения выносятся на рассмотрение 3-м подкомитетом (Технологии основных финансовых (банковских) операций) Технического комитета 122 Банка России;
- при положительном решении Технического комитета редакции Стандартов утверждаются и публикуются Банком России.

Управление процессом обратной совместимости между разными версиями Стандартов описано в технической части Стандартов Открытых банковских интерфейсов и осуществляется АФТ.

## 5.6. Независимость Открытых банковских интерфейсов от способа платежа

Открытые банковские интерфейсы разрабатываются таким образом, чтобы они не зависели от отдельно взятой платежной схемы, которая отвечает за проведение платежа.

Как результат, формат предоставления данных не проектируется так, чтобы они соответствовали только отдельно взятым сообщениям.

## 5.7. Физический уровень описания сообщений

На физическом уровне при проектировании сообщений используется язык описания интерфейсов OpenAPI 3-й версии<sup>9</sup> в формате YAML<sup>10</sup>.

## 5.8. Защита прав потребителей

Участники среды Открытых банковских интерфейсов обеспечивают раскрытие минимального объема информации Пользователям, который включает помимо прочего следующую информацию:

- полное и (или) сокращенное (при наличии) фирменное наименование участника среды Открытых банковских интерфейсов, адрес в пределах места нахождения, адрес электронной почты, контактный телефон, адрес официального сайта в информационно-телекоммуникационной сети “Интернет” (при наличии);
- порядок получения услуги (сервиса) Пользователем, возможные риски Пользователя;
- права Пользователя и порядок действий при возникновении технических неполадок на стороне участника среды Открытых банковских интерфейсов;
- способы и адреса направления обращений (жалоб) Пользователем;
- способы защиты прав Пользователя;
- порядок и срок предоставления по требованию Пользователя документов (в т.ч. в форме электронного документа), связанных с оказанием услуг (предоставлением сервиса) участником среды Открытых банковских интерфейсов;
- порядок предоставления Пользователю экземпляра договора и (или) иного документа, подтверждающего оказание услуги (предоставление сервиса) участником среды Открытых банковских интерфейсов в письменной форме (в т.ч. в форме электронного документа);

<sup>9</sup> Детальное описание подхода можно посмотреть на сайте: <https://swagger.io/specification/>.

<sup>10</sup> Детальное описание подхода можно посмотреть на сайте: <https://yaml.org/spec/1.2/spec.html>.

- способы направления информации Пользователю в связи с оказанием услуги (предоставления сервиса) участником среды Открытых банковских интерфейсов;
- сведения о распределении ответственности между участниками среды Открытых банковских интерфейсов, а также третьими лицами, привлекаемыми участниками среды Открытых банковских интерфейсов, для оказания услуги (сервиса) за неисполнение либо ненадлежащее исполнение обязанностей в рамках оказания услуг (предоставления сервиса) Пользователю;
- ограничения (при наличии), условия и способы использования услуг (сервиса) участника среды Открытых банковских интерфейсов, изменения условий и отказа от услуг (сервиса) Пользователем;
- порядок возмещения убытков, понесенных Пользователем;
- сведения о предполагаемых сроках восстановления режима оказания услуг (предоставления сервиса) в случае возникновения технических неполадок;
- требований, предъявляемых участником среды Открытых банковских интерфейсов к иным участникам среды Открытых банковских интерфейсов, во взаимодействии с которыми осуществляется оказание услуги (предоставление сервиса) Пользователю.

Участник среды Открытых банковских интерфейсов обеспечивает соответствие порядка раскрытия информации Пользователю следующим критериям:

- обеспечение возможности доступа к информации Пользователем на равных правах и в равном объеме;
- исключение раскрытия информации, которая может повлечь неоднозначное толкование свойств услуги (сервиса);
- обеспечение изложения информации на русском языке (за исключением указания обозначений на иностранном языке, являющихся товарными знаками) в доступной форме (с использованием удобочитаемых шрифтов, форматов предоставления);
- обеспечение возможности на основании волеизъявления Пользователя получать уведомление о совершении операции / оказании услуги в порядке и на условиях, установленных договором, но не позднее одного дня со дня совершения операции / оказания услуги (если иное не предусмотрено законодательством Российской Федерации);

- обеспечение возможности получения по требованию Пользователя документов (в т.ч. в форме электронного документа), связанных с оказанием услуги (предоставлением сервиса);
- обеспечение возможности получения экземпляра договора и (или) иного документа, подтверждающего оказание услуги (предоставления сервиса) Пользователем.

Участник среды Открытых банковских интерфейсов обеспечивает:

- конфиденциальность и защиту персональных данных Пользователя в случае, если получателем услуги (сервиса) выступает физическое лицо;
- защиту имущественных интересов Пользователя в части несанкционированного совершения операций с денежными средствами и электронными денежными средствами пользователя;
- уведомление Пользователя о распределении ответственности между участниками среды Открытых банковских интерфейсов, а также иными третьими лицами за неисполнение либо ненадлежащее исполнение обязанностей в рамках оказания услуги (предоставления сервиса);
- использование только целевой необходимой и достаточной информации при совершении операций (предоставлении сервиса) посредством Открытых банковских интерфейсов;
- исключение неавторизованных Пользователем операций;
- исключение использования для целей формирования предложения Пользователю информации, на обработку которой явным образом не получено согласие от Пользователя;
- исключение возможности обуславливать предоставление одной услуги (сервиса) в зависимости от предоставления иной услуги (сервиса);
- исключение возможности потери контроля над данными, переданными Пользователем участникам среды Открытых банковских интерфейсов для целей оказания услуги (предоставления сервиса);
- возможность Пользователя осуществлять оплату услуги (сервиса) в доступной для Пользователя форме;
- исключение навязывания заведомо невыгодных условий оказания услуг (сервиса) Пользователю.

## 6. Роли и участники процесса

Среда Открытых банковских интерфейсов определяет следующих участников:

**Пользователь** — физическое или юридическое лицо, являющееся плательщиком или получателем средств.

**Сторонний поставщик** — юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ).

**ППУ** (Поставщик платежных услуг) — кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы.

Среда Открытых банковских интерфейсов определяет следующие роли участников:

**СПИУ** (Сторонний поставщик информационных услуг) — юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя.

**СППУ** (Сторонний поставщик платежных услуг) — юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств.

Среда Открытых банковских интерфейсов определяет следующие взаимоотношения между участниками и ролями:

Роль \ Участник	Пользователь	ППУ	Сторонний поставщик
<b>СПИУ</b>	Нет	Да	Да
<b>СППУ</b>	Нет	Да	Да

Таблица 2. Роли и участники

Открытые банковские интерфейсы регламентируют взаимодействие только между следующими участниками:

- **ППУ** предоставляет Открытые банковские интерфейсы для Стороннего поставщика. Получает сообщения запросов через свои Открытые банковские интерфейсы и отправляет соответствующие ответные сообщения Стороннему поставщику.
- **Сторонний поставщик** может получить доступ к счету Пользователя, управляемому ППУ через Открытые банковские интерфейсы, при согласии Пользователя. Сторонний поставщик отправляет сообщения запроса через Открытые банковские интерфейсы ППУ и получает соответствующие ответные сообщения от этого ППУ.

## 7. Принципы построения взаимодействия

На следующем рисунке показано логическое месторасположение Открытых банковских интерфейсов в создаваемой Среде:

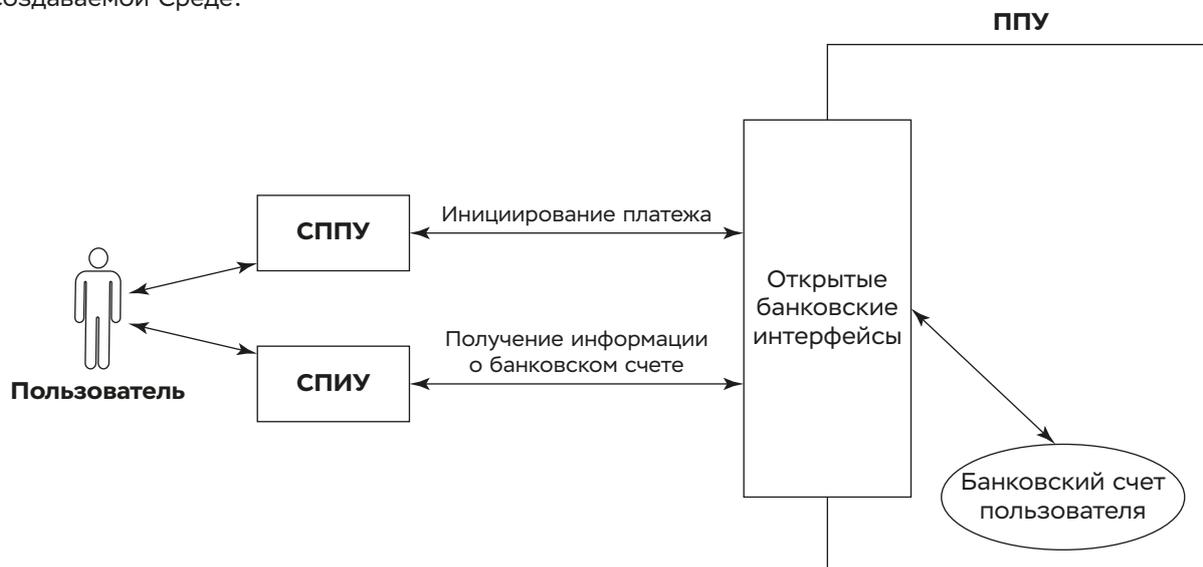


Рисунок 1. Принципы построения взаимодействия

Между участниками среды Открытых банковских интерфейсов возможны следующие типовые взаимодействия:

1. Предоставление Пользователем ППУ долгосрочного согласия на использование его данных Сторонним поставщиком.
2. Предоставление Пользователем ППУ краткосрочного согласия на инициирование перевода денежных средств Сторонним поставщиком.
3. Обмен данными между Сторонним поставщиком и ППУ с согласия Пользователя.

### 7.1. Предоставление Пользователем долгосрочного согласия

Долгосрочное согласие дается Пользователем на стороне ППУ для осуществления обмена данными между Сторонним поставщиком и ППУ на длительный срок, без непосредственного участия Пользователя. После предоставления такого согласия взаимодействие происходит следующим образом: Пользователь — Сторонний поставщик, Сторонний поставщик — ППУ.

В любой момент Пользователь может отозвать долгосрочное согласие как через Открытые банковские интерфейсы, так и через предоставляемые средства ППУ.

Долгосрочное согласие на получение данных о банковском счете схематически можно показать следующим образом:

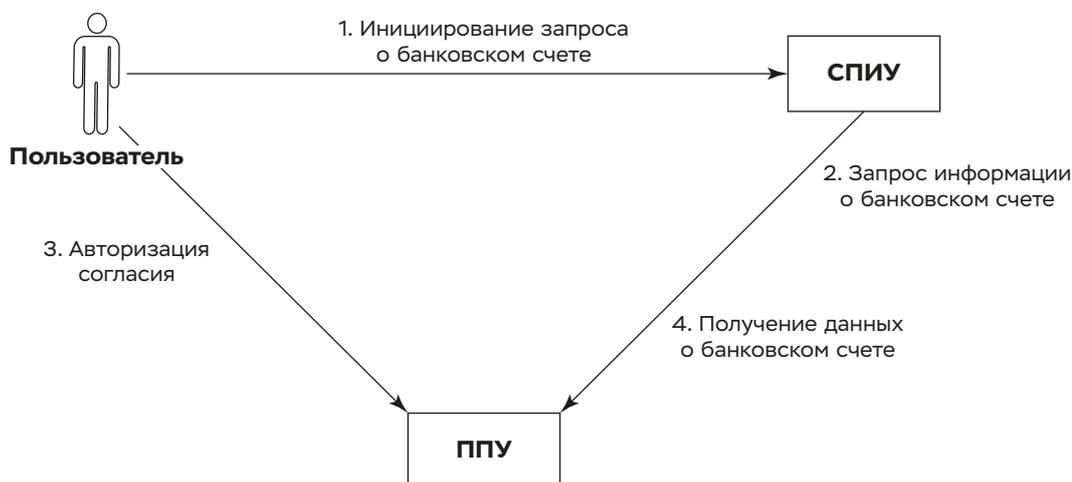


Рисунок 2. Предоставление Пользователем долгосрочного согласия

## 7.2. Предоставление Пользователем краткосрочного согласия

Краткосрочное согласие дается Пользователем при необходимости для перевода денежных средств с его банковского счета. Краткосрочное согласие является подтверждением инициирования перевода денежных средств Сторонним поставщиком.

Краткосрочное согласие на инициирование платежной услуги схематически можно показать следующим образом:

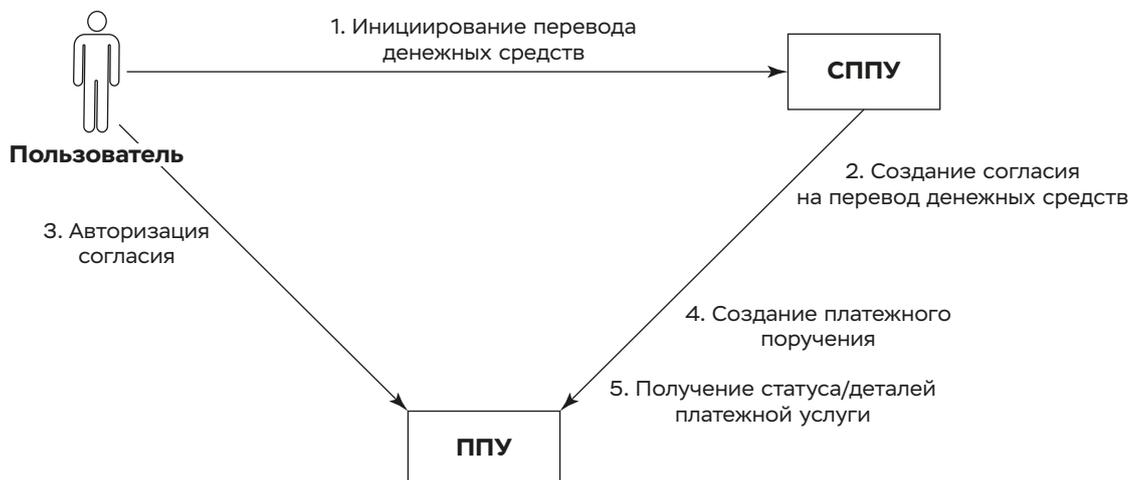


Рисунок 3. Предоставление Пользователем краткосрочного согласия

## 7.3. Обмен данными между Сторонним поставщиком и ППУ

Обмен данными между Сторонним поставщиком и ППУ происходит после предоставления Пользователем ППУ краткосрочного или долгосрочного согласия.

На следующем рисунке показано взаимодействие Стороннего поставщика и ППУ через Открытые банковские интерфейсы:

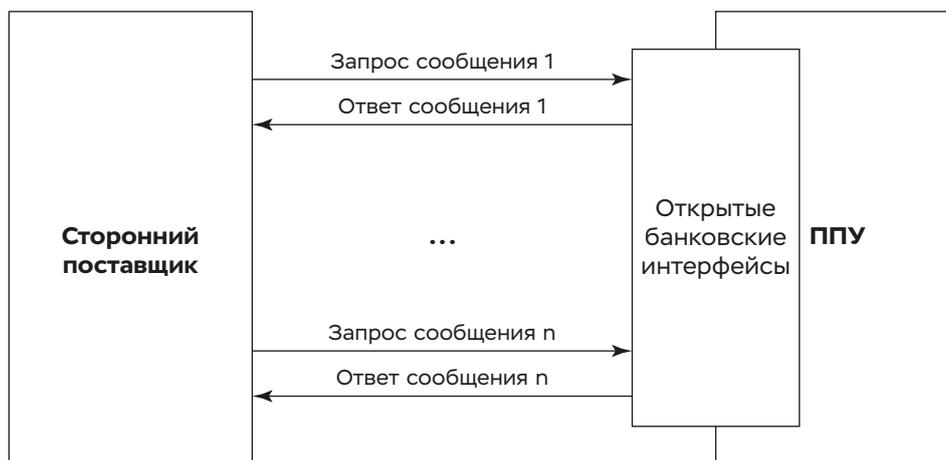


Рисунок 4. Обмен данными между Сторонним поставщиком и ППУ

Как уже упоминалось выше, ППУ может также выступать в одной из ролей Стороннего поставщика (СПИУ и (или) СППУ) и может получать доступ к Открытым банковским интерфейсам других ППУ.

## 8. Взаимодействие участников процесса предоставления данных

Открытые банковские интерфейсы поддерживают следующие сценарии использования обмена данными:

№	Сценарий использования	Типовое взаимодействие	Роль Стороннего поставщика	Участие Пользователя	Документ технического стандарта
1	Инициирование одиночного перевода денежных средств	Предоставление Пользователем краткосрочного согласия Обмен данными между Сторонним поставщиком и ППУ	СППУ	Да	Открытые банковские интерфейсы. Инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации: <ul style="list-style-type: none"> <li>• Внутренние переводы денежных средств, спецификация API</li> </ul>
2	Создание долгосрочного согласия на получение информации о счете	Предоставление Пользователем долгосрочного согласия	СПИУ	Да	Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной: <ul style="list-style-type: none"> <li>• Согласие на доступ к счету, спецификация API</li> </ul>
3	Получение списка счетов, к которым получен доступ	Обмен данными между Сторонним поставщиком и ППУ	СПИУ	Нет	Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной: <ul style="list-style-type: none"> <li>• Согласие на доступ к счету, спецификация API;</li> <li>• Счета, спецификация API</li> </ul>
4	Получение детальной информации о счете, к которому получен доступ	Обмен данными между Сторонним поставщиком и ППУ	СПИУ	Нет	Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной: <ul style="list-style-type: none"> <li>• Согласие на доступ к счету, спецификация API;</li> <li>• Счета, спецификация API</li> </ul>
5	Получение баланса по счету, к которому получен доступ	Обмен данными между Сторонним поставщиком и ППУ	СПИУ	Нет	Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной: <ul style="list-style-type: none"> <li>• Согласие на доступ к счету, спецификация API;</li> <li>• Баланс, спецификация APIs</li> </ul>
6	Получение списка транзакций по счету, к которому получен доступ	Обмен данными между Сторонним поставщиком и ППУ	СПИУ	Нет	Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной: <ul style="list-style-type: none"> <li>• Согласие на доступ к счету, спецификация APIs;</li> <li>• Транзакции, спецификация API</li> </ul>

Таблица 3. Базовые сценарии использования Открытых банковских интерфейсов

Также Открытые банковские интерфейсы поддерживают технические сценарии использования согласно RESTful API – подходу, которые могут не использоваться в рамках вышеупомянутых сценариев использования. Например, получение деталей объекта согласия или других созданных ресурсов. Детали технических сценариев использования подробно описаны в документах:

- Открытые банковские интерфейсы. Получение информации о счете клиента третьей стороной;
- Открытые банковские интерфейсы. Инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации.

## 8.1. Инициирование одиночного перевода денежных средств

Этот сценарий используется для инициирования одиночной платежной услуги в форме перевода денежных средств по инициативе Плательщика.

Если Стороннего поставщика невозможно идентифицировать корректно через Открытые банковские интерфейсы, то ППУ отклонит запрос.

Если у Стороннего поставщика нет роли СППУ, то ППУ отклонит запрос.

На следующей схеме показаны высокоуровневые информационные потоки для инициирования одиночного перевода денежных средств:



Рисунок 5. Сценарий использования “Инициирование одиночного перевода денежных средств”

## 8.2. Создание долгосрочного согласия на получение информации о счете

Сторонний поставщик выполняет этот сценарий использования, чтобы получить доступ на получение данных в соответствии с другими сценариями использования доступными для роли СПИУ.

Если Стороннего поставщика невозможно идентифицировать корректно через Открытые банковские интерфейсы, то ППУ отклонит запрос.

Если у Стороннего поставщика нет роли СПИУ, то ППУ отклонит запрос.

С согласия Пользователя СПИУ может получить доступ к следующим сценариям:

- Получение списка счетов;
- Получение детальной информации о счете;
- Получение баланса по счету;
- Получение списка транзакций по счету.



Рисунок 6. Сценарий использования “Создание долгосрочного согласия на получение информации о счете”

### 8.3. Получение списка счетов

В этом сценарии использования Сторонний поставщик получает с ранее предоставленного долгосрочного согласия Пользователя список счетов Пользователя.

Если Стороннего поставщика невозможно идентифицировать корректно через Открытые банковские интерфейсы, то ППУ отклонит запрос.

Если у Стороннего поставщика нет роли СПИУ, то ППУ отклонит запрос.

На следующей схеме показаны высокоуровневые информационные потоки для получения списка счетов:

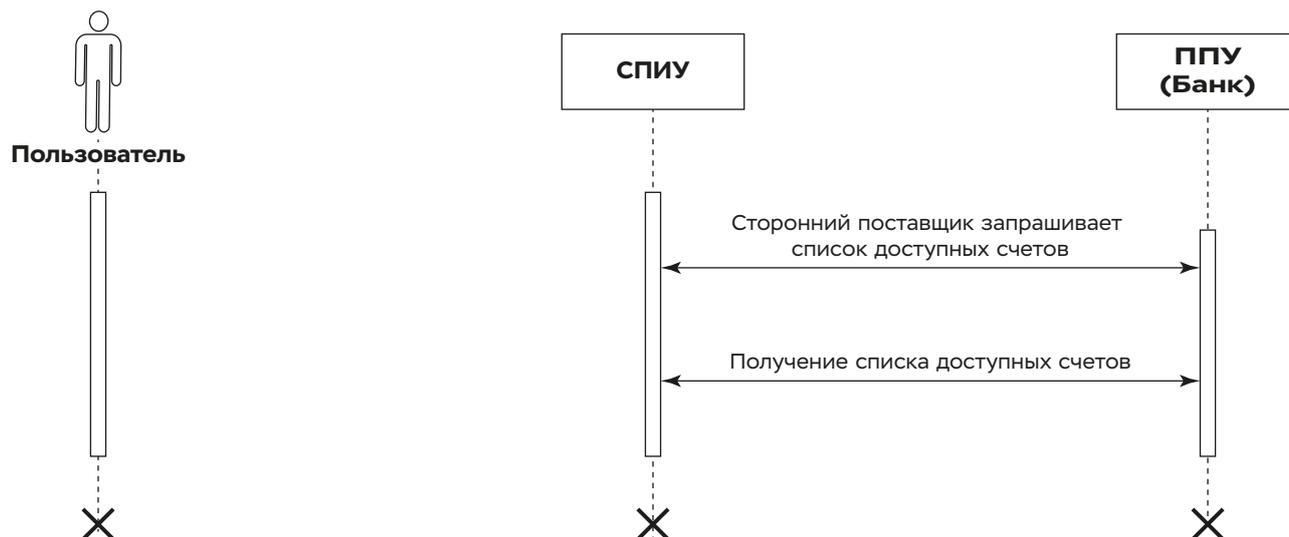


Рисунок 7. Сценарий использования "Получение списка счетов"

### 8.4. Получение детальной информации о счете

В этом сценарии использования Сторонний поставщик получает с ранее предоставленного долгосрочного согласия Пользователя детали счета Пользователя.

Если Стороннего поставщика невозможно идентифицировать корректно через Открытые банковские интерфейсы, то ППУ отклонит запрос.

Если у Стороннего поставщика нет роли СПИУ, то ППУ отклонит запрос.

На следующей схеме показаны высокоуровневые информационные потоки для получения детальной информации о счете:



Рисунок 8. Сценарий использования "Получение детальной информации о счете"

## 8.5. Получение баланса по счету

В этом сценарии использования Сторонний поставщик получает с ранее предоставленного долгосрочного согласия Пользователя баланс счета Пользователя.

Если Стороннего поставщика невозможно идентифицировать корректно через Открытые банковские интерфейсы, то ППУ отклонит запрос.

Если у Стороннего поставщика нет роли СПИУ, то ППУ отклонит запрос.

На следующей схеме показаны высокоуровневые информационные потоки для получения баланса по счету:



Рисунок 9. Сценарий использования "Получение баланса по счету"

## 8.6. Получение списка транзакций по счету

В этом сценарии использования Сторонний поставщик получает с ранее предоставленного долгосрочного согласия Пользователя транзакции по счету Пользователя.

Если Стороннего поставщика невозможно идентифицировать корректно через Открытые банковские интерфейсы, то ППУ отклонит запрос.

Если у Стороннего поставщика нет роли СПИУ, то ППУ отклонит запрос.

На следующей схеме показаны высокоуровневые информационные потоки для получения списка транзакций по счету:

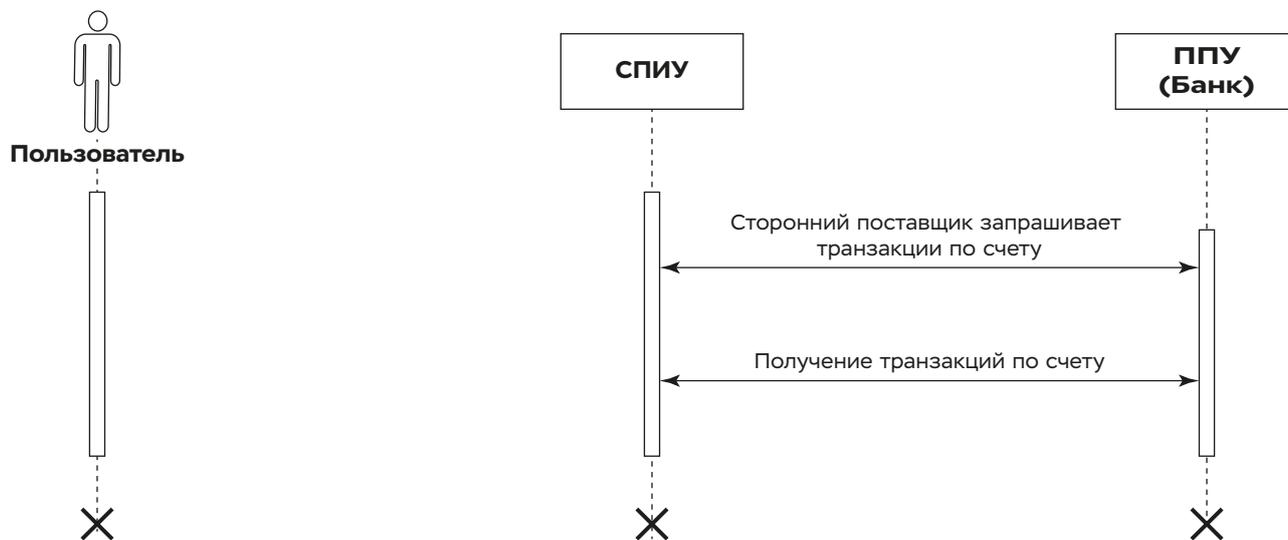


Рисунок 10. Сценарий использования "Получение списка транзакций по счету"



**СТАНДАРТ БАНКА РОССИИ**

## **ОТКРЫТЫЕ БАНКОВСКИЕ ИНТЕРФЕЙСЫ**

**Инициирование перевода денежных средств клиента  
третьей стороной в валюте Российской Федерации**

**Дата введения: 2020-10-23**

**Москва  
2020**

# Оглавление

<b>1. История изменений</b>	<b>22</b>
<b>2. Введение</b>	<b>23</b>
2.1. Область применения	23
2.2. Термины и определения	23
2.3. Принципы архитектуры	24
2.3.1. Идентификаторы	24
2.3.2. Коды статусов	24
<b>3. Основы</b>	<b>25</b>
3.1. Роли участников	25
3.2. Кодировка символов	25
3.3. Формат даты	25
3.4. Структура пути URI-ресурса	26
3.5. Заголовки сообщений (headers)	27
3.5.1. Заголовки запросов	27
3.5.2. Заголовки ответов	29
3.6. Коды статусов HTTP	29
3.6.1. 400 (Bad Request) или 404 (Not Found)	30
3.6.2. 403 (Forbidden)	31
3.6.3. 401 (Unauthorized)	31
3.6.4. 429 (Too Many Requests)	31
3.7. Идемпотентность	31
3.8. Фильтрация	32
3.9. Нумерация страниц	32
3.10. Архивирование	32
3.11. Дополнительные данные	33
<b>4. Модель данных</b>	<b>34</b>
4.1. Справочники и перечисления	34
4.2. Общая структура полезной нагрузки	34
4.2.1. Структура запроса	34
4.2.2. Структура ответа	35
4.2.3. Структура ответа с ошибками	35
4.2.4. Необязательные поля	36
4.2.5. Ссылки	37
4.2.6. Метаданные	37
<b>5. Примеры использования</b>	<b>38</b>
5.1. Потоки с нумерацией страниц	38
<b>6. Инициирование переводов денежных средств, спецификация API – v1.2.1</b>	<b>39</b>
6.1. Введение	39
6.1.1. Принципы проектирования	39
6.2. Основы	39
6.2.1. Общее описание процесса	39
6.2.2. Ограничения при инициировании платежей	42
6.2.3. Сохранение обратной совместимости между мажорными версиями	42
6.3. Конечные точки	43
6.4. Безопасность и контроль доступа	43
6.4.1. Scopes	43
6.4.2. Grants Types	44

6.4.3. Авторизация согласия . . . . .	44
6.4.4. Информация по оценке рисков . . . . .	45
6.5. Модель данных . . . . .	45
6.5.1. Повторно используемые объекты . . . . .	45
6.5.2. Поля для идентификаций . . . . .	51
6.5.3. Виды платежей . . . . .	52
6.5.4. Справочники . . . . .	52
6.6. Внутренние переводы денежных средств, спецификация API – v1.2.1 . . . . .	54
6.6.1. Конечные точки . . . . .	54
6.6.2. Модель данных . . . . .	59
6.6.3. Примеры использования . . . . .	73
<b>7. Динамические справочники в пространствах имен – v1.0.0. . . . .</b>	<b>86</b>
7.1. Введение . . . . .	86
7.2. Основы . . . . .	86
7.2.1. Принципы проектирования . . . . .	86
7.2.2. Управление версиями . . . . .	86
7.3. Общие справочники . . . . .	86

## 1. История изменений

Версия	Дата	Автор	Комментарий
1.0.0	21.08.2019	АФТ, Направление открытых API	Создана первая версия документа
1.0.1	27.11.2019	АФТ, Направление открытых API	Внесены минорные изменения
1.0.2	10.02.2020	АФТ, Направление открытых API	Внесены минорные изменения по замечаниям Центрального банка Российской Федерации
1.1.0	08.04.2020	АФТ, Направление открытых API	Внесены изменения по замечаниям Банка России и участников Ассоциации развития финансовых технологий. Для повторно используемого класса "Initiation" добавлены разделы: <ul style="list-style-type: none"><li>• CreditorParty;</li><li>• CreditorAgent</li></ul>
1.2.1	17.06.2020	АФТ, Направление открытых API	Внесены изменения по замечаниям рабочей группы ТК122 ПКЗ. Добавлен динамичный справочник "OBRUErrorResponseErrorCode" для классификации низкоуровневых ошибок

## 2. Введение

Настоящий Стандарт содержит описание элементов, которые являются общими для всех API на инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации.

### 2.1. Область применения

Настоящий Стандарт рекомендован к использованию организациями при обмене финансовыми сообщениями, связанными с переводом денежных средств в валюте Российской Федерации.

Настоящий Стандарт предназначен для:

- участников получения информации о банковском счете (банки и их клиенты, а также Сторонние поставщики<sup>1</sup>);
- участников перевода денежных средств (банки и их клиенты, а также Сторонние поставщики<sup>2</sup>);
- разработчиков информационного и программного обеспечения.

Положения настоящего Стандарта применяются на добровольной основе.

По предложениям участников настоящий Стандарт может дополняться ролями и сценариями, принятыми в международной практике.

### 2.2. Термины и определения

В настоящем Стандарте применяются следующие термины и определения:

Наименование	Описание
API (Application Programming Interface)	Набор процедур, протоколов и инструментов для создания программных приложений. API определяет, как взаимодействуют программные компоненты
Открытые банковские интерфейсы	Общедоступные интерфейсы прикладного программирования (API), которые предоставляют разработчикам программный доступ к финансовым данным в финансовых сервисах
Пользователь	Физическое или юридическое лицо, являющееся плательщиком или получателем средств
Сторонний поставщик	Юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ)
ППУ	Кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы
СПИУ	Юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя
СППУ	Юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств
Среда Открытых банковских интерфейсов	Комплекс стандартов Открытых банковских интерфейсов, управление, системы, процессы, безопасность и процедуры, используемые для поддержки участников
Участники среды Открытых банковских интерфейсов	Пользователи, банки, поставщики финансовых услуг и разработчики программного обеспечения, которые участвуют в создании и развитии среды Открытых банковских интерфейсов
Плательщик	Пользователь, осуществляющий перевод денежных средств (либо от имени которого осуществляется перевод денежных средств)
Получатель средств	Пользователь, в пользу которого осуществляется перевод денежных средств
Стандарт ISO 20022	Международный стандарт обмена электронными сообщениями между организациями финансовой отрасли

<sup>1</sup> Получают доступ к информации с согласия владельца счета.

<sup>2</sup> Получают доступ к счету с согласия владельца счета.

Наименование	Описание
Многофакторная аутентификация Пользователя	Многофакторная аутентификация Пользователя — это аутентификация, которая основана на использовании двух или более элементов, классифицированных как знания, владение и неотъемлемость. Эти пункты являются независимыми, поскольку нарушение одного не угрожает надежности других
Идемпотентность	Идемпотентность — это свойство объекта или операции при повторном применении операции к объекту давать тот же результат, что и при первом
Ресурс	Ресурсом является представление любой сущности (например, платеж, счет, транзакция) в определенном формате (например, JSON). Каждый ресурс идентифицируется посредством постоянного идентификатора, который не меняется при изменении состояния ресурса
Полезная нагрузка	Часть пакета данных (сообщения) без служебной информации (без заголовка, битов синхронизации и т.п.). Детальное описание структуры полезной нагрузки находится в разделе 4.2

Таблица 1. Термины и определения

## 2.3. Принципы архитектуры

Архитектура среды Открытых банковских интерфейсов соответствует концепции RESTful API<sup>3</sup>.

Данная концепция была выбрана на основании отзывов участников рынка, а также согласно опыту мировых практик.

Однако приоритет заключается в том, чтобы API были просты для понимания и просты в использовании. В случаях, когда следование принципам RESTful является сложным, принципы не соблюдаются.

### 2.3.1. Идентификаторы

Ресурс REST имеет уникальный идентификатор, который используется для идентификации ресурса. Эти уникальные идентификаторы используются для создания URL-адресов для идентификации и адресации конкретных ресурсов.

### 2.3.2. Коды статусов

API используют два кода состояния, которые служат двум различным целям:

- Код состояния HTTP отражает результат вызова API (операция HTTP на ресурсе).
- Поле состояния в некоторых полезных нагрузках ресурса отражает состояние ресурсов.

<sup>3</sup> Детальное описание подхода можно посмотреть на сайте: <https://restfulapi.net/>.

## 3. Основы

### 3.1. Роли участников

Среда Открытых банковских интерфейсов определяет следующих участников:

**Пользователь** — физическое или юридическое лицо, являющееся плательщиком или получателем средств.

Сторонний поставщик — юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ).

**ППУ** (Поставщик Платежных Услуг) — кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы.

Среда Открытых банковских интерфейсов определяет следующие роли участников:

**СПИУ** (Сторонний Поставщик Информационных Услуг о банковском счете) — юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя.

**СППУ** (Сторонний Поставщик Платежных Услуг) — юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств.

Среда Открытых банковских интерфейсов определяет следующие взаимоотношения между участниками и ролями:

Роль \ Участник	Пользователь	ППУ	Сторонний поставщик
<b>СПИУ</b>	Нет	Да	Да
<b>СППУ</b>	Нет	Да	Да

Таблица 2. Роли участников

Открытые банковские интерфейсы регламентируют взаимодействие только между следующими участниками:

- **ППУ** предоставляет Открытые банковские интерфейсы для Стороннего поставщика. Получает сообщения запросов через свои Открытые банковские интерфейсы и отправляет соответствующие ответные сообщения Стороннему поставщику.
- **Сторонний поставщик** может получить доступ к счету Пользователя, управляемому ППУ через Открытые банковские интерфейсы, при согласии Пользователя. Сторонний поставщик отправляет сообщения запроса через Открытые банковские интерфейсы ППУ и получает соответствующие ответные сообщения от этого ППУ.

### 3.2. Кодировка символов

Запросы и ответы API используют кодировку UTF-8. Это кодировка символов по умолчанию для JSON (RFC 7158 — раздел 8.1).

Однако нисходящая система ППУ может не принимать некоторые символы UTF-8, такие как символы emoji (например, идеограммы и смайлики могут не быть приемлемой ссылкой на платеж). Если ППУ отклоняет сообщение с символом UTF-8, которое не может быть обработано, то ППУ отвечает кодом состояния HTTP 400 (неверный запрос).

### 3.3. Формат даты

ППУ принимает в запросах все действующие форматы даты стандарта ISO 8601, включая его разрешенные вариации.

Все даты в полезных нагрузках JSON представлены в формате dateTime стандарта ISO 8601. Все поля dateTime в ответах включают часовой пояс. Например:

```
2019-07-08T11:23:03+00:00
2019-05-03T18:13:23Z
```

Таблица 3. Поля dateTime с часовым поясом

Все даты в параметрах query имеют формат dateTime стандарта ISO 8601 и не включают часовой пояс. Например:

```
2019-07-01T09:23:01
2019-02-07
```

Таблица 4. Поля dateTime с обрезанием часового пояса

Все даты в заголовках HTTP представлены как полные даты RFC 7231. Пример:

```
Mon, 26 Aug 2019 14:23:51 GMT
```

Таблица 5. Формат представления полной даты

Все даты в параметрах claims JWT имеют формат number JSON, представляющий количество секунд с 1970-01-01T0:0:0Z, измеренное в GMT до текущей даты (dateTime).

```
//Mon, 26 Aug 2019 14:23:51 GMT
1987646700
```

Таблица 6. Формат представления полной даты с секундами

### 3.4. Структура пути URI-ресурса

Путь URI соответствует следующей структуре:

- [participant-path-prefix]/open-banking/[version]/[resource]/[resource-id]/[sub-resource]

Структура URI-пути состоит из следующих элементов:

- [participant-path-prefix]  
Необязательный префикс ППУ.
- open-banking  
Постоянное значение “open-banking”.
- [version]  
Версия API, выраженная в виде /v[major-version].[minor-version]/.
- [resource]/[resource-id]  
Наименование ресурса и его идентификатор.
- [sub-resource]  
Наименование подресурса (ресурса 2-го уровня).  
ППУ использует один и тот же participant-path-prefix и host name для всех своих ресурсов.

Примеры:

```
https://bank.ru/oapi-channel/open-banking/v1.1/payments
https://bank.ru/oapi-channel/open-banking/v1.1/account-consents
https://bank.ru/oapi-channel/open-banking/v1.1/accounts
https://bank.ru/oapi-channel/open-banking/v1.1/accounts/1234
https://bank.ru/oapi-channel/open-banking/v1.1/accounts/1234/transactions
```

## 3.5. Заголовки сообщений (headers)

### 3.5.1. Заголовки запросов

Параметр header	Комментарий	POST-запрос	GET-запрос	DELETE-запрос	PUT-запрос
x-fapi-auth-date	Время последнего входа Пользователя в систему с TPP. Значение предоставляется в виде HTTP-date, как в разделе 7.1.1.1 [RFC7231]. Например, x-fapi-auth-date: Mon, 26 Aug 2019 12:23:11 GMT	Необязательно	Необязательно	Необязательно	Не используется
x-fapi-customer-ip-address	IP-адрес Пользователя, если Пользователь в данный момент подключен к Стороннему Поставщику (залогинен в приложении Стороннего Поставщика)	Необязательно	Необязательно	Необязательно	Не используется
x-fapi-interaction-id	RFC4122 UID, используемый в качестве идентификатора корреляции.  Если необходимо, то ППУ передает обратно значение идентификатора корреляции в заголовке ответа x-fapi-interaction-id	Обязательно	Обязательно	Обязательно	Обязательно
Authorization	Стандартный заголовок HTTP. Позволяет предоставлять учетные данные серверу авторизации и (или) серверу ресурсов в зависимости от типа запрашиваемого ресурса. Для OAuth 2.0 / OIDC включает в себя Basic или Bearer схемы аутентификации	Обязательно	Обязательно	Обязательно	Обязательно
Content-Type	Стандартный заголовок HTTP. Представляет формат полезной нагрузки в запросе.  Устанавливается значение application/json, за исключением конечных точек, которые поддерживают Content-Type, отличный от application/json (например, POST /file-payment-consents/{consentId}/file).  Устанавливается значение application/jose+jwe для зашифрованных запросов.  Сторонний Поставщик может предоставлять дополнительную информацию.  Если установлено другое значение, то ППУ присылает ответ: 415 Unsupported Media Type	Обязательно	Не используется	Не используется	Обязательно
Accept	Стандартный HTTP-заголовок, определяющий тип контента, который требуется от сервера.	Необязательно	Необязательно	Не используется	Необязательно

Параметр header	Комментарий	POST-запрос	GET-запрос	DELETE-запрос	PUT-запрос
	<p>Если Сторонний Поставщик ожидает незашифрованный ответ, то он указывает явно, что только ответ в формате JSON принимается (передавая значение application/json) в качестве заголовка контента для всех конечных точек, которые отвечают в формате JSON.</p> <p>Если Сторонний Поставщик ожидает зашифрованный ответ, то он указывает явно, что принимается только ответ JWT (передавая значение application/jose+jwe) в качестве заголовка контента для всех конечных точек, которые отвечают JSON.</p> <p>Для конечных точек, которые не отвечают в формате JSON (например, GET ../statements/{statementId}/file), ППУ указывает доступные параметры на своем портале для разработчиков. Сторонний Поставщик может предоставлять дополнительную информацию.</p> <p>Если установлено недопустимое значение, то ППУ отвечает: 406 (Not Acceptable).</p> <p>Если значение не указано, по умолчанию используется application/json</p>				
x-idempotency-key	<p>Нестандартный HTTP-заголовок. Уникальный идентификатор запроса для поддержки идемпотентности. Обязательно для запросов POST к конечным точкам идемпотентного ресурса.</p> <p>Для других запросов не указывается</p>	Необязательно	Не используется	Не используется	Не используется
x-jws-signature	<p>Указывает, что тело запроса подписано.</p> <p>В документации на ресурсы отдельно определяется, когда это поле в заголовке указывается</p>	Условно (зависит от API)	Условно (зависит от API)	Условно (зависит от API)	Обязательно
x-customer-user-agent	<p>В заголовке указывается тип устройства (user-agent), который использует Пользователь.</p> <p>Сторонний Поставщик может заполнить это поле значением типа устройства (user-agent), указанным Пользователем.</p> <p>Если Пользователь использует мобильное приложение Стороннего Поставщика, Сторонний Поставщик проверяет, что строка типа устройства (user-agent) отличается от строки типа устройства (user-agent) на основе браузера</p>	Необязательно	Необязательно	Необязательно	Необязательно

Таблица 7. Заголовки запросов

Наличие или отсутствие Пользователя определяется через заголовок `x-fapi-customer-ip-address`. Если указан IP-адрес Пользователя, то предполагается, что Пользователь присутствует во время взаимодействия.

Последствием этого является, что ППУ полагаются на информацию, предоставленную СППУ.

### 3.5.2. Заголовки ответов

Параметр header	Комментарий	Обязательность
Content-Type	Стандартный параметр заголовка HTTP. Представляет формат полезной нагрузки, возвращаемой в ответе. ППУ возвращает значение Content-Type, равное <code>application/json</code> , в качестве заголовка для всех незашифрованных конечных точек. ППУ возвращает значение Content-Type, равное <code>application/jwe</code> , для всех зашифрованных конечных точек	Обязательно
Retry-After	Параметр заголовка, указывающий время (в секундах), в течение которого Сторонний Поставщик ожидает перед повторением операции. ППУ следует включать этот заголовок вместе с ответами с кодом состояния HTTP 429 (Too Many Requests)	Необязательно
x-fapi-interaction-id	RFC4122 UID, используемый в качестве идентификатора корреляции. ППУ заполняет параметр заголовка ответа <code>x-fapi-interaction-id</code> значением полученным в соответствующем параметре заголовка запроса или значением UID RFC4122, если значение не было предоставлено в запросе для отслеживания взаимодействия	Обязательно
x-jws-signature	Указывает, что тело ответа подписано. В документации на ресурсы отдельно определяется, когда указывается это поле в заголовке	Условно (зависит от API)

Таблица 8. Заголовки ответов

## 3.6. Коды статусов HTTP

Ниже приведены коды ответов HTTP для различных методов HTTP, для всех конечных точек API.

Ситуация	HTTP-статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
Запрос успешно выполнен	200 OK		нет	да	нет	да
Операция создания выполнена успешно	201 Created	Результатом операции является создание нового ресурса	да	нет	нет	нет
Операция удаления успешно завершена	204 No Content		нет	нет	да	нет
Запрос имеет неверные отсутствующее или несовместимое тело JSON, параметры URL или поля заголовка	400 Bad Request	Запрошенная операция не будет выполнена	да	да	да	да
Заголовок авторизации отсутствует или неверный токен	401 Unauthorized	Операции было отказано в доступе. Повторная аутентификация Пользователя может привести к созданию соответствующего токена, который может быть использован	да	да	да	да
Токен имеет неверную область действия или была нарушена политика безопасности	403 Forbidden	Операции было отказано в доступе. Повторная аутентификация Пользователя может привести к созданию соответствующего токена, который может быть использован	да	да	да	да
1. Сторонний Поставщик пытается получить ресурс, который указан в спецификации, но не реализован на стороне ППУ (например, ППУ решил не реализовывать конечную точку API-статуса для внутренних запланированных платежей)	404 (Not Found)		да	да	да	да

Ситуация	HTTP-статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
2. Сторонний Поставщик пытается получить ресурс, который не определен						
Сторонний Поставщик попытался получить доступ к ресурсу с помощью метода, который не поддерживается	405 Method Not Allowed		да	да	да	да
Запрос содержал параметр заголовка Ассерта, отличный от разрешенных media types, и набор символов, отличный от UTF-8	406 Not Acceptable		да	да	да	да
Операция была отклонена, поскольку полезная нагрузка находится в формате, не поддерживаемом этим методом на целевом ресурсе	415 Unsupported Media Type		да	нет	нет	да
Операция была отклонена, так как слишком много запросов было сделано в течение определенного периода времени	429 Too Many Requests	ППУ ограничивают запросы, если они сделаны сверх их политики добросовестного использования. ППУ документируют свои политики добросовестного использования на своих порталах для разработчиков. ППУ отвечают этим статусом, если количество запросов в единицу времени было превышено. ППУ следует включать заголовок Retry-After в ответ, указывающий, как долго Сторонний Поставщик ожидает перед повторением операции	да	да	да	да
Что-то пошло не так на стороне ППУ	500 Internal Server Error	Операция не удалась	да	да	да	да
Устаревшая версия сервиса	503 Service Unavailable	Если API устарел и больше не поддерживается ППУ, его путь URI все еще может быть активным и принимать запросы. В этом контексте рекомендуется вернуть 503 Service Unavailable, чтобы TPP знал, что версия API находится в offline-режиме	да	да	да	да

Таблица 9. Коды статусов HTTP

ППУ возвращают другие стандартные коды состояния HTTP (например, от шлюзов и других периферийных устройств), как описано в RFC7231 — раздел 6.

ППУ отвечают ошибкой в потоке OAuth / OIDC с обязательным выравниванием кодов ошибок, указанными в разделе 3.1.2.6 OpenID Connect Core Specification.

ППУ отвечают на все некорректные запросы общей структурой ошибок Открытых банковских интерфейсов.

### 3.6.1. 400 (Bad Request) или 404 (Not Found)

Если Сторонний Поставщик пытается запросить URL-ресурса с идентификатором ресурса, который не существует, то ППУ отвечает 400 (неверный запрос), а не 404 (не найдено).

Например, если Сторонний Поставщик пытается выполнить запрос GET/payment/22289, где 22289 не является действительным paymentId, ППУ отвечает 400.

Если Сторонний Поставщик пытается получить доступ к URL-адресу ресурса, который не определен этими спецификациями (например, GET /card-accounts), то ППУ отвечает 404 (Not Found).

Если ППУ не реализовал конечную точку API, то он отвечает 404 (не найдено) для запросов к этому URL.

Таблица ниже иллюстрирует некоторые примеры предсказуемого поведения:

Ситуация	Запрос	Ответ
Сторонний Поставщик пытается получить платеж с несуществующим идентификатором paymentId	GET /payments/1001	400 (Bad Request)
Сторонний Поставщик пытается получить ресурс, который указан в спецификации, но не реализован на стороне ППУ. Например, ППУ решил не реализовывать конечную точку API для получения транзакций по счету	GET /accounts/{accountId}/transactions	404 (Not Found)
Сторонний Поставщик пытается получить ресурс, который не определен	GET /bulk	404 (Not Found)

Таблица 10. Возможные ситуации для ответов 400 (Bad Request) и 404 (Not Found)

### 3.6.2. 403 (Forbidden)

Когда Сторонний Поставщик пытается получить доступ к ресурсу, к которому у него нет разрешения, ППУ возвращает 403 (Forbidden) с необязательным телом ответа об ошибке.

Ситуация возможна в следующих случаях:

- Сторонний Поставщик использует токен доступа (access-token), который не имеет соответствующего scope для доступа к запрошенному ресурсу.
- Сторонний Поставщик попытался получить доступ к ресурсу с идентификатором, к которому у него нет доступа. Например, попытка получить доступ к GET /payments/1001, где платежный ресурс с идентификатором 1001 принадлежит другому Стороннему Поставщику.
- Сторонний Поставщик пытается получить доступ к ресурсу транзакции, а у Стороннего Поставщика нет согласия на авторизацию с правами доступа к запрашиваемому ресурсу.
- Сторонний Поставщик пытается получить доступ к ресурсу транзакции, а у Стороннего Поставщика нет согласия на авторизацию для accountId. Например, попытка получить доступ к GET /accounts/2001 или GET /accounts/2001/transactions, когда Пользователь не выбрал accountId 2001 для авторизации.
- Сторонний Поставщик пытается получить доступ к ресурсу, а ППУ решает повторно аутентифицировать Пользователя. ППУ отвечает соответствующим кодом ошибки, чтобы указать, что требуется повторная аутентификация.

### 3.6.3. 401 (Unauthorized)

Когда Сторонний Поставщик использует токен доступа с истекшим сроком, ППУ возвращает 401 (Unauthorized) без какого-либо сообщения об ошибке.

Ситуация возникает, если ППУ завершил срок действия токена доступа по любой из следующих причин:

1. Истек срок действия согласия.
2. Подозрительное использование токена доступа или подозрение в мошенничестве.
3. Плановая реализация многофакторной аутентификации.

Эта ошибка потенциально может быть исправлена, если Пользователь повторно пройдет аутентификацию или аутентифицируется с соответствующими разрешениями.

### 3.6.4. 429 (Too Many Requests)

Если Сторонний Поставщик пытается получить доступ к ресурсу слишком часто, то ППУ может вернуть 429 (Too Many Requests). Это нефункциональное требование, и отдельные ППУ определяют метрику запросов в единицу времени.

Ситуация возникает, когда:

- Сторонний Поставщик решает реализовать функцию “Статус платежа в реальном времени” для своих пользователей и делает это некорректно, опрашивая конечную точку методом GET.
- Сторонний Поставщик решает использовать конечную точку для разового единичного платежа, как если бы она была конечной точкой пакетной оплаты, и отправляет большое количество запросов на оплату в очень короткий промежуток времени, так что это превышает политику использования ППУ.

## 3.7. Идемпотентность

Ключ идемпотентности используется для защиты от создания дубликатов ресурсов при использовании метода POST для конечных точек API.

Если для конечной точки API требуется ключ идемпотентности:

- Параметр заголовка x-idempotency-key содержит не более 40 символов. Если длина поля превышает 40 символов, то ППУ отклоняет запрос с кодом состояния 400 (Bad Request).

- Сторонний Поставщик не меняет тело запроса при использовании одинакового ключа `x-idempotency-key`. Если Сторонний Поставщик изменяет тело запроса, то ППУ не меняет конечный ресурс. ППУ рассматривает это как мошенническое действие.
- ППУ обрабатывает запрос как идемпотентный, если он получил запрос с существующим параметром `x-idempotency-key` от того же Стороннего Поставщика в течение 24 часов.
- ППУ не создает новый ресурс для запроса `POST`, если он определен как идемпотентный запрос.
- ППУ отвечает на запрос текущим статусом ресурса (или статусом, максимально близким к текущему, который можно получить в данный момент времени на существующем онлайн-канале) и кодом статуса `HTTP 201 (Created)`.
- Сторонний Поставщик не использует ключ идемпотентности при опросе состояния ресурсов.
- ППУ использует подпись сообщения вместе с ключом идемпотентности, чтобы гарантировать, что тело запроса не изменилось.

Если ключ идемпотентности не требуется для конечной точки API, но содержится в запросе, то ППУ игнорирует ключ идемпотентности.

### 3.8. Фильтрация

ППУ обеспечивает ограниченную поддержку фильтрации для операций `GET`, которые возвращают множественные записи.

Параметры фильтра всегда разные для конкретного поля (полей) ресурса и следуют правилам/форматам, определенным в справочниках для ресурса.

Для параметров фильтра типа `DateTime` значения соответствуют стандарту ISO 8601. Если значение поля типа `DateTime` содержит часовой пояс, то ППУ игнорирует эту информацию.

Предполагается, что значения фильтра относятся к тому же часовому поясу, что и часовой пояс, в котором поддерживается ресурс.

### 3.9. Нумерация страниц

ППУ предоставляет постраничный ответ для операций `GET`, которые возвращают множественные записи.

В такой ситуации ППУ:

- Если существует следующая страница записей ресурсов, то ППУ предоставляет ссылку на следующую страницу ресурсов в поле `Links.Next` ответа. Отсутствие следующей ссылки будет означать, что текущая страница является последней страницей результатов.
- Если предыдущая страница записей ресурсов существует, то ППУ предоставляет ссылку на предыдущую страницу ресурсов в поле `Links.Prev` ответа. Отсутствие предыдущей ссылки указывает на то, что текущая страница является первой страницей результатов.

Для разбитых на страницы ответов ППУ гарантирует, что количество записей на странице находится в разумных пределах, минимум 25 записей (кроме последней страницы, где больше нет записей) и максимум 1000 записей.

Дополнительно ППУ предоставляет:

- Ссылку на первую страницу результатов в поле `Links.First`.
- Ссылку на последнюю страницу результатов в поле `Links.Last`.
- Общее количество страниц в поле `Meta.TotalPages`.

ППУ включает `self`-ссылку на ресурс в поле `Links.Self`, как описано в разделе “Ссылки”.

Этот стандарт не определяет, каким образом параметры перелистывания страниц передаются ППУ, каждый ППУ может использовать свои собственные механизмы для разбивки ответа.

Если исходный запрос от СППУ включал параметры фильтра, то в ответе возвращаются только те результаты (разбитые на страницы), которые соответствуют фильтру.

### 3.10. Архивирование

Архивация ресурсов будет определяться для ППУ на основе их внутренних требований.

В дополнение стоит заметить, что:

- ППУ удаляют просроченные идентификаторы согласия (`consentId`) только через 24 часа после создания.
- ППУ могут архивировать просроченные идентификаторы согласия.

### 3.11. Дополнительные данные

Ряд ресурсов в спецификации включает в себя раздел для дополнительных данных (Supplementary Data). Данный раздел позволит ППУ принимать или передавать данные, которые не предусмотрены основной структурой ресурса.

Раздел дополнительных данных определяется как пустой объект JSON в данной спецификации.

Если ППУ использует структуру с дополнительными данными (Supplementary Data), то он выкладывает детальное описание структуры у себя на портале для разработчиков.

ППУ не используют структуру Supplementary Data, если добавляемый туда элемент уже существует в текущей версии документа Открытых банковских интерфейсов.

## 4. Модель данных

### 4.1. Справочники и перечисления

Спецификации Открытых банковских интерфейсов содержат поля со справочными данными.

Справочники бывают двух видов:

- фиксированные;
- гибкие.

В случае фиксированных справочников все возможные значения будут задаваться статично в Стандарте Открытых банковских интерфейсов.

В случае гибких справочников начальные возможные значения будут задаваться в Стандарте Открытых банковских интерфейсов, но каждый ППУ может динамично управлять ими, расширяя своими кастомизированными значениями. Кастомизированные значения, которые используют ППУ, могут в будущем включаться в список начальных возможных значений для гибких справочников.

И фиксированные, и гибкие справочники находятся в разделе “Справочники и перечисления”.

### 4.2. Общая структура полезной нагрузки

В этом разделе приводится обзор структуры верхнего уровня для полезных нагрузок Открытых банковских интерфейсов.

Данные, которые содержатся в разделе “Data”, документируются для каждой отдельно взятой конечной точки API.

#### 4.2.1. Структура запроса

Структура верхнего уровня для запросов Открытых банковских интерфейсов имеет следующий вид:

```
{
  "Data": {
    ...
  },
  "Risk": {
    ...
  }
}
```

Таблица 11. Верхнеуровневая структура сообщений для запросов

#### **Data**

Раздел “Data” содержит данные запроса для конкретного запроса API.

Структура этого элемента отличается для каждой конечной точки API.

#### **Risk**

Раздел “Risk” содержит индикаторы риска для конкретного запроса API, предоставленного Сторонним поставщиком.

Индикаторы риска, содержащиеся в этом элементе, могут отличаться для каждой конечной точки API.

#### 4.2.2. Структура ответа

Структура верхнего уровня для ответов Открытых банковских интерфейсов имеет следующий вид:

```
{
  "Data": {
    ...
  },
  "Risk": {
    ...
  },
  "Links": {
    ...
  },
  "Meta": {
    ...
  }
}
```

Таблица 12. Верхнеуровневая структура сообщений для ответов

В соответствии с принципом API RESTful полный ресурс воспроизводится как часть ответа.

В ответ включаются следующие дополнительные разделы высокого уровня:

- Links
- Meta

#### 4.2.3. Структура ответа с ошибками

Структура для ответов с ошибками Открытых банковских интерфейсов имеет следующий вид:

```
{
  "code": "...",
  "id": "...",
  "message": "...",
  "Errors": [
    {
      "errorCode": "...",
      "message": "...",
      "path": "...",
      "url": "..."
    }
  ]
}
```

Таблица 13. Структура сообщений для ответов с ошибками

#### UML-диаграмма

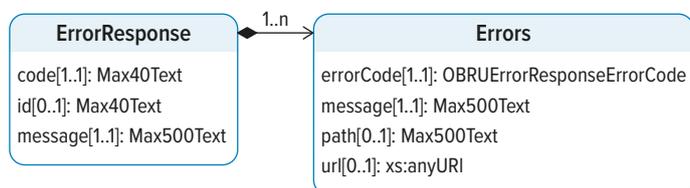


Рисунок 1. UML-диаграмма для структуры ответов с ошибками

## Спецификация данных

Наименование	Кратность	XPath	Подробное описание	Тип данных	Значения	Шаблон
OBRUErrorResponse		OBRUErrorResponse	Массив подробных кодов ошибок, сообщений и URL-адресов к документации для помощи в исправлении	OBRUErrorResponse		
code	1..1	OBRUErrorResponse/Code	Высокоуровневый текстовый код ошибки, необходимый для классификации	Max40Text		
id	0..1	OBRUErrorResponse/Id	Уникальный идентификатор ошибки для целей аудита в случае неизвестных / неклассифицированных ошибок	Max40Text		
message	1..1	OBRUErrorResponse/Message	Краткое сообщение об ошибке. Например, “что-то не так с предоставленными параметрами запроса”	Max500Text		
Errors	1..n	OBRUErrorResponse/Errors		OBRUError		
errorCode	1..1	OBRUErrorResponse/Errors/ErrorCode	Низкоуровневое текстовое описание ошибки. Например, RU.SBRF.Field.Missing	OBRUErrorResponse/ErrorCode		
message	1..1	OBRUErrorResponse/Errors/Message	Описание ошибки. Например, “Обязательное поле не указано”	Max500Text		
path	0..1	OBRUErrorResponse/Errors/Path	Путь к элементу с ошибкой в JSON объекте. Рекомендуемое, но необязательное поле	Max500Text		
url	0..1	OBRUErrorResponse/Errors/Url	URL для помощи в устранении проблемы. Также через URL можно предоставлять дополнительную информацию	xs:anyURI		

Таблица 14. Детальное описание элементов ответа с ошибками

## 4.2.4. Необязательные поля

В объектах, где значение для необязательного поля не указано, поле исключается из полезной нагрузки JSON.

В объектах, где поле массива определено как имеющее значения 0..n, поле массива включается в полезную нагрузку с пустым массивом.

```
{
  "name": "", // Неправильно. Поле "Name" нужно исключить из полезной
  "age": 0, // Неправильно. Значение "0" не используется
  // для указания неопределенного возраста.
  "creditorAccount": {}, // Неправильно. Поле "CreditorAccount" нужно исключить.
  "balances": [] // Правильно. Таким образом передается пустой массив.
}
```

Таблица 15. Примеры передачи необязательных полей

#### 4.2.5. Ссылки

Раздел “Links” является обязательным и всегда будет содержать абсолютные URI для связанных ресурсов.

Ссылка “self” является обязательной.

```
"Links": {  
  "self": "https://api.bank.ru/open-banking/v3.1/payments/58923"  
}
```

Таблица 16. Пример передачи одинарной ссылки “Self”

При передаче большого количества данных раздел Links может также содержать элементы First, Prev, Next и Last.

```
"Links": {  
  "self": "http://rocks.ru/articles?page[number]=3&page[size]=25",  
  "first": "http://rocks.ru/articles?page[number]=1&page[size]=25",  
  "prev": "http://rocks.ru/articles?page[number]=2&page[size]=25",  
  "next": "http://rocks.ru/articles?page[number]=4&page[size]=25",  
  "last": "http://rocks.ru/articles?page[number]=6&page[size]=25"  
}
```

Таблица 17. Пример передачи всех элементов раздела Links

#### 4.2.6. Метаданные

Раздел Meta обязателен, но может быть пустым. Необязательный элемент — “TotalPages”, указывает на количество передаваемых страниц. Если передается более одной страницы, то элемент “TotalPages” обязательно присутствует.

```
"Meta": {  
  "totalPages": 6  
}
```

Таблица 18. Пример передачи раздела Meta

## 5. Примеры использования

Примеры использования для отдельных API задокументированы на соответствующих страницах. В этом разделе приведены примеры использования некоторых шаблонов.

### 5.1. Поток с нумерацией страниц

Приведенный ниже пример иллюстрирует возможность ППУ возвращать многостраничный ответ.

```
GET /accounts/34566/transactions HTTP/1.1
Authorization: Bearer Az567A0Jtyue
x-fapi-auth-date: Mon, 2 Sep 2019 12:33:12 GMT
x-fapi-customer-ip-address: 10.5.412.45
x-fapi-interaction-id: 11bac543-d5de-3446-b687-880a5018434d
Accept: application/json
```

Таблица 19. Запрос списка транзакций

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 11bac543-d5de-3446-b687-880a5018434d
Content-Type: application/json

{
  "Data": {
    ...
  },
  "Links": {
    "self": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions/",
    "last": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions?pg=6",
    "first": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions/",
    "next": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions?pg=2"
  },
  "Meta": {
    "totalPages": 6,
    "firstAvailableDateTime": "2019-05-03T00:00:00+00:00",
    "lastAvailableDateTime": "2019-12-03T00:00:00+00:00"
  }
}
```

Таблица 20. Пример передачи всех элементов раздела Links

## 6. Инициирование переводов денежных средств, спецификация API – v1.2.1

### 6.1. Введение

В этой спецификации инициирования платежей описываются потоки и полезные данные для инициирования платежа на основании распоряжения о переводе денежных средств.

Описанные здесь конечные точки API позволяют СППУ:

- Создать ресурс согласия для инициирования платежа.
- Подтвердить достаточность средств для осуществления платежа (необязательная конечная точка).
- Подтвердить инициирование платежа и отправить его на обработку.
- Получить статус осуществления платежа.
- Получить статус согласия на проведение платежа.

#### 6.1.1. Принципы проектирования

##### 6.1.1.1. Агностичность к способу платежа

Открытые банковские интерфейсы разработаны таким образом, чтобы не зависеть от базовой схемы оплаты, которая определяет порядок осуществления платежа.

Определения полей и их длина установлены согласно стандарту ISO 20022.

##### 6.1.1.2. Коды статусов

Открытые банковские интерфейсы используют следующие виды кодов статуса:

- Код состояния HTTP отражает результат вызова API (операция HTTP на ресурсе). Передается в заголовке сообщения и отображает техническую составляющую процесса.
- Поле “Status” для сообщения с согласием на проведение платежа отражает статус авторизации согласия Пользователем. Передается в сообщении с полезной нагрузкой и отображает бизнес-составляющую процесса.
- Поле “Status” для ресурса платежа отражает статус инициирования или проведения платежа. Передается в сообщении с полезной нагрузкой и отображает бизнес-составляющую процесса.

### 6.2. Основы

В данном разделе описываются потоки данных с позиции бизнес-процессов. Детальное описание потоков данных находится в стандарте “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.

#### 6.2.1. Общее описание процесса

##### 6.2.1.1. Пошаговое описание

Шаг 1: Согласие на инициирование платежа

- Этот поток начинается с согласия Пользователя на совершение платежа. Согласие направляется между Пользователем и СППУ.
- На этом этапе могут быть указаны реквизиты счета Получателя средств.

Шаг 2: Настройка согласия на платеж

- Между СППУ и сервером авторизации ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
- СППУ с помощью потока Client Credentials Grant получает на сервере авторизации ППУ токен доступа (access-token).
- Между СППУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.

- СППУ подключается к ППУ, который обслуживает платежный счет Пользователя и создает новый ресурс согласия на платеж (payment-consent). Это информирует ППУ о том, что один из его Пользователей намерен выполнить платеж. ППУ отвечает идентификатором ресурса согласия на платеж (consentId).
- Этот шаг выполняется с помощью запроса POST к ресурсу согласия на платеж.

#### Шаг 3: Авторизация согласия

- СППУ запрашивает у Пользователя авторизацию согласия. СППУ выполняет этот запрос, используя поток перенаправления (redirection flow):
  - В потоке перенаправления СППУ перенаправляет Пользователя на страницу ППУ.
  - Перенаправление содержит consentId, созданный на предыдущем шаге.
  - Это позволяет ППУ коррелировать согласие на проведение платежа, которое было установлено.
  - ППУ аутентифицирует Пользователя.
  - Пользователь выбирает счет Получателя средств на этом этапе (если счет Получателя средств не был выбран ранее на шаге 1).
  - ППУ обновляет статус ресурса согласия на платеж, фиксируя, что согласие было авторизовано.
  - Как только согласие было авторизовано, Пользователь перенаправляется обратно к СППУ с кодом авторизации (authorization-code).
  - Между СППУ и сервером авторизации ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
  - СППУ обменивается на сервере авторизации ППУ код авторизации (authorization-code) на токен доступа (access-token).

#### Шаг 4: Проведение платежа

- Между СППУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
- СППУ создает ресурс платежа (payment) для указания того, что распоряжение о переводе средств, созданное на предыдущих шагах, направляется на обработку.
- Это выполняется с помощью запроса POST на создание ресурса платежа (payment).
- ППУ возвращает идентификатор ресурса платежа к СППУ (paymentId).

#### Шаг 5: Получение статуса согласия / статуса платежа / деталей платежа

- Между СППУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
- СППУ проверяет статус согласия на проведение платежа (с consentId) или статус ресурса платежа (с paymentId), или детали платежа (с paymentId).
- Это выполняется с помощью запроса GET к ресурсу согласия на проведение платежа (**payment-consent**) или ресурсу платежа (**payment**).

6.2.1.2. Диаграмма последовательности

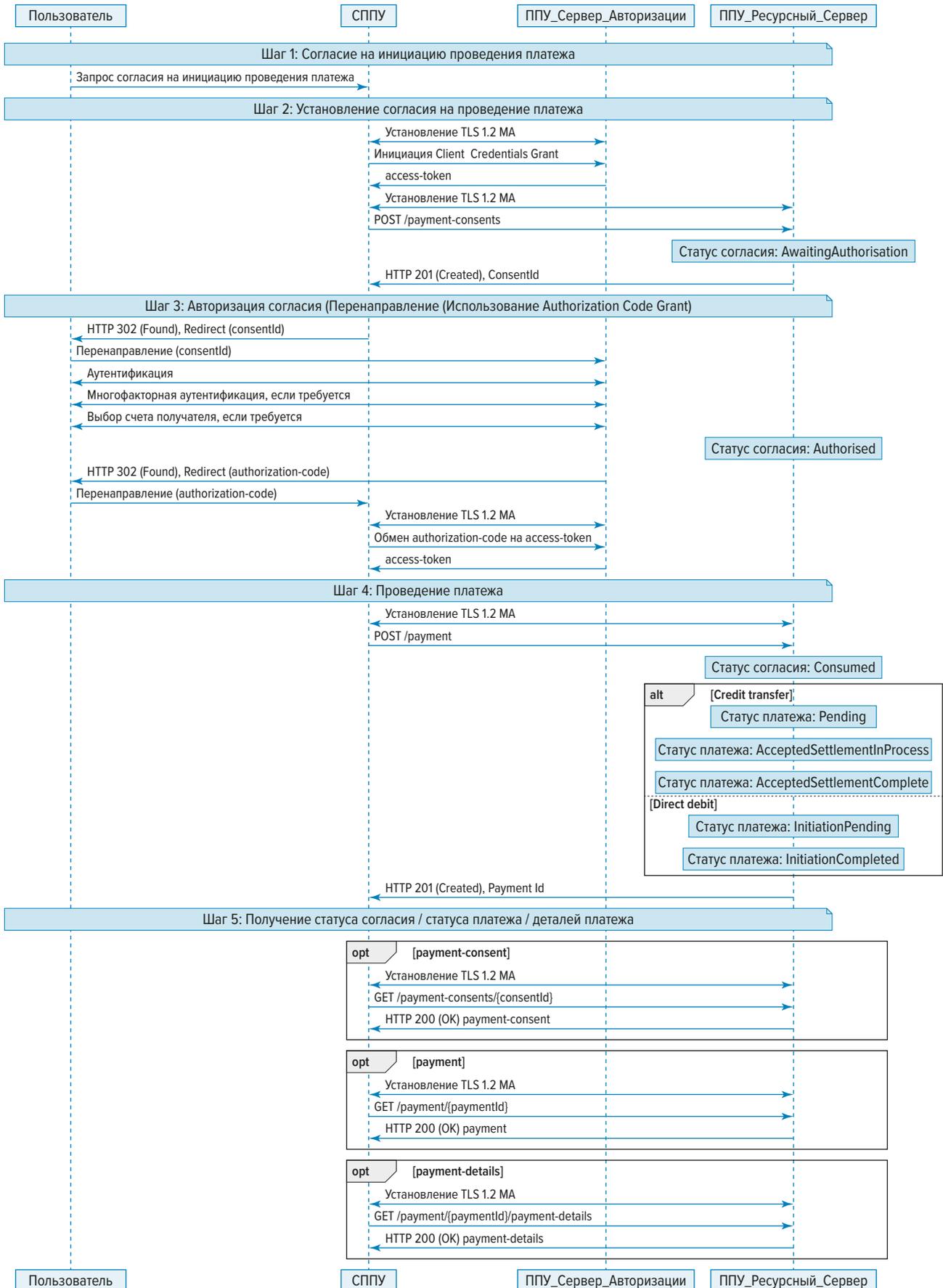


Рисунок 2. Диаграмма последовательности для инициирования платежа

## 6.2.2. Ограничения при инициировании платежей

Стандарт не предусматривает единого набора ограничений для типов платежей, которые могут поддерживаться Открытыми банковскими интерфейсами.

Например:

- Максимальное допустимое значение `InstructedAmount`.
- Максимальное значение даты в будущем для запланированного платежа.

Каждый ППУ определяет соответствующие ограничения, которые они поддерживают, основываясь на их индивидуальных методах, стандартах и ограничениях. Эти ограничения документируются на порталах ППУ для разработчиков.

ППУ отклоняет согласие на платеж, если ППУ не может обработать запрос.

### 6.2.2.1. Использование элемента `CutOffDateTime`

ППУ может вернуть конкретный `CutOffDateTime` при ответе на запрос о согласии на проведение платежа.

ППУ описывает проведение квитанции об оплате до и после `CutOffDateTime` для исполнения распоряжения о переводе средств.

Два варианта развития событий:

- Отклонить распоряжение о переводе средств (и шаги, связанные с созданием данного распоряжения), если оно получено после соответствующего `CutOffDateTime`.
- Принять распоряжение о переводе средств (и шаги, связанные с созданием данного распоряжения), если оно получено после соответствующего `CutOffDateTime`.

#### 6.2.2.1.1. Отклонение распоряжения о переводе средств

В этом случае выполнение платежа явно определено для СППУ и для Пользователя:

- ППУ отклоняет согласие на исполнение распоряжения о переводе средств, если истек `CutOffDateTime` для определенного типа распоряжения.
- ППУ отклоняет запрос авторизации согласия на осуществление платежа, если авторизуемый ресурс согласия связан с истекшим `CutOffDateTime`. ППУ не выдает токен доступа в такой ситуации. ППУ устанавливает статус ресурса согласия на платеж в значение “Отклонено”.
- ППУ отклоняет ресурс платежа, если `CutOffDateTime` для определенного ресурса платежа был установлен и истек.
- СППУ проверяет, что авторизация согласия Пользователя завершена и ресурс платежа создан до того, как истечет `CutOffDateTime`.

#### 6.2.2.1.2. Принятие распоряжения о переводе средств

В этом случае выполнение платежа не является определенным для СППУ и Пользователя и платеж будет выполнен в следующий доступный рабочий день.

- ППУ принимает согласие на проведение платежа, если истек `CutOffDateTime` для определенного типа платежа.
- ППУ принимает запрос авторизации согласия на проведение платежа, когда дата `CutOffDateTime` истекла для ресурса согласия.
- ППУ принимает ресурс платежа, если `CutOffDateTime` для платежа был установлен и истек.
- ППУ обновляет согласие на проведение платежа или ресурс платежа с помощью `CutOffDateTime`, `ExpectedExecutionDateTime` и `ExpectedSettlementDateTime`, чтобы сообщить об ожидаемом поведении при исполнении распоряжения, если `CutOffDateTime` истек.

## 6.2.3. Сохранение обратной совместимости между мажорными версиями

В этом разделе дается обзор принципов сохранения обратной совместимости между версиями Открытых банковских интерфейсов для API-инициирования платежей. Он применяется ко всем ресурсам согласия и ресурсам платежей, указанных в разделе “Конечные точки”.

### 6.2.3.1. Ресурс согласия на проведение платежа

#### 6.2.3.1.1. POST

- СППУ не создает ресурс согласия на проведение платежа с идентификатором `consentId` для более новой версии и не использует его для создания ресурса платежа предыдущей версии.
  - Например, ресурс с идентификатором `consentId`, созданный в версии 2, не используется для создания ресурса платежа версии 1.

- СППУ не создает ресурс согласия на проведение платежа с идентификатором `consentId` для предыдущей версии и не использует его для создания ресурса платежа более новой версии.
  - Например, ресурс с идентификатором `consentId` версии 1 не используется для создания ресурса платежа версии 2.

#### 6.2.3.1.2. GET

- СППУ не получает доступ к ресурсу согласия на проведение платежа с идентификатором `consentId`, созданному в более новой версии, через конечную точку предыдущей версии.
- Поскольку для платежей используется краткосрочное согласие, то ППУ может давать или не давать доступ к ресурсу согласия с идентификатором `consentId` с разных версий конечных точек.

#### 6.2.3.2. Подтверждение наличия денежных средств

##### 6.2.3.2.1. GET

- СППУ не подтверждает наличие денежных средств, используя `consentId` согласия на проведение платежа, созданное в другой версии.
  - Например, ресурс с идентификатором `consentId`, созданный во 2-й версии, не используется для подтверждения денежных средств через конечную точку 1-й версии.

#### 6.2.3.3. Ресурс платежа

##### 6.2.3.3.1. POST

- СППУ использует ресурс согласия на проведение платежа (`consentId`) такой же версии, как и версия ресурса проведения платежа (`paymentId`).
- ППУ не позволяет СППУ использовать `consentId` из предыдущей версии для инициирования платежа в более новой версии, и наоборот.

##### 6.2.3.3.2. GET

- СППУ обращается к онлайн-порталу для разработчиков ППУ для получения рекомендаций по доступности ресурса платежа в более новой версии.
- СППУ не получает доступ к ресурсам платежей более новой версии через конечную точку более старой версии.
- ППУ документирует доступность ресурса платежа в более новой версии на онлайн-портале разработчиков ППУ.
- ППУ разрешает доступ к ресурсу платежа, созданному в предыдущей версии, на конечной точке более новой версии (в зависимости от требований ППУ по хранению данных), подробнее требования будут описаны в следующих версиях спецификации.

## 6.3. Конечные точки

В этом разделе приведен список доступных конечных точек API для выполнения потоков платежей. Если реализована конечная точка POST, конечная точка GET также реализовывается.

Ссылка	Ресурс	Конечная точка
Внутренние переводы денежных средств, спецификация API — v1.2.1	payment-consents payment-consents payments payments payment-details	POST /payment-consents GET /payment-consents/{consentId} POST /payments GET /payments/{paymentId} GET /payments/{paymentId}/payment-details

Таблица 21. Конечные точки для инициирования платежа

## 6.4. Безопасность и контроль доступа

### 6.4.1. Scopes

Токен доступа, требуемый для доступа к платежным API, имеет как минимум следующий scope:

`payments: Возможность проведения платежа`

Таблица 22. Score для инициирования платежа

## 6.4.2. Grants Types

СППУ использует поток `client credentials grant` для получения токена доступа на создание ресурса согласия на проведение платежа (выполнения запроса POST к соответствующей конечной точке). В спецификации этот тип `grant` называется “Client Credentials”.

СППУ использует поток `authorization code grant` (перенаправление или разъединенный поток) для получения токена доступа на создание ресурса платежа (выполнения запроса POST к соответствующей конечной точке). Этот токен также может использоваться для подтверждения наличия денежных средств. В спецификации этот тип `grant` называется “Authorization Code”.

СППУ использует `client credentials grant` для получения токена доступа на выполнение запросов GET (за исключением подтверждения наличия денежных средств).

## 6.4.3. Авторизация согласия

СППУ начинает выполнение платежа с создания ресурса согласия на проведение платежа (`payment-consent`) с помощью операции POST. Этот ресурс определяет разрешения (`permissions`), которые присылает СППУ от имени Пользователя. На начальном этапе согласие не авторизовано, поскольку ППУ еще не верифицировал разрешения во взаимодействии непосредственно с самим Пользователем.

ППУ отвечает сообщением, которое содержит идентификатор ресурса согласия `consentId`. Далее этот идентификатор используется при инициации потока `authorization code grant`, который нужен для подтверждения Пользователем разрешений.

Во время `authorization code grant`:

- ППУ аутентифицирует Пользователя.
- ППУ направляет согласие, полученное от СППУ, обратно Пользователю для непосредственной авторизации (Пользователь подтверждает то, что давал такие разрешения СППУ, но теперь на странице банка). Пользователь может принять или отклонить согласие только целиком (нет выбора параметров).
- Если в согласии не был указан счет Плательщика, то ППУ предоставляет Пользователю список счетов на выбор.

Как только эти шаги выполнены, согласие авторизовано Пользователем.

### 6.4.3.1. Множественная авторизация

В контексте множественной авторизации те же этапы авторизации согласия выполняются для первого Пользователя, чтобы авторизовать или подготовить согласие на проведение платежа.

Для ресурса согласия на проведение платежа:

- СППУ запрашивает `AuthorisationType` для платежа (`Single`, `Any` или `Multiple`). Если значение не указано, то ППУ будет интерпретировать `AuthorisationType` как “Any”.
- СППУ запрашивает `CompletionDateTime` для завершения авторизации платежа. Если значение не указано, то ППУ будет интерпретировать значение `CompletionDateTime` как неограниченное.
- ППУ отклоняет согласие на проведение платежа, если запрашиваемый СППУ тип `AuthorisationType` не совпадает с `DebtorAccount` в запросе.
- ППУ устанавливает статус согласия платежа на `Rejected`, если запрашиваемый СППУ тип `AuthorisationType` не может быть удовлетворен после аутентификации Пользователя:
  - ППУ отвечает ошибкой OAuth, указанной как `invalid_request` и `error_description`, содержащей соответствующее описание.
- ППУ ограничивает выбор `DebtorAccount` (в онлайн-канале ППУ) счетами, которые соответствуют типу `AuthorisationType`, запрошенному СППУ.

Для ресурса платежа:

- ППУ отвечает объектом `MultiAuthorisation`, если распоряжение о переводе средств требует множественной авторизации. Объект `MultiAuthorisation` указывает СППУ, что распоряжение о переводе средств требует множественной авторизации.
- ППУ заполняет поле `Status` объекта `MultiAuthorisation` значениями, соответствующими множественной авторизации.
- ППУ может заполнить объект `MultiAuthorisation` дополнительно необязательными полями для множественной авторизации:
  - Количество требуемых авторизаций (общее количество, которое необходимо с начала множественной авторизации).
  - Количество выполненных авторизаций (завершенных на данный момент).
  - Дата и время последнего обновления авторизации.
  - Дата и время, когда процесс авторизации завершается.

#### 6.4.3.2. Условия возникновения ошибок

Если Пользователь не завершает авторизацию согласия успешно (например, если пользователь не прошел аутентификацию успешно), поток authorization code grant заканчивается перенаправлением на сторону СППУ с ответом об ошибке, как описано в разделе 4.1.2.1 RFC6749. Пользователь перенаправляется на сторону СППУ с параметром ошибки, указывающим на возникшую ситуацию.

#### 6.4.3.3. Отзыв согласия

Пользователь не может отозвать согласие на проведение платежа после его авторизации.

#### 6.4.3.4. Изменения выбранного счета

Для согласия на проведение платежа выбранный счет Плательщика не может быть изменен после того, как согласие было авторизовано.

#### 6.4.3.5. Повторная аутентификация

Согласие на проведение платежа является краткосрочным согласием и не может быть повторно аутентифицировано Пользователем.

### 6.4.4. Информация по оценке рисков

Из мирового опыта была взята практика оценки рисков для платежей, инициированных через API платежей.

Информация по оценке рисков располагается в:

- FAPI HTTP-заголовках. Это определено в разделе 6.3 спецификации FAPI и в разделе заголовков выше.
- Дополнительные поля, определенные отраслью как проблемы безопасности бизнес-логики, будут переданы в разделе “Risk” полезной нагрузки в объекте JSON.

Это набор дополнительных полей в разделе риска полезной нагрузки для v1.0.0, который будет определен на стороне СППУ:

- PaymentContextCode.
- MerchantCategoryCode.
- MerchantCustomerIdentification.
- DeliveryAddress.

PaymentContextCode описывает контекст платежа и может иметь следующие значения:

- BillPayment.
- EcommerceGoods.
- EcommerceServices.
- Other.
- PartyToParty.

В платежах EcommerceGoods и EcommerceServices могут быть заполнены MerchantCategoryCode и MerchantCustomerIdentification (рекомендовано заполнять). В платежах EcommerceGoods также рекомендуется заполнять адрес доставки.

Эти поля описаны в разделе “Модель данных”.

## 6.5. Модель данных

### 6.5.1. Повторно используемые объекты

#### 6.5.1.1. Risk

В этом разделе описывается класс Risk, который используется в ресурсах согласия на проведение платежа и непосредственно платежа.

##### 6.5.1.1.1. Диаграмма UML

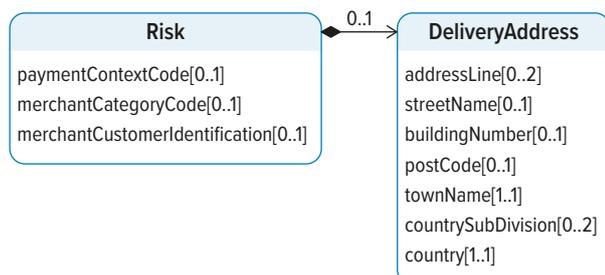


Рисунок 3. Диаграмма объекта Risk

## 6.5.1.1.2. Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Risk		Risk	Раздел "Risk" отправляется инициатором в ППУ. Он используется для указания дополнительных деталей для оценки рисков для платежей	RiskComplexType		
paymentContextCode	0..1	Risk/paymentContextCode	Контекст проведения платежа	PaymentContextStaticType	BillPayment EcommerceGoods EcommerceServices Other PartyToParty	
merchantCategoryCode	0..1	Risk/merchantCategoryCode	Код категории соответствует ISO 18245 и относится к типу услуг или товаров, которые продавец предоставляет для транзакции	Min3Max4Text		
merchantCustomerIdentification	0..1	Risk/merchantCustomerIdentification	Уникальный идентификатор покупателя, который продавец присвоил Пользователю	Max70Text		
DeliveryAddress	0..1	Risk/DeliveryAddress	Информация, которая находит и идентифицирует определенный адрес, как определено почтовыми службами или в произвольном текстовом формате	PostalAddress		
addressLine	0..2	Risk/DeliveryAddress/addressLine[ ]	Информация, которая определяет и идентифицирует определенный адрес, как это определено почтовыми службами, в форме свободного текста	Max70Text		
streetName	0..1	Risk/DeliveryAddress/streetName	Наименование улицы или проспекта	Max70Text		
buildingNumber	0..1	Risk/DeliveryAddress/buildingNumber	Номер, который определяет положение здания на улице	Max16Text		
postCode	0..1	Risk/DeliveryAddress/postCode	Идентификатор, состоящий из группы букв и (или) цифр, которые добавляются к почтовому адресу для упрощения сортировки почты	Max16Text		
townName	1..1	Risk/DeliveryAddress/townName	Наименование застроенного пространства с определенными границами и органами местного самоуправления	Max35Text		
countrySubDivision	0..2	Risk/DeliveryAddress/countrySubDivision[ ]	Определяет часть страны, например, край, область, республика	Max35Text		
country	1..1	Risk/DeliveryAddress/country	Государство	CountryCode		^[A-Z]{2,2}\$

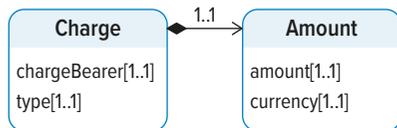
Таблица 23. Состав данных объекта Risk

### 6.5.1.2. Charge

В этом разделе описывается класс Charge, который используется в полезных нагрузках ответа в ресурсах согласия на проведение платежа.

#### 6.5.1.2.1. Диаграмма UML

Рисунок 4. Диаграмма объекта Charge



#### 6.5.1.2.2. Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Charge		Charge	Набор элементов, используемый для предоставления деталей о плате за инициацию платежа. Сколько будет стоить для Пользователя проведение платежа	ChargeComplexType		
chargeBearer	1..1	Charge/chargeBearer	Указывается, какая сторона/стороны будут нести (оплачивать) расходы, связанные с обработкой платежной инструкции	ChargeBearerType1Code	CRED DEBT SLEV SHAR	
type	1..1	Charge/type	Тип комиссии за проведение платежа	ExternalChargeType1Code	BRKF BTCH COMM SUMM	
Amount	1..1	Charge/Amount	Сумма комиссии за проведение платежа	ActiveOrHistoricCurrencyAndAmount		
amount	1..1	Charge/Amount/amount	Сумма комиссии	OBActiveCurrencyAndAmount_SimpleType		^d{1,13}\.d{1,5}\$
currency	1..1	Charge/Amount/currency	Код валюты, соответствующий международному стандарту ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$

Таблица 24. Состав данных объекта Charge

### 6.5.1.3. Authorisation

В этом разделе описывается класс Authorisation, который используется в полезных нагрузках запроса и ответа к ресурсу согласия на проведение платежа.

#### 6.5.1.3.1. Диаграмма UML

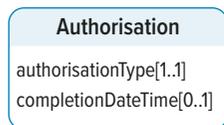


Рисунок 5. Диаграмма объекта Authorisation

### 6.5.1.3.2. Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Authorisation		Authorisation	Запрос типа авторизации от СППУ	AuthorisationComplexType		
authorisationType	1..1	Authorisation/authorisationType	Тип запрашиваемого потока авторизации	AuthorisationStaticType	Any Single Multiple	
completionDateTime	0..1	Authorisation/completionDateTime	Дата и время, когда запрашиваемый поток авторизации завершается	ISODateTime		

Таблица 25. Состав данных объекта Authorisation

### 6.5.1.4. MultiAuthorisation

В этом разделе описывается класс MultiAuthorisation, который используется в полезных нагрузках ответов ресурсов платежей.

#### 6.5.1.4.1. Диаграмма UML

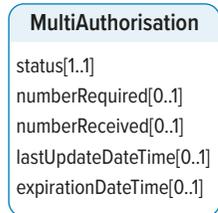


Рисунок 6. Диаграмма объекта MultiAuthorisation

### 6.5.1.4.2. Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
MultiAuthorisation		MultiAuthorisation	Ответ потока множественной авторизации от ППУ	MultiAuthorisationComplexType		
status	1..1	MultiAuthorisation/status	Состояние потока авторизации	MultiAuthorisationStatusStaticType	Authorised AwaitingFurtherAuthorisation Rejected	
numberRequired	0..1	MultiAuthorisation/numberRequired	Количество авторизаций, необходимых для инициирования проведения платежа (общее количество, необходимое при старте мультиавторизации)	Number		
numberReceived	0..1	MultiAuthorisation/numberReceived	Количество полученных авторизаций	Number		
lastUpdateDateTime	0..1	MultiAuthorisation/lastUpdateDateTime	Дата и время последнего обновления авторизационного потока	ISODateTime		
expirationDateTime	0..1	MultiAuthorisation/expirationDateTime	Дата и время, когда поток авторизации завершается	ISODateTime		

Таблица 26. Состав данных объекта MultiAuthorisation

### 6.5.1.5. PaymentDetails

В этом разделе описывается класс PaymentDetails, который используется в полезных нагрузках ответов подресурсов деталей платежа.

#### 6.5.1.5.1. Диаграмма UML

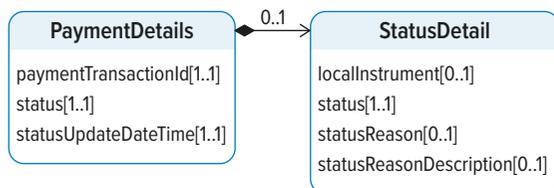


Рисунок 7. Диаграмма объекта PaymentDetails

#### 6.5.1.5.2. Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
PaymentDetails		PaymentDetails	Детали статуса оплаты	PaymentDetailsComplexType		
paymentTransactionId	1..1	PaymentDetails/paymentTransactionId	Уникальный идентификатор транзакции внутри обслуживающего учреждения. Этот идентификатор является уникальным и неизменным	Max210Text		
status	1..1	PaymentDetails/status	Статус платежа, назначенный администратором транзакции	ExternalPaymentTransactionStatus1Code	ACCC ACCP ACFC ACSC ACSP ACTC ACWC ACWP CANC PATC PDNG RCVD RJCT	
statusUpdateDateTime	1..1	PaymentDetails/statusUpdateDateTime	Дата и время, когда статус был присвоен переводу	ISODateTime		
StatusDetail	0..1	PaymentDetails/StatusDetail	Детали статуса оплаты	PaymentStatusDetailComplexType		
localInstrument	0..1	PaymentDetails/StatusDetail/localInstrument	Специальный инструмент, используемый сообществом. Использование: этот элемент используется для указания местного инструмента, варианта локального клиринга и (или) для дополнительного описания сервиса или уровня обслуживания	ExternalLocalInstrument1Code		
status	1..1	PaymentDetails/StatusDetail/status	Статус платежа, назначенный администратором транзакции	Max128Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
statusReason	0..1	PaymentDetails/StatusDetail/statusReason	Код причины статуса платежа	TransactionIndividualStatusReasonStaticType	Cancelled PendingFailingSettlement PendingSettlement Proprietary ProprietaryRejection Suspended Unmatched	
statusReasonDescription	0..1	PaymentDetails/StatusDetail/statusReasonDescription	Детальное описание причины статуса платежа	Max256Text		

Таблица 27. Состав данных объекта PaymentDetails

#### 6.5.1.6. SCASupportData

В этом разделе описывается класс SCASupportData, который используется во всех ресурсах запроса согласия на проведение платежа, что позволяет СППУ предоставлять вспомогательные данные при запросе ППУ для исключения SCA.

##### 6.5.1.6.1. Диаграмма UML



Рисунок 8. Диаграмма объекта SCASupportData

##### 6.5.1.6.2. Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
SCASupportData		SCASupportData	Вспомогательные данные, предоставленные СППУ, при запросе SCA	SCASupportData		
requestedSCAExemptionType	0..1	SCASupportData/requestedSCAExemptionType	Поле позволяет СППУ запрашивать конкретное исключение SCA для инициирования платежа	SCAExemptionTypeStaticType	BillPayment ContactlessTravel EcommerceGoods EcommerceServices Kiosk Parking PartyToParty	
appliedAuthenticationApproach	0..1	SCASupportData/appliedAuthenticationApproach	Поле показывает, подвергался ли Пользователь SCA, выполняемой СППУ	AppliedAuthenticationApproachStaticType	CA SCA	
referencePaymentOrderId	0..1	SCASupportData/referencePaymentOrderId	Использование: если платеж повторяется, то идентификатор транзакции предыдущего вхождения платежа позволяет ППУ убедиться, что СППУ, сумма и получатель платежа совпадают с предыдущим вхождением	Max128Text		

Таблица 28. Состав данных объекта SCASupportData

## 6.5.2. Поля для идентификаций

В этом разделе описываются идентификаторы, используемые в потоках API-платежей, направление потока и как они используются.

Создание	Идентификатор	Бизнес-описание
Продавец/СППУ Отправляется в полезной нагрузке API	endToEndIdentification	Уникальная идентификация, присвоенная иницирующей стороной для однозначной идентификации операции. Эта идентификация передается без изменений по всей цепочке от начала до конца.  Использование: сквозной идентификатор может использоваться для выверки расчетов для связки задач, относящихся к операции. Она может включаться в несколько сообщений, относящихся к операции
Продавец/СППУ Отправляется в полезной нагрузке API	instructionIdentification	СППУ генерирует InstructionIdentification, который является уникальным идентификатором транзакции, и передает его в ППУ (это обязательно), но это не идет дальше в потоке платежей.  Уникальная идентификация, присвоенная инструктирующей стороной для проинструктированной стороны, для однозначной идентификации инструкции.  Использование: идентификация инструкции является сквозным идентификатором, который может использоваться между инструктирующей стороной и проинструктированной стороной для ссылки на индивидуальную инструкцию. Она может включаться в несколько сообщений, относящихся к инструкции.  Ожидается, что это уникально на неопределенный срок для всех периодов времени. СППУ может гарантировать, что это неопределенно уникально, включив в поле элемент даты или даты и времени, или вставив уникальный идентификатор
Продавец/СППУ Отправляется в полезной нагрузке API	remittanceInformation	RemittanceInformation — это информация о платеже, которую согласовывает Получатель средств, для обеспечения квитанции с позициями, которые должен обеспечить перевод, например, оплата по коммерческим счетам, по которым ведется учет в системе ожидаемых поступлений
ППУ / API система	consentId	Уникальная идентификация, назначенная ППУ для однозначного определения ресурса согласия на проведение платежа
ППУ / API система	paymentId	Уникальная идентификация, назначенная ППУ для однозначного определения платежа (ресурса платежа). ● paymentId
ППУ / Способ платежа	scheme Payment ID	Создается ППУ для однозначной идентификации способа платежа при его проведении

Таблица 29. Идентификаторы, используемые в потоках API-платежей

В приведенных ниже таблицах указан участник (эктор), который изначально создает каждый из идентификаторов сообщений, а также их передача и видимость другим участникам.

Эти потоки являются ориентировочными и будут зависеть от того, какие платежные схемы могут поддерживать.

Приняты следующие обозначения:

+ — определяет участника, который создает идентификатор

>> — направление прямого потока

<< — направление обратного потока

### 6.5.2.1. Поток продавца

Идентификатор	Пользователь	Продавец	СППУ	ППУ	Платежная схема	Бенифициар
endToEndIdentification		+	>>	>>	>>	>>
remittanceInformation		+	>>	>>	>>	>>
instructionIdentification			+	>>		
consentId			<<	+		
paymentId			<<	+		
scheme Payment ID				+	>>	>>

Таблица 30. Поток продавца

## 6.5.2.2. Поток между пользователями

Идентификатор	Пользователь	Продавец	СППУ	ППУ	Платежная схема	Бенифициар
endToEndIdentification			+	>>	>>	>>
remittanceInformation	+		>>	>>	>>	>>
instructionIdentification			+	>>		
consentId			<<	+		
payment Order Id			<<	+		
scheme Payment ID				+	>>	>>

Таблица 31. Поток между Пользователями

## 6.5.3. Виды платежей

Каждый из типов платежей задокументирован на подстраницах данной спецификации. Каждый тип платежа в документации содержит:

- Конечные точки
  - Конечные точки API, доступные для ресурса.
- Модель данных
  - Определение ресурса
  - Диаграмму UML
  - Разрешения, связанные с доступом к ресурсу
  - Спецификацию данных, которая определяет поля, повторно используемые классы, обязательные и необязательные поля
- Примеры использования.

## 6.5.4. Справочники

## PaymentContextStaticTypeAcceptedSettlementCompleted

№	Значение	Описание
1	BillPayment	Контекстом инициирования платежа является оплата по счету
2	EcommerceGoods	Контекст инициирования платежа относится к товарам через канал электронной коммерции
3	EcommerceServices	Контекст инициирования платежа относится к услугам через канал электронной коммерции
4	PartyToParty	Контекстом инициирования платежа является перевод между двумя сторонами
5	Other	Контекст инициирования платежа относится к другому типу

Таблица 32. Контекст инициирования платежа

## TransactionStatusStaticType

№	Значение	Описание
1	AcceptedSettlementCompleted	Расчет по счету Плательщика завершен. СППУ не используют этот статус в качестве подтверждения завершения расчетов по счету Получателя средств
2	AcceptedSettlementInProgress	Все предыдущие проверки, такие как техническая проверка и профиль клиента, были успешными, и поэтому запрос об инициировании платежа был принят к исполнению
3	Pending	Ожидается инициирование платежа или отдельной транзакции, включенной в инициирование платежа
4	Rejected	Инициирование платежа или отдельная транзакция, включенная в инициирование платежа, были отклонены
5	AcceptedWithoutPosting	Платежная инструкция, включенная в перевод, принимается без зачисления на счет Получателя средств
6	AcceptedCreditSettlementCompleted	Расчет по счету Получателя средств завершен

Таблица 33. Статус транзакции

### ConsentStatusStaticType

№	Значение	Описание
1	AwaitingAuthorisation	Ресурс согласия на проведение платежа ожидает авторизации пользователя
2	Rejected	Ресурс согласия на проведение платежа был отклонен
3	Authorised	Ресурс согласия на проведение платежа был успешно авторизован
4	Consumed	Ресурс согласия на проведение платежа был успешно задействован

Таблица 34. Статус согласия на проведение платежа

### AuthorisationStaticType

№	Значение	Описание
1	Any	Любой тип авторизации запрашивается
2	Multiple	Требуется несколько типов авторизации
3	Single	Запрашивается один тип авторизации

Таблица 35. Тип авторизации

#### 6.5.4.1. Справочники ISO

Следующие справочники ISO используются в API инициирования платежей.

№	Значение	Описание
1	ActiveOrHistoricCurrencyCode	<a href="https://www.iso20022.org/external_code_list.page">https://www.iso20022.org/external_code_list.page</a>
2	CountryCode	Стандарт ГОСТ 7.67.2003 (ИСО 3166-1:1997), alpha 2 code — <a href="http://docs.cntd.ru/document/1200035671">http://docs.cntd.ru/document/1200035671</a> . Таблица 1, двухзначный буквенный код на латинице
3	Min3Max4Text	<a href="https://www.iso.org/standard/33365.html">https://www.iso.org/standard/33365.html</a>

Таблица 36. Международные и общероссийские справочники

### ChargeBearerType1Code (ISO 20022)

№	Значение	Описание
1	CRED	Все расходы по сделке несет Получатель средств
2	DEBT	Все расходы по сделке несет Плательщик
3	SLEV	Плата взимается в соответствии с правилами, согласованными на уровне обслуживания и (или) схеме
4	SHAR	В контексте перевода денежных средств по инициативе Плательщика означает, что расходы по сделке на стороне отправителя несет Плательщик, а расходы по сделке на стороне получателя несет Получатель средств. В контексте прямого дебета означает, что расходы по сделке на стороне отправителя несет Получатель средств, а расходы по сделке на стороне получателя несет Плательщик

Таблица 37. Расходы по операции

## 6.6. Внутренние переводы денежных средств, спецификация API – v1.2.1

### 6.6.1. Конечные точки

Ресурс	Метод HTTP	Конечная точка	Scope	Grant type	Подписание	Ключ идемпотентности	Объект запроса	Объект ответа
payment-consents	POST	POST /payment-consents	payments	Client Credentials	Подписывается запрос Подписывается ответ	Да	ConsentRequest	ConsentResponse
payment-consents	GET	GET /payment-consents/ {consentId}	payments	Client Credentials	Подписывается ответ	Нет		ConsentResponse
payments	POST	POST /payments	payments	Authorization Code	Подписывается запрос Подписывается ответ	Да	PaymentRequest	PaymentResponse
payments	GET	GET /payments/{paymentId}	payments	Client Credentials	Подписывается ответ	Нет		PaymentResponse

Таблица 38. Конечные точки для ресурса платежа

### 6.6.1.1. POST /payment-consents

POST /payment-consents

Таблица 39. Создание ресурса согласия на проведение платежа

Данная конечная точка позволяет выполнять следующее взаимодействие:

- Конечная точка позволяет с помощью метода POST создавать на стороне ППУ ресурс согласия на проведение платежа (payment-consent) без предварительной идентификации Пользователя.
- Полезная нагрузка запроса может не содержать отдельных необязательных полей, например, она может не содержать информацию о том, какой именно счет Пользователя, открытый в банке (ППУ), будет в итоге использован для списания денежных средств.
- Конечная точка позволяет СППУ отправлять согласие (между Пользователем и СППУ) в качестве информации для ППУ по поддержке процесса авторизации Пользователя.
- Конечная точка позволяет ППУ отправлять СППУ идентификатор созданного ресурса согласия на проведение платежа (consentId).
- Сразу после создания ресурса согласия на проведение платежа ППУ присваивает ему статус по умолчанию "AwaitingAuthorisation".

### 6.6.1.2. GET /payment-consents/{consentId}

GET /payment-consents/{consentId}

Таблица 40. Получение статуса ресурса согласия на проведение платежа

Данная конечная точка позволяет СППУ получать детали и статус созданного им ресурса согласия на проведение платежа по его идентификатору.

#### 6.6.1.2.1. Статус ресурса согласия на проведение платежа

После того как Пользователь авторизовал согласие на проведение платежа, ППУ изменяет статус ресурса на "Authorised".

При отклонении согласия Пользователем ППУ изменяет статус ресурса на "Rejected".

При возникновении иных причин, препятствующих проведению платежа, ППУ изменяет статус ресурса на "Rejected".

После того как ресурс согласия был задействован при выполнении платежа, ППУ изменяет статус ресурса на "Consumed".

Статус ресурса согласия на проведение платежа имеет следующие значения:

№	Статус	Описание
1	Rejected	Согласие на проведение платежа отклонено
2	AwaitingAuthorisation	Согласие на проведение платежа ожидает авторизации
3	Authorised	Согласие на проведение платежа успешно авторизовано
4	Consumed	Согласие на проведение платежа было задействовано и больше не может использоваться при проведении платежа

Таблица 41. Статусы ресурса согласия на проведение платежа

### 6.6.1.3. POST /payments

GET /payments

Таблица 42. Создание ресурса платежа

Конечная точка используется после того, как Пользователь авторизовал ресурс согласия на проведение платежа.

Конечная точка используется для подтверждения инициирования проведения платежа.

Данная конечная точка позволяет выполнять следующее взаимодействие:

- Конечная точка позволяет с помощью метода POST создавать на стороне ППУ ресурс платежа (payment).
- Запрос, выполненный с помощью конечной точки, является инструкцией для ППУ на выполнение платежа.
- СППУ обеспечивает соответствие разделов "Initiation" и "Risk" ресурса платежа соответствующим разделам "Initiation" и "Risk" ресурса согласия на проведение платежа (значения всех присутствующих

и там, и там элементов должны быть равны). Если есть хотя бы одно несовпадение, ППУ отмечает ошибкой с кодом 400 (Bad Request).

- Любые операции с ресурсом платежа не приводят к изменению статуса ресурса платежа.

#### 6.6.1.4. GET /payments/{paymentId}

GET /payments/{paymentId}

Таблица 43. Получение статуса ресурса платежа

Данная конечная точка позволяет СППУ получать статус созданного ресурса платежа по его идентификатору.

##### 6.6.1.4.1. Статус

Статус ресурса платежа может иметь следующие значения:

№	Статус	Наименование статуса	Описание
1	PDNG	Pending	Начальный статус выполнения платежа
2	RJCT	Rejected	Платеж отклонен
3	ACSP	AcceptedSettlementInProgress	Все проверки, такие как техническая проверка и профиль клиента, были успешными, и поэтому инициирование платежа было принято к исполнению
4	ACSC	AcceptedSettlementCompleted	Расчет по счету Плательщика выполнен
5	ACWP	AcceptedWithoutPosting	Платежная инструкция, включенная в перевод, принимается без зачисления на счет Получателя средств
6	ACCC	AcceptedCreditSettlementCompleted	Расчет по счету Получателя средств завершен

Таблица 44. Статусы ресурса платежа

##### 6.6.1.4.2. Модель статусов обработки

##### 6.6.1.4.3. Статус операций для ресурса согласия на проведение платежа

Модель состояний для ресурса согласия на проведение платежа соответствует следующей модели:

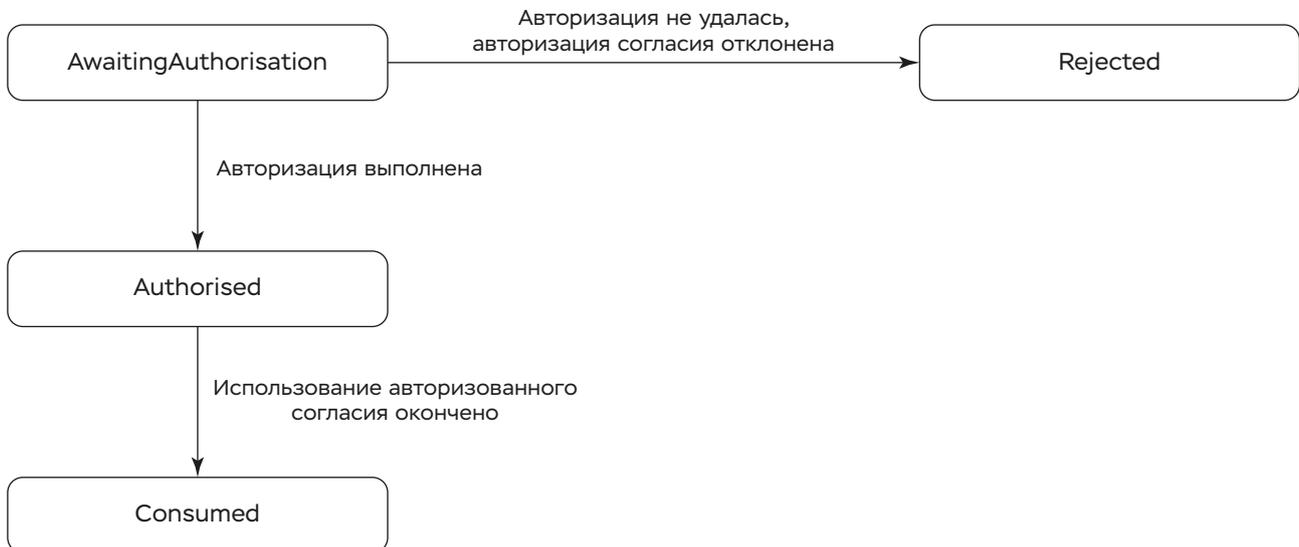


Рисунок 9. Диаграмма состояний ресурса согласия на проведение платежа

Описание статусов:

№	Статус	Описание
1	Rejected	Согласие на проведение платежа отклонено
2	AwaitingAuthorisation	Согласие на проведение платежа ожидает авторизации
3	Authorised	Согласие на проведение платежа успешно авторизовано
4	Consumed	Согласие на проведение платежа было задействовано и больше не может использоваться при проведении платежа

Таблица 45. Статусы ресурса согласия на проведение платежа

#### 6.6.1.4.4. Статус операций для ресурса платежа

Модель состояния для ресурса платежа соответствует статусам из стандарта ISO 20022.

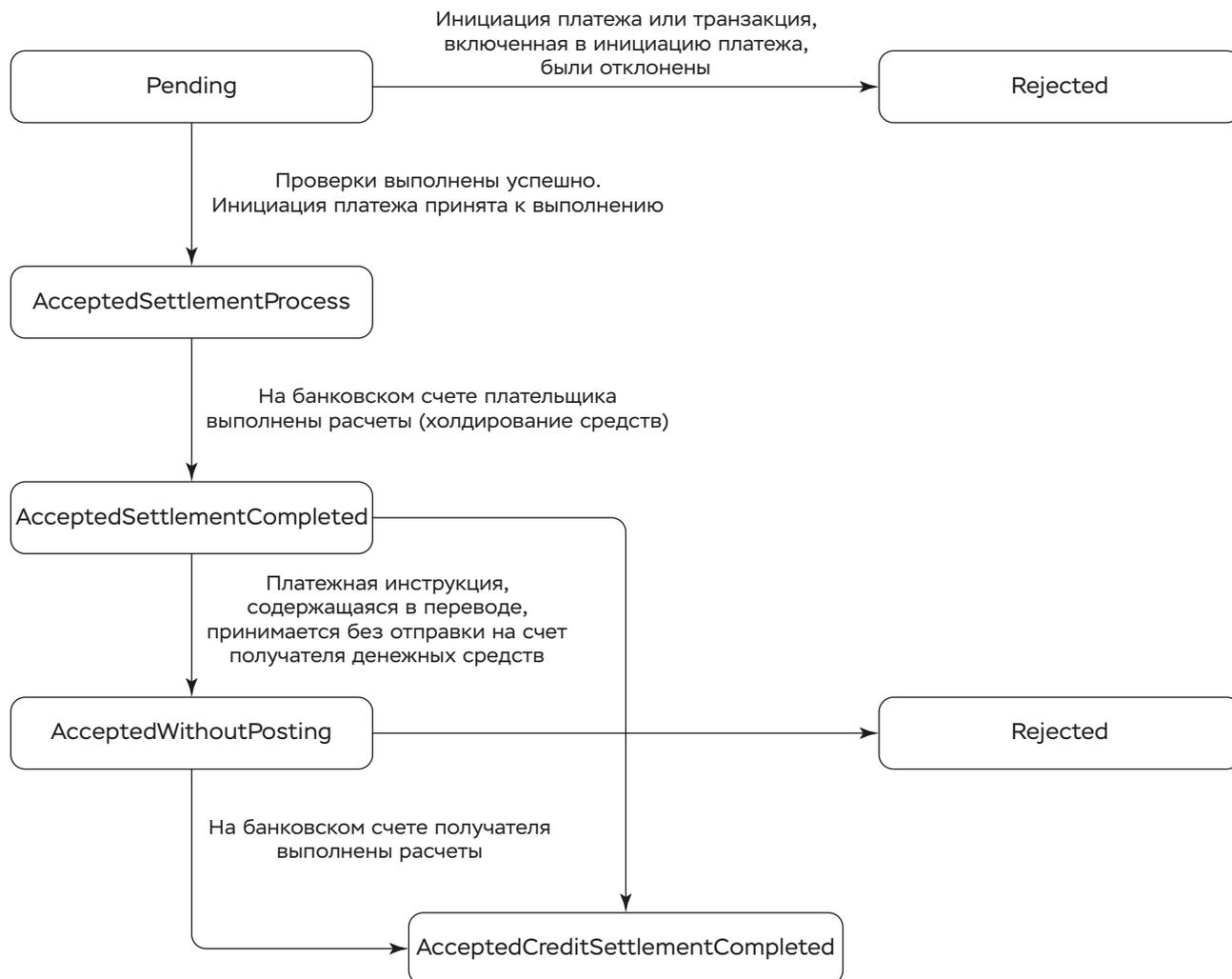


Рисунок 10. Диаграмма состояний статусов ресурса платежа

Описание статусов:

№	Статус	Наименование статуса	Описание
1	PDNG	Pending	Начальный статус выполнения платежа
2	RJCT	Rejected	Платеж отклонен
3	ACSP	AcceptedSettlementInProcess	Все проверки, такие как техническая проверка и профиль клиента, были успешными, и поэтому инициирование платежа было принято к исполнению
4	ACSC	AcceptedSettlementCompleted	Расчет по счету Плательщика выполнен
5	ACWP	AcceptedWithoutPosting	Платежная инструкция, включенная в перевод, принимается без зачисления на счет Получателя средств
6	ACCC	AcceptedCreditSettlementCompleted	Расчет по счету Получателя средств завершен

Таблица 46. Статусы ресурса платежа

**6.6.1.4.5. Множественная авторизация**

Если распоряжение о переводе средств требует множественной авторизации, то статус авторизации от нескольких Пользователей обновляется в объекте “MultiAuthorisation”.

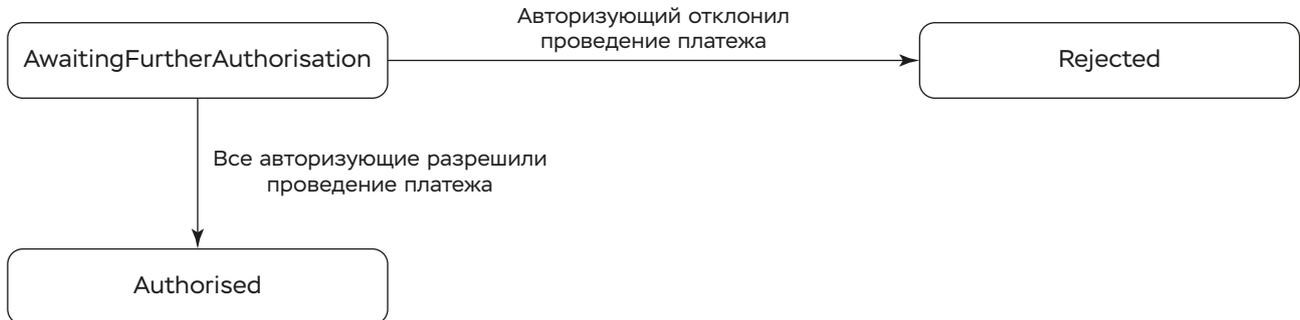


Рисунок 11. Диаграмма статусов ресурса платежа для множественной авторизации

Описание статусов:

№	Статус	Описание состояния
1	AwaitingFurtherAuthorisation	Ресурс платежа ожидает дальнейшей авторизации. Не все необходимые участники авторизовались
2	Rejected	Ресурс платежа был отклонен как минимум одним Пользователем, выполняющим авторизацию
3	Authorised	Ресурс платежа был успешно авторизован всеми необходимыми для этого Пользователями

Таблица 47. Статусы ресурса платежа для множественной авторизации

## 6.6.2. Модель данных

В данном разделе приведены подробные описания полезных нагрузок запросов и ответов для потоков API-платежей, осуществляемых на территории Российской Федерации.

### 6.6.2.1. Повторно используемые объекты

#### 6.6.2.1.1. Initiation

В данном разделе описывается объект Initiation, который повторно используется в ресурсах согласия на проведение платежа (payment-consent) и платежа (payment).

##### 6.6.2.1.1.1. UML-диаграмма

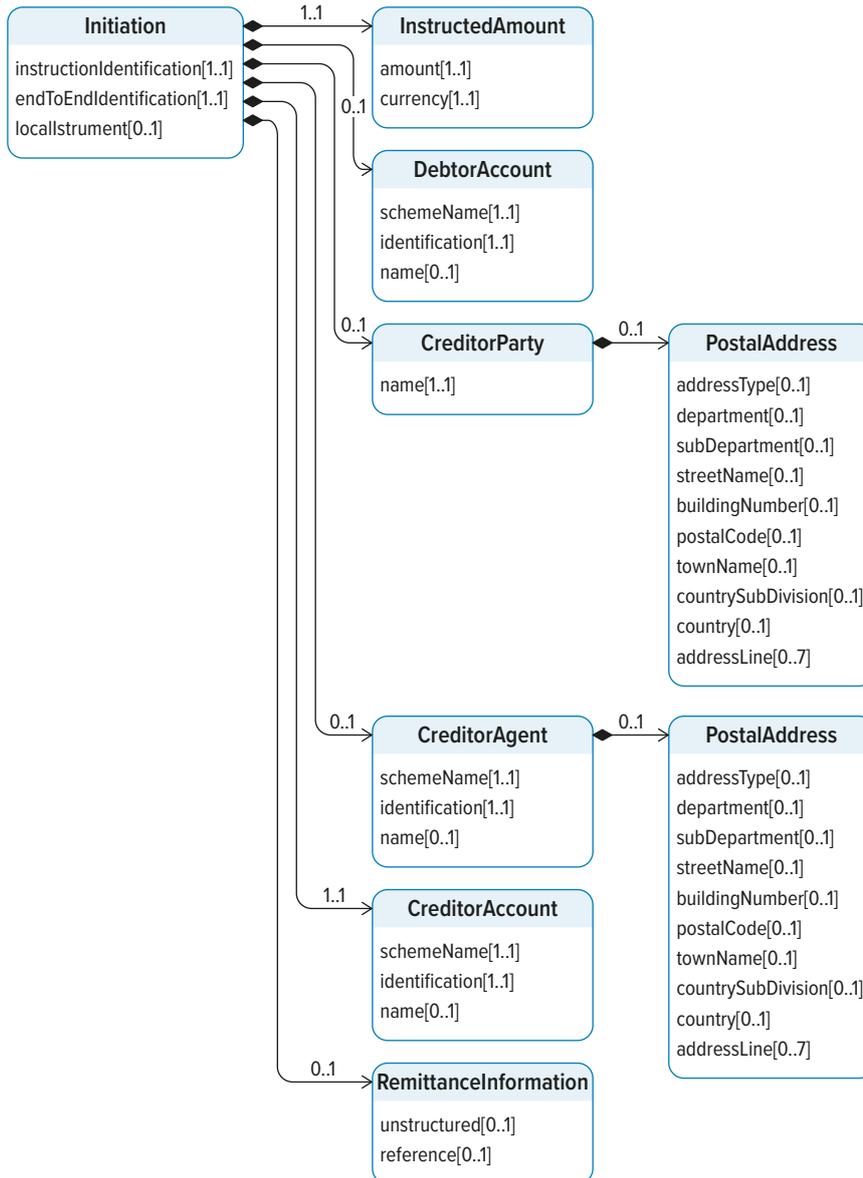


Рисунок 12. Диаграмма объекта Initiation

Требования к объекту “Initiation”, которые выполняют участники:

- Все элементы в объекте “Initiation”, которые заполнялись на стороне СППУ, не изменяются на стороне ППУ, поскольку это является официальным согласием Пользователя.
- Если ППУ определяет проблему с полезной нагрузкой во время вызова API, то ППУ немедленно отклоняет запрос получения согласия на проведение платежа.
- Если ППУ определяет проблему с согласием на платеж после вызова API, то ППУ изменяет статус ресурса согласия на проведение платежа на “Rejected”.
- Раздел “DebtorAccount” является необязательным, поскольку СППУ может не знать идентификационные данные банковского счета Пользователя.
- Если значения элементов объекта “DebtorAccount” заданы на стороне СППУ и являются неверными для Пользователя, то после аутентификации Пользователя статус согласия на проведение платежа меняется на “Rejected”.
- Элемент “CreditorReferenceInformation” был переименован на “Reference”, поскольку это уникальный элемент ISO 20022, используемый в сообщении pain.001, а не компонент сообщения ISO 20022.
- Поскольку продавец (merchant, а не Пользователь) может инициировать платеж через СППУ, в полезную нагрузку включены два идентификатора:
  - Идентификатор “instructionIdentification” генерируется СППУ.
  - СППУ гарантирует уникальность идентификатора “instructionIdentification” на неопределенный промежуток времени.
  - Идентификатор “endToEndIdentification” генерируется продавцом.
- Идентификатор “instructionIdentification” не используется в качестве идентификатора “consentId”, поскольку последний создается на стороне ППУ.
- Идентификатор “endToEndIdentification” не используется в качестве идентификатора “consentId”, поскольку последний создается на стороне ППУ.

### 6.6.2.1.2. Состав данных объекта Initiation

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
<b>Initiation</b>		<b>Initiation</b>	<b>Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса о переводе средств со счета Плательщика на счет Получателя средств для одиночного внутреннего платежа</b>	<b>InitiationComplexType</b>		
instructionIdentification	1.1	Initiation/ instructionIdentification	Уникальная идентификация, присвоенная инструктирующей стороной для проинструктированной стороны, для однозначной идентификации инструкции.  Использование: идентификация инструкции является сквозным идентификатором, который может использоваться между инструктирующей стороной и проинструктированной стороной для ссылки на индивидуальную инструкцию. Она может включаться в несколько сообщений, относящихся к инструкции	Max35Text		
endToEndIdentification	1.1	Initiation/ endToEndIdentification	Уникальная идентификация, присвоенная иницирующей стороной для однозначной идентификации операции. Эта идентификация передается без изменений по всей цепочке от начала до конца.  Использование: сквозной идентификатор может использоваться для выверки расчетов для связки задач, относящихся к операции. Она может включаться в несколько сообщений, относящихся к операции	Max35Text		
localInstrument	0..1	Initiation/localInstrument	Специальный инструмент, используемый сообществом.  Использование: этот элемент используется для указания местного инструмента, варианта локального клиринга и (или) для дополнительного описания сервиса или уровня обслуживания	ExternalLocalInstrument1 Code		
<b>InstructedAmount</b>	<b>1..1</b>	<b>Initiation/ InstructedAmount</b>	<b>Сумма денег, подлежащая переводу между Плательщиком и Получателем средств, до вычета комиссии, выраженная в валюте поручения инициатора</b>	<b>AmountComplexType</b>		
amount	1.1	Initiation/ InstructedAmount/amount	Сумма денежных средств, подлежащая перемещению между Плательщиком и Получателем средств, до вычета сборов, с указанием валюты, которую определила иницирующая сторона	ActiveCurrencyAnd Amount_SimpleType		^d{1,13}\.d{1,5}\$
currency	1.1	Initiation/ InstructedAmount/currency	Валюта согласно стандарту ISO 4217	ActiveOrHistoricCurrency Code		^[A-Z]{3,3}\$
<b>DebtorAccount</b>	<b>0..1</b>	<b>Initiation/DebtorAccount</b>	<b>Однозначная идентификация счета Плательщика, по которому будет отражена дебетовая запись в результате операции</b>	<b>CashAccountComplex Type</b>		
schemeName	1.1	Initiation/DebtorAccount/ schemeName	Наименование схемы идентификации	AccountIdentification DynamicType		
identification	1.1	Initiation/DebtorAccount/ identification	Уникальная и однозначная идентификация счета, используемая между владельцем счета и организацией, обслуживающей счет	Max256Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
name	0..1	Initiation/DebtorAccount/ name	Имя владельца или владельцев учетной записи (банковского счета, карты, телефона)	Max70Text		
<b>CreditorParty</b>	<b>0..1</b>	<b>Initiation/CreditorParty</b>	<b>Информация о контрагенте</b>	<b>PartyIdentification ComplexType</b>		
name	1..1	Initiation/CreditorParty/ name	Наименование контрагента	Max160Text		
<b>PostalAddress</b>	<b>0..1</b>	<b>Initiation/CreditorParty/ PostalAddress</b>	<b>Информация, которая находит и идентифицирует определенный адрес, как определено почтовыми службами</b>	<b>PostalAddressComplex Type</b>		
addressType	0..1	Initiation/CreditorParty/ PostalAddress/ addressType	Тип адреса	AddressTypeStaticType	Business Correspondence DeliveryTo MailTo POBox Postal Residential Statement	
department	0..1	Initiation/CreditorParty/ PostalAddress/department	Департамент или строение	Max70Text		
subDepartment	0..1	Initiation/CreditorParty/ PostalAddress/ subDepartment	Идентификация подразделения организации или идентификация здания	Max70Text		
streetName	0..1	Initiation/CreditorParty/ PostalAddress/streetName	Наименование улицы или проспекта	Max70Text		
buildingNumber	0..1	Initiation/CreditorParty/ PostalAddress/ buildingNumber	Номер, который определяет положение здания на улице	Max16Text		
postCode	0..1	Initiation/CreditorParty/ PostalAddress/postCode	Идентификатор, состоящий из группы букв и (или) цифр, которые добавляются к почтовому адресу для упрощения сортировки почты	Max16Text		
townName	0..1	Initiation/CreditorParty/ PostalAddress/townName	Наименование застроенного пространства с определенными границами и органами местного самоуправления	Max35Text		
countrySubDivision	0..1	Initiation/CreditorParty/ PostalAddress/ countrySubDivision	Определяет часть страны, например край, область, республика	Max35Text		
country	0..1	Initiation/CreditorParty/ PostalAddress/country	Государство	CountryCode		^[A-Z]{2,2}\$
addressLine	0..7	Initiation/CreditorParty/ PostalAddress/addressLine	Информация, которая определяет и идентифицирует определенный адрес, как это определено почтовыми службами, в форме свободного текста	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
<b>CreditorAgent</b>	<b>0..1</b>	<b>Initiation/CreditorAgent</b>	<b>Финансовая организация, обслуживающая счет Получателя средств</b>	<b>BranchAndFinancial InstitutionIdentification ComplexType</b>		
schemeName	1.1	Initiation/CreditorAgent/ schemeName	БИК/SWIFT банка контрагента	FinancialInstitution IdentificationDynamic Type		
identification	1.1	Initiation/CreditorAgent/ identification	БИК/SWIFT банка контрагента	Max35Text		
name	0..1	Initiation/CreditorAgent// name	Наименование банка контрагента	Max160Text		
<b>PostalAddress</b>	<b>0..1</b>	<b>Initiation/CreditorAgent/ PostalAddress</b>	<b>Информация, которая находит и идентифицирует определенный адрес, как определено почтовыми службами</b>	<b>PostalAddressComplex Type</b>		
addressType	0..1	Initiation/CreditorAgent/ PostalAddress/ addressType	Тип адреса	AddressTypeStaticType	Business Correspondence DeliveryTo MailTo POBox Postal Residential Statement	
department	0..1	Initiation/CreditorAgent/ PostalAddress/department	Департамент или строение	Max70Text		
subDepartment	0..1	Initiation/CreditorAgent/ PostalAddress/ subDepartment	Идентификация подразделения организации или идентификация здания	Max70Text		
streetName	0..1	Initiation/CreditorAgent/ PostalAddress/streetName	Наименование улицы или проспекта	Max70Text		
buildingNumber	0..1	Initiation/CreditorAgent/ PostalAddress/ buildingNumber	Номер, который определяет положение здания на улице	Max16Text		
postCode	0..1	Initiation/CreditorAgent/ PostalAddress/postCode	Идентификатор, состоящий из группы букв и (или) цифр, которые добавляются к почтовому адресу для упрощения сортировки почты	Max16Text		
townName	0..1	Initiation/CreditorAgent/ PostalAddress/townName	Наименование застроенного пространства с определенными границами и органами местного самоуправления	Max35Text		
countrySubDivision	0..1	Initiation/CreditorAgent/ PostalAddress/ countrySubDivision	Определяет часть страны, например край, область, республика	Max35Text		
country	0..1	Initiation/CreditorAgent/ PostalAddress/country	Государство	CountryCode		^[A-Z]{2,2}\$

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
addressLine	0..7	Initiation/CreditorAgent/PostalAddress/addressLine	Информация, которая определяет и идентифицирует определенный адрес, как это определено почтовыми службами, в форме свободного текста	Max70Text		
<b>CreditorAccount</b>	<b>1.1</b>	<b>Initiation/ CreditorAccount</b>	<b>Идентификатор счета получателя, на котором будет проведена запись о кредите в результате платежной транзакции</b>	<b>CashAccountComplexType</b>		
schemeName	1.1	Initiation/CreditorAccount/schemeName	Наименование схемы идентификации	AccountIdentificationDynamicType		
identification	1.1	Initiation/CreditorAccount/identification	Идентификатор счета Получателя средств, на котором будет проведена кредитная запись в результате платежной транзакции	Max256Text		
name	0..1	Initiation/CreditorAccount/name	Имя владельца или владельцев учетной записи (банковского счета, карты, телефона). ППУ могут выполнять проверку имени для подтверждения получателя, но это не обязательно	Max70Text		
<b>RemittanceInformation</b>	<b>0..1</b>	<b>Initiation/ RemittanceInformation</b>	<b>Предоставляемая информация, позволяющая сопоставить запись с позициями, для которых предназначен перевод, такими как коммерческие счета в системе дебиторской задолженности</b>	<b>RemittanceInformationComplexType</b>		
unstructured	0..1	Initiation/RemittanceInformation/unstructured	Информация, позволяющая сопоставлять/сверять записи с позициями, для которых предназначен платеж, такими как коммерческие счета в системе дебиторской задолженности, в неструктурированной форме	Max140Text		
reference	0..1	Initiation/RemittanceInformation/reference	Уникальная ссылка, присвоенная получателем, для однозначной ссылки на платежную транзакцию	Max35Text		

Таблица 48. Состав данных объекта Initiation

### 6.6.2.2. ConsentRequest

Объект ConsentRequest используется в запросе при вызове:

- POST /payment-consents

#### 6.6.2.2.1. UML-диаграмма

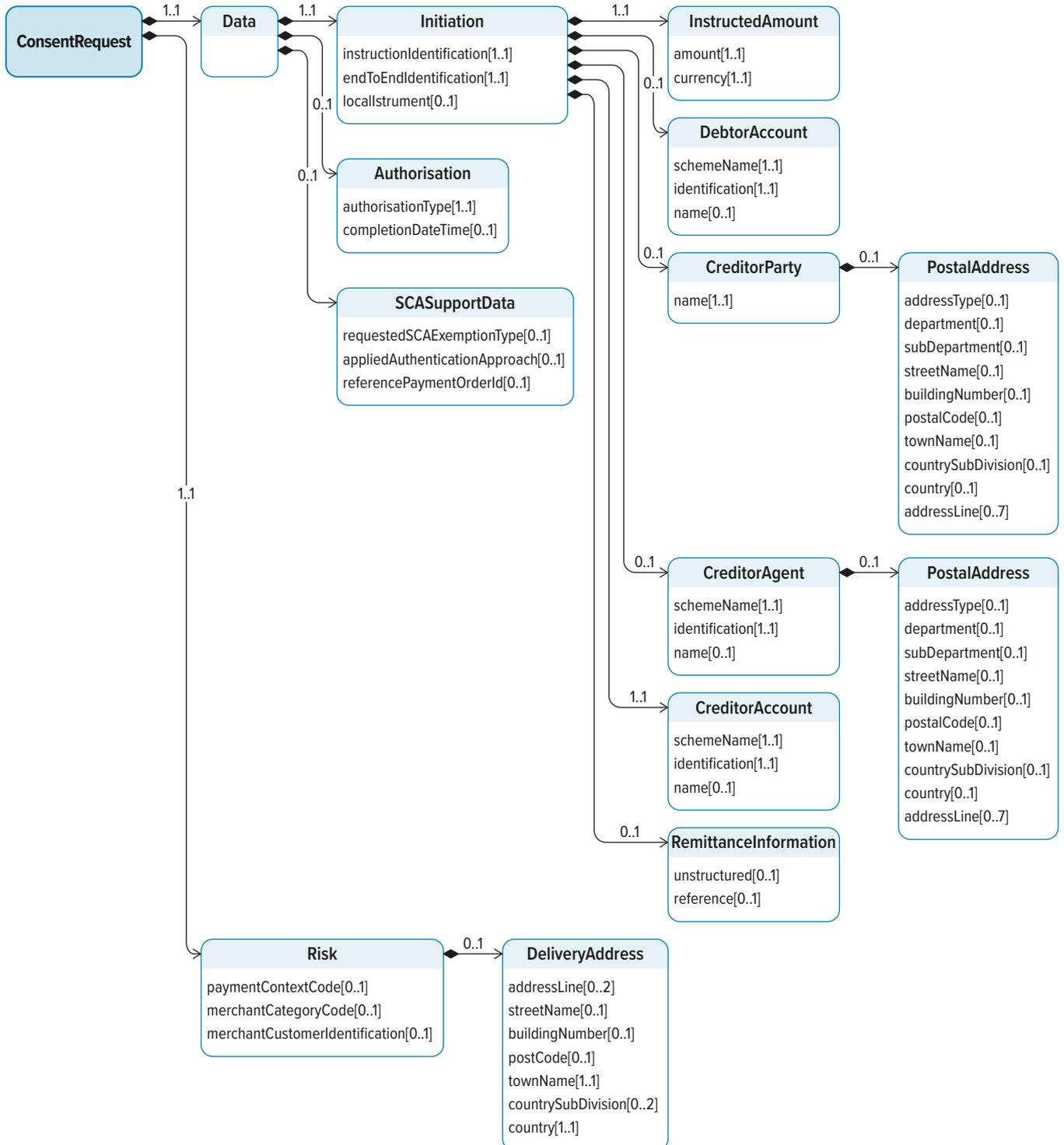


Рисунок 13. Диаграмма объекта ConsentRequest

Объект ConsentRequest состоит из следующих разделов (объектов):

- Initiation
- Authorisation
- SCASupportData
- Risk

#### 6.6.2.2.2. Состав данных объекта ConsentRequest

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
ConsentRequest				ConsentRequestComplexType		
Data	1..1	ConsentRequest/Data		DataConsentRequestComplexType		
Initiation	1..1	ConsentRequest/Data/Initiation	Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса перевода средств со счета Плательщика на счет Получателя средств для одиночного внутреннего платежа	InitiationComplexType		
Authorisation	0..1	ConsentRequest/Data/Authorisation	Запрос типа авторизации от СППУ	AuthorisationComplexType		
SCASupportData	0..1	ConsentRequest/Data/SCASupportData	Вспомогательные данные, предоставленные СППУ, при запросе SCA	SCASupportDataComplexType		
Risk	1..1	ConsentRequest/Risk	Раздел "Risk" отправляется инициатором в ППУ, используется для указания дополнительных деталей для оценки рисков при проведении платежей	RiskComplexType		

Таблица 49. Состав данных объекта ConsentRequest

### 6.6.2.3. ConsentResponse

Класс объекта ConsentResponse используется при ответе на вызов:

- POST /payment-consents
- GET /payment-consents/{consentId}

#### 6.6.2.3.1. UML-диаграмма

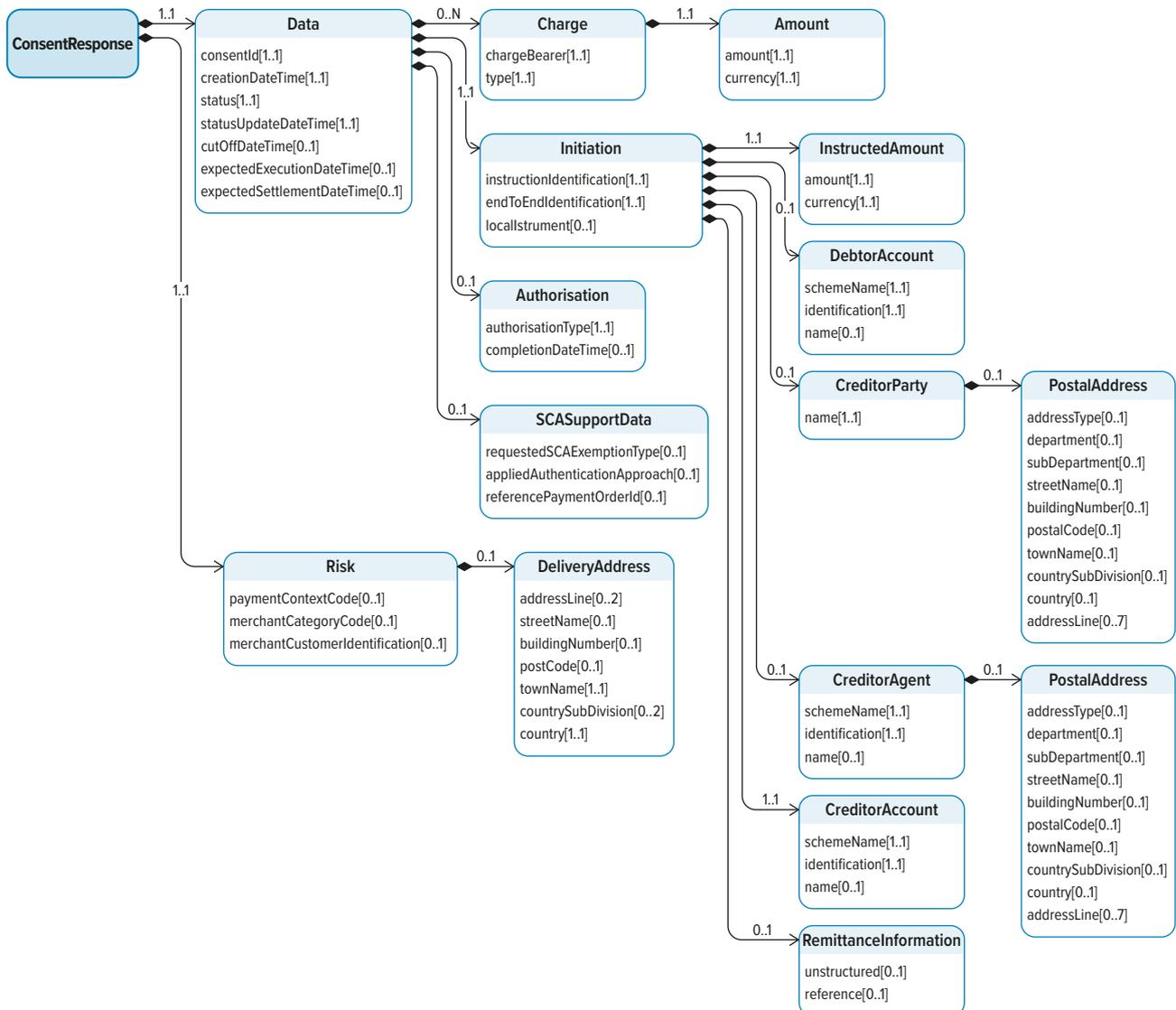


Рисунок 14. Диаграмма объекта ConsentResponse

Ответ согласия на проведение платежа содержит полную полезную нагрузку запроса согласия на проведение платежа с дополнительными элементами:

- Идентификатор ресурса согласия на проведение платежа consentId, созданного на стороне ППУ.
- Элемент creationDateTime — дата и время создания ресурса согласия на проведение платежа.
- Элементы status и statusUpdateDateTime — статус, дата и время изменения статуса ресурса согласия на проведение платежа.
- Элемент CutOffDateTime и его использование описаны в разделе “Использование элемента CutOffDateTime”
- Элемент “expectedExecutionDateTime” означает прогнозируемую дату и время выполнения платежа и предназначен для ресурса платежа. Если ресурс платежа создан до даты “cutOffDateTime”, то “expectedExecutionDateTime” означает ожидаемую дату и время списания средств со счета Плательщика. ППУ обновляет значение этого поля при любых изменениях (например, после авторизации Пользователя).
- Элемент “expectedSettlementDateTime” означает прогнозируемую дату и время выполнения платежа и предназначен для ресурса платежа. Если ресурс платежа создан до даты “cutOffDateTime”, то “expectedSettlementDateTime” означает ожидаемую дату и время зачисления средств на счет Получателя средств. ППУ обновляет значение этого поля при любых изменениях (например, после авторизации Пользователя).
- Элемент “Charges” — это массив для разбивки применимых сборов ППУ (разбиение комиссий).

## 6.6.2.3.2. Состав данных объекта ConsentResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
ConsentResponse				ConsentResponseComplexType		
Data	1..1	ConsentResponse/Data		DataConsentRequestComplexType		
consentId	1..1	ConsentResponse/Data/consentId	Идентификатор ресурса согласия на проведение платежа, присвоенный на стороне ППУ	Max128Text		
creationDateTime	1..1	ConsentResponse/Data/creationDateTime	Дата и время создания ресурса согласия на проведение платежа	ISODateTime		
status	1..1	ConsentResponse/Data/status	Текущий статус ресурса согласия на проведение платежа	ConsentStatusStaticType	Authorised AwaitingAuthorisation Consumed Rejected	
statusUpdateDateTime	1..1	ConsentResponse/Data/statusUpdateDateTime	Дата и время обновления статуса ресурса согласия на проведение платежа	ISODateTime		
cutOffDateTime	0..1	ConsentResponse/Data/cutOffDateTime	Дата и время актуальности согласия на проведение платежа (после этого согласие является недействующим)	ISODateTime		
expectedExecutionDateTime	0..1	ConsentResponse/Data/expectedExecutionDateTime	Ожидаемая дата и время исполнения платежа (платежного ресурса)	ISODateTime		
expectedSettlementDateTime	0..1	ConsentResponse/Data/expectedSettlementDateTime	Ожидаемая дата и время расчета платежа (платежного ресурса)	ISODateTime		
Charges	0..N	ConsentResponse/Data/Charges	Сборы при проведении платежа	ChargeComplexType		
Initiation	1..1	ConsentResponse/Data/Initiation	Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса перевода средств со счета плательщика на счет получателя для одиночного внутреннего платежа	InitiationComplexType		
Authorisation	0..1	ConsentResponse/Data/Authorisation	Запрос типа авторизации от СППУ	AuthorisationComplexType		
SCASupportData	0..1	ConsentResponse/Data/SCASupportData	Вспомогательные данные, предоставленные СППУ, при запросе SCA	SCASupportDataComplexType		
Risk	1..1	ConsentResponse/Risk	Раздел "Risk" отправляется инициатором в ППУ, используется для указания дополнительных деталей для оценки рисков при проведении платежей	RiskComplexType		

Таблица 50. Состав данных объекта ConsentResponse

#### 6.6.2.4. PaymentRequest

Объект PaymentRequest используется в запросе при вызове:

- POST /payments
- GET /payments/{paymentId}

##### 6.6.2.4.1. UML-диаграмма

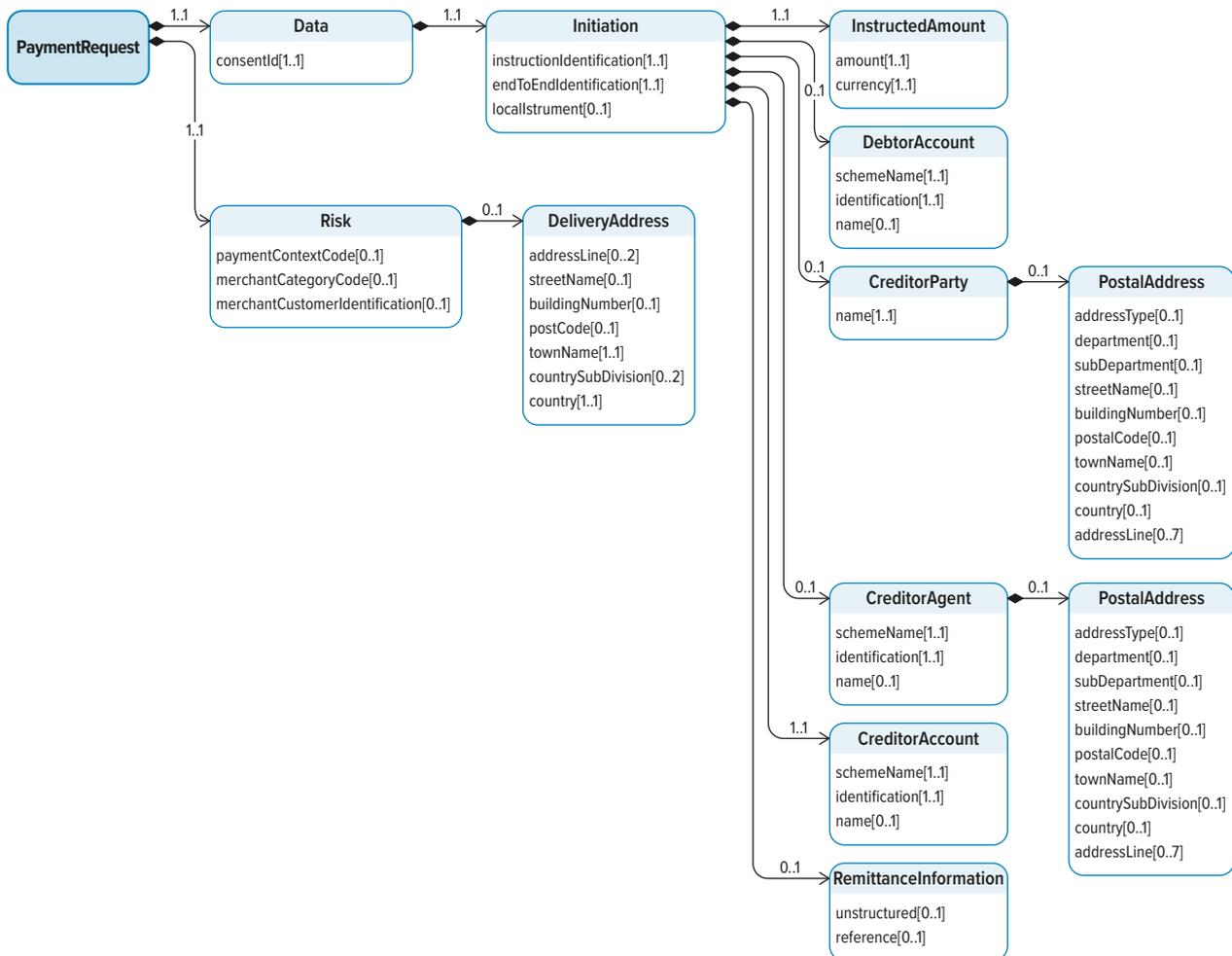


Рисунок 15. Диаграмма объекта PaymentRequest

Объект “PaymentRequest” содержит следующие элементы:

- Идентификатор соответствующего ресурса согласия на проведение платежа consentId.
- Объекты “Initiation” и “Risk” и все их элементы, которые содержались в запросе на согласие о проведении платежа.

Все элементы объектов “Initiation” и “Risk” в запросе на проведение платежа должны совпадать с соответствующими элементами объектов “Initiation” и “Risk”, которые были в запросе согласия на проведение платежа. Проверка соответствия осуществляется как на стороне СППУ, так и на стороне ППУ. В случае несовпадения ресурс платежа не создается, ресурс согласия переводится в статус “Rejected”, и ППУ отвечает СППУ сообщением об ошибке несовпадения данных.

#### 6.6.2.4.2. Состав данных объекта PaymentRequest

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
PaymentRequest				PaymentRequestComplexType		
Data	1..1	PaymentRequest/Data		DataPaymentRequestComplexType		
consentId	1..1	PaymentRequest/Data/consentId	Идентификатор ресурса согласия на проведение платежа, присвоенный на стороне ППУ	Max128Text		
Initiation	1..1	PaymentRequest/Data/Initiation	Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса перевода средств со счета плательщика на счет получателя для одиночного внутреннего платежа	InitiationComplexType		
Risk	1..1	PaymentRequest/Risk	Раздел "Risk" отправляется инициатором в ППУ, используется для указания дополнительных деталей для оценки рисков при проведении платежей	RiskComplexType		

Таблица 51. Состав данных объекта PaymentRequest

### 6.6.2.5. PaymentResponse

Объект PaymentResponse используется при ответе на вызов:

- POST /domestic-payments

#### 6.6.2.5.1. UML-диаграмма

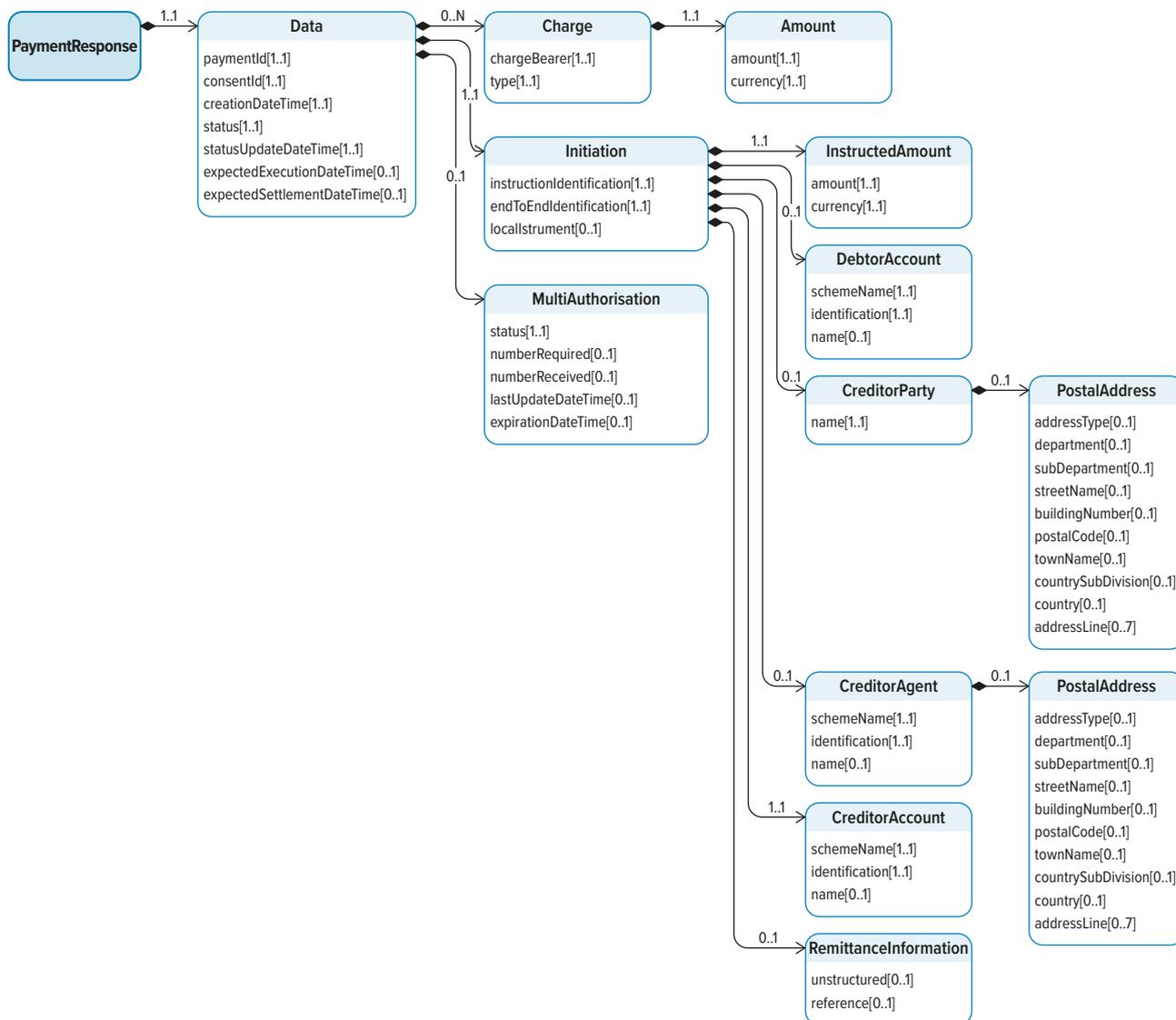


Рисунок 16. Диаграмма объекта PaymentResponse

Объект PaymentResponse содержит следующие элементы:

- Элемент paymentId — идентификатор ресурса платежа, созданный на стороне ППУ.
- Элемент consentId — идентификатор ресурса согласия на проведение платежа, созданный на стороне ППУ.
- Элемент creationDateTime — дата и время создания ресурса платежа.
- Элементы status и statusUpdateDateTime — статус, дата и время изменения статуса ресурса платежа.
- Элемент expectedExecutionDateTime означает ожидаемую дату и время списания средств со счета плательщика.
- Элемент expectedSettlementDateTime означает ожидаемую дату и время зачисления средств на счет получателя.
- Элемент Charges — это массив для разбивки применимых сборов ППУ (разбиение комиссий).
- Объект Initiation, соответствующий одноименному объекту из запроса согласия на проведение платежа.
- Объект MultiAuthorisation, если для ресурса платежей требуется множественная авторизация.

## 6.6.2.5.2. Состав данных объекта PaymentResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
PaymentResponse				PaymentResponseComplexType		
Data	1..1	PaymentResponse/Data		DataPaymentRequestComplexType		
consentId	1..1	PaymentResponse/Data/consentId	Идентификатор ресурса согласия на проведение платежа, присвоенный на стороне ППУ	Max128Text		
creationDateTime	1..1	PaymentResponse/Data/creationDateTime	Дата и время создания ресурса согласия на проведение платежа	ISODatetime		
status	1..1	PaymentResponse/Data/status	Текущий статус ресурса согласия на проведение платежа	TransactionStatusStaticType	AcceptedCreditSettlementCompleted AcceptedWithoutPosting AcceptedSettlementCompleted AcceptedSettlementInProgress Pending Rejected	
statusUpdateDateTime	1..1	PaymentResponse/Data/statusUpdateDateTime	Дата и время обновления статуса ресурса согласия на проведение платежа	ISODatetime		
expectedExecutionDateTime	0..1	PaymentResponse/Data/expectedExecutionDateTime	Ожидаемая дата и время исполнения платежа (платежного ресурса)	ISODatetime		
expectedSettlementDateTime	0..1	PaymentResponse/Data/expectedSettlementDateTime	Ожидаемая дата и время расчета платежа (платежного ресурса)	ISODatetime		
Charges	0..N	PaymentResponse/Data/Charges	Сборы при проведении платежа	ChargeComplexType		
Initiation	1..1	PaymentResponse/Data/Initiation	Полезная нагрузка отправляется иницилирующей стороной в ППУ. Используется для запроса перевода средств со счета плательщика на счет получателя для одиночного внутреннего платежа	InitiationComplexType		
MultiAuthorisation	0..1	PaymentResponse/Data/MultiAuthorisation	Ответ потока множественной авторизации от ППУ	MultiAuthorisationComplexType		

Таблица 52. Состав данных объекта PaymentResponse

### 6.6.3. Примеры использования

#### 6.6.3.1. Инициирование продавцом через СППУ

Для данного случая оплаты:

- Продавец не указывает реквизиты счета Плательщика.
- Пользователь (Плательщик) выберет реквизиты своего счета во время авторизации согласия.

Пользователь Иван Иванов, зарегистрированный в магазине электронной коммерции “MERCHANT Inc” под идентификатором 053598653254, приобретает товары на сумму 23 463,00 рубля с указанием адреса доставки. Продавец присваивает данной платежной инструкции сквозной идентификатор (endToEndIdentification) MERCHANT.256702.IDN.12 и намеревается инициировать через СППУ, в которой зарегистрирован Пользователь, проведение оплаты.

##### 6.6.3.1.1. Создание согласия на проведение платежа

Запрос POST /payment-consents. СППУ присваивает инициируемой платежной инструкции номер (instructionIdentification) PISP412 и создает на стороне ППУ ресурс согласия на проведение платежа вызовом конечной точки POST /payment-consents, указывая идентификаторы (“instructionIdentification”: “PISP412”, “endToEndIdentification”: “MERCHANT.256702.IDN.12”), реквизиты “MERCHANT Inc” и указание дополнительных деталей для оценки рисков платежей в разделе Risk.

Ответ POST /payment-consents. ППУ производит обработку запроса и возвращает идентификатор согласия (consentId) “58923” со статусом “AwaitingAuthorisation”, указав дату создания (creationDateTime) (дата обновления статуса (statusUpdateDateTime) в данном случае идентична дате создания).

Далее СППУ перенаправляет Пользователя на страницу ППУ и аутентифицирует его. Пользователь производит авторизацию согласия предоставления своих персональных данных от ППУ к СППУ для создания возможности проведения на стороне СППУ инициирования платежа с реквизитами Пользователя.

```

POST /payment-consents HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature: UTOmZSdzIGEgam91ckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rkl4fIG1yIHR5bGVyIGdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    }
  },
  "Risk": {
    "paymentContextCode": "EcommerceGoods",
    "merchantCategoryCode": "5967",
    "merchantCustomerIdentification": "053598653254",
    "DeliveryAddress": {
      "addressLine": [
        "Шлюзовая наб., 4, Москва, 115114",
        "Rosso Riva"
      ],
      "streetName": "Шлюзовая наб.",
      "buildingNumber": "4",
      "postCode": "115114",
      "townName": "Moscow",
      "countySubDivision": [
        "Moscow"
      ],
      "country": "RU"
    }
  }
}

```

Таблица 53. Запрос POST /payment-consents

```
HTTP/1.1 201 Created
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "AwaitingAuthorisation",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    }
  },
  "Risk": {
    "paymentContextCode": "EcommerceGoods",
    "merchantCategoryCode": "5967",
    "merchantCustomerIdentification": "053598653254",
    "DeliveryAddress": {
      "addressLine": [
        "Шлюзовая наб., 4, Москва, 115114",
        "Rosso Riva"
      ],
      "streetName": "Шлюзовая наб.",
      "buildingNumber": "4",
      "postCode": "115114",
      "townName": "Moscow",
      "countySubDivision": [
        "Moscow"
      ],
      "country": "RU"
    }
  }
}
```

Таблица 54. Ответ POST /payment-consents

#### 6.6.3.1.2. Создание платежа

Запрос POST /payment. СППУ, получив данные Пользователя, дополняет платежную инструкцию данными Пользователя, указав его идентификатор счета и имя ("Identification": "40817810621234567232", "Name": "Иван Иванов"), а также идентификатор согласия "consentId": "58923" и создает на стороне ППУ ресурс платежа вызовом конечной точки POST /payments.

Ответ на POST /payment-consents. ППУ производит обработку (и процесс исполнения) платежа и возвращает СППУ его идентификатор (paymentId) 58923-001.

```

POST /payments HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature: UTOmZSdzIGEgam9lckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rkl4fIGlyIHR
5bGVyIGdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json
{
  "Data": {
    "consentId": "58923",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567232",
        "Name": "Иван Иванов"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код
операции 1234567"
      }
    }
  },
  "Risk": {
    "paymentContextCode": "EcommerceGoods",
    "merchantCategoryCode": "5967",
    "merchantCustomerIdentification": "053598653254",
    "DeliveryAddress": {
      "addressLine": [
        "Шлюзовая наб., 4, Москва, 115114",
        "Rosso Riva"
      ],
      "streetName": "Шлюзовая наб.",
      "buildingNumber": "4",
      "postCode": "115114",
      "townName": "Moscow",
      "countySubDivision": [
        "Moscow"
      ],
      "country": "RU"
    }
  }
}

```

Таблица 55. Запрос POST /payment

```
HTTP/1.1 201 Created
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567232",
        "Name": "Иван Иванов"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 56. Ответ POST /payment

#### 6.6.3.1.3. Получение согласия на проведение платежа

Запрос GET /payment-consents. СППУ выполняет запрос к ППУ на получение списка ранее данных Пользователем согласий на проведение платежей вызовом конечной точки GET /payment-consents без параметров.

Ответ на GET /payment-consents. ППУ возвращает СППУ согласие на проведение платежа с идентификатором (consentId) "58923", имеющего статус "Authorized".

```
GET /payment-consents/58923 HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 57. Запрос GET /payment-consents

```

HTTP/1.1 200 OK
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "Authorised",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
    },
    "CreditorAccount": {
      "schemeName": "RU.CBR.AccountNumber",
      "identification": "40817810621234567890",
      "name": "MERCHANT Inc"
    },
    "RemittanceInformation": {
      "Reference": "CBR-130",
      "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
    }
  },
  "Risk": {
    "paymentContextCode": "EcommerceGoods",
    "merchantCategoryCode": "5967",
    "merchantCustomerIdentification": "053598653254",
    "DeliveryAddress": {
      "addressLine": [
        "Шлюзовая наб., 4, Москва, 115114",
        "Rosso Riva"
      ],
      "streetName": "Шлюзовая наб.",
      "buildingNumber": "4",
      "postCode": "115114",
      "townName": "Moscow",
      "countySubDivision": [
        "Moscow"
      ],
      "country": "RU"
    }
  }
}

```

Таблица 58. Ответ GET /payment-consents

**6.6.3.1.4. Получение платежа**

Запрос GET /payments. СППУ выполняет запрос к ППУ на получение списка инициированных платежей вызовом конечной точки GET /payments без параметров.

Ответ на GET /payments. ППУ возвращает СППУ платеж с идентификатором (paymentId) "58923-001", имеющий согласие (consentId): "58923" и статус "AcceptedSettlementInProgress".

```
GET /payments/58923-001 HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 59. Запрос GET /payment

```
HTTP/1.1 200 OK
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQOK..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQOK
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567232",
        "Name": "Иван Иванов"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 60. Ответ GET /payment

### 6.6.3.2. Инициирование пользователем через СППУ

Для данного случая оплаты:

- Пользователь (Платательщик) выбирает реквизиты своего счета во время инициирования платежа (при создании ресурса согласия на проведение платежа).

Пользователь Петр Петров намерен инициировать через СППУ перевод денежных средств в размере 23 463,00 рубля в пользу Петрова Петра, указав его номер счета "40817810621234567754".

#### 6.6.3.2.1. Создание согласия на проведение платежа

Запрос POST /payment-consents. СППУ присваивает данной платежной инструкции номер (instructionIdentification) PISP412 и создает на стороне ППУ ресурс согласия на проведение платежа вызовом конечной точки POST /payment-consents, указывая номер инструкции и реквизиты участников перевода.

Ответ POST /payment-consents. ППУ производит обработку запроса и возвращает идентификатор согласия (consentId) "58923" со статусом "AwaitingAuthorisation", указав дату создания (creationDateTime) (дата обновления статуса (statusUpdateDateTime) в данном случае идентична дате создания).

Далее СППУ перенаправляет Пользователя на страницу ППУ и аутентифицирует его. Пользователь производит авторизацию согласия проведения на стороне СППУ инициирования платежа.

```
POST /payment-consents HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature: UTOmZSdzIGEgam91ckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rk14fIG1yIHR5bGVyIGdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 61. Запрос POST /payment-consents

```
HTTP/1.1 201 Created
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "AwaitingAuthorisation",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 62. Ответ POST /payment-consents

**6.6.3.2.2. Создание платежа**

Запрос POST /payment. СППУ, получив согласие Пользователя на проведение платежа, дополняет платежную инструкцию идентификатором согласия "consentId": "58923" и создает на стороне ППУ ресурс платежа вызовом конечной точки POST /payments.

Ответ на POST /payment-consents. ППУ производит обработку (и процесс исполнения) платежа и возвращает СППУ его идентификатор (paymentId) 58923-001.

```
POST /payments HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature: UTOmZSdzIGEgam9lckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rk14fIG1yIHR5bGVyIGdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json
{
  "Data": {
    "consentId": "58923",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 63. Запрос POST /payment

```
HTTP/1.1 201 Created
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств. Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 64. Ответ POST /payment

#### 6.6.3.2.3. Получение согласия на проведение платежа

Запрос GET /payment-consents. СППУ выполняет запрос к ППУ на получение списка ранее предоставленных Пользователем согласий на проведение платежей вызовом конечной точки GET /payment-consents без параметров.

Ответ на GET /payment-consents. ППУ возвращает СППУ согласие на проведение платежа с идентификатором (consentId) "58923", имеющего статус "Consumed".

```
GET /payment-consents/58923 HTTP/1.1
Authorization: Bearer 3YotnDFGEjr1zCsicMWpSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 65. Запрос GET /payment-consents

```

HTTP/1.1 200 OK
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "Consumed",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}

```

Таблица 66. Ответ GET /payment-consents

**6.6.3.2.4. Получение платежа**

Запрос GET /payments. СППУ выполняет запрос к ППУ на получение списка инициированных платежей вызовом конечной точки GET /payments без параметров.

Ответ на GET /payments. ППУ возвращает СППУ платеж с идентификатором (paymentId) "58923-001", имеющий согласие (consentId): "58923" и статус "AcceptedSettlementInProgress".

```

GET /payments/58923-001 HTTP/1.1
Authorization: Bearer 3YotnDFGEjr1zCsicMwPSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json

```

Таблица 67. Запрос GET /payment

```
HTTP/1.1 200 OK
x-jws-signature: D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств. Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 68. Ответ GET /payment

## 7. Динамические справочники в пространствах имен – v1.0.0

### 7.1. Введение

Спецификация определяет поля только с фиксированным набором возможных значений, а дальнейшее добавление значений требует изменения спецификации.

В рамках текущей версии определены новые типы данных, представляющие собой расширяемый список значений. Любые расширения этого стандартного списка значений могут быть сделаны ППУ с соответствующей документацией на их порталах для разработчиков.

Значения расширяемых типов данных располагаются в соответствующих пространствах имен, чтобы помочь идентифицировать источник значения и соответствующее значение.

### 7.2. Основы

Элементы справочников, определенные участниками Ассоциации развития финансовых технологий, задокументированы здесь и будут иметь префикс RU.CBR.

#### 7.2.1. Принципы проектирования

При добавлении собственных значений в справочники участники среды Открытых банковских интерфейсов с ролью ППУ помещают такие значения в пространство имен, состоящее из двухбуквенного кода страны (код ISO 3166-1 Alpha-2), после которого ставится точка, после которой следует наименование организации. Например:

- RU.SBRF.Int-payments
- RU.VTB.Int-payments
- KE.Safaricom.M-Pesa

#### 7.2.2. Управление версиями

В данном разделе будут описываться требования к обратной совместимости при разработке последующих версий Стандарта.

### 7.3. Общие справочники

#### AccountIdentificationDynamicType

Значение	Описание
RU.CBR.PAN	Primary Account Number — схема идентификатора, используемая для идентификации карточного счета
RU.CBR.CellphoneNumber	Схема для осуществления платежа по номеру телефона
RU.CBR.BBAN	Схема для осуществления платежа по номеру счета

Таблица 69. Идентификатор счета

#### FinancialInstitutionIdentificationDynamicType

Значение	Описание
RU.CBR.BICFI	BIC для финансовых учреждений согласно ISO 9362
RU.CBR.BIK	Уникальный идентификатор банка, используемый в платежных документах на территории Российской Федерации

Таблица 70. Идентификатор финансового учреждения

### OBRUErrorResponseErrorCode

Тип данных, который дает низкоуровневый текстовый код ошибки для ее классификации. Применяется также код ответа HTTP.

Значение	HTTP-статус	Описание
RU.CBR.Field.Expected	400	Если поля передаются парой (ключ-значение) и значение не было передано. В поле path должен передаваться путь к ожидаемому полю (например, ErrorResponse.Errors.path == "AccountResponse.Data.Account.AccountDetails.identification"). Например, для допустимого значения поля "schemeName" должно передаваться соответствующее значение идентификатора в поле "identification"
RU.CBR.Field.Invalid	400	В поле указано недопустимое значение или длина предоставленного значения превышает соответствующую максимальную длину поля в домене ППУ. Ссылка на недопустимое поле должна быть указана в поле path (например, ErrorResponse.Errors.path == "AccountResponse.Data.Account.AccountDetails.schemeName"). В поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение. Проблема должна быть подробно описана в сообщении об ошибке (поле ErrorResponse.Errors.message)
RU.CBR.Field.InvalidDate	400	Указана неверная дата. Например, когда ожидается будущая дата, а указана дата в прошлом или текущая дата. В сообщении можно указать актуальную проблему с датой. Ссылка на недопустимое поле должна быть указана в поле path, а в поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение
RU.CBR.Field.Missing	400	Обязательное поле, необходимое для API, отсутствует в полезной нагрузке. Данный код ошибки можно использовать, если ошибка еще не определена при проверке RU.CBR.Resource.InvalidFormat. Ссылка на отсутствующее поле должна быть указана в поле path, а в поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение
RU.CBR.Header.Invalid	400	В элементе заголовка HTTP указано неверное значение. Элемент заголовка HTTP должен быть указан в элементе пути
RU.CBR.Header.Missing	400	Обязательный элемент HTTP-заголовка не был предоставлен. Элемент заголовка HTTP должен быть указан в элементе path
RU.CBR.Resource.ConsentMismatch	400	Несоответствие ресурсов "payment-consent" и "payment". Например, если элемент в разделе "Initiation" или "Risk" ресурса платежа не совпадает с одноименным элементом в соответствующем разделе ресурса согласия. Элемент пути должен быть заполнен элементом ресурса платежа, который не соответствует согласию
RU.CBR.Resource.InvalidConsentStatus	400	Согласие, соответствующее ресурсу, находится в некорректном статусе, который бы позволил создать ресурс или выполнить запрос. Например, если ресурс согласия имеет статус AwaitingAuthorisation или Rejected, то ресурс не может быть создан с таким статусом соответствующего ему согласия. Элемент пути должен быть заполнен элементом ресурса согласия, который является недопустимым
RU.CBR.Resource.InvalidFormat	400	Когда json-схема полезной нагрузки не соответствует конечной точке. Например, конечная точка POST /payments вызывается с полезной нагрузкой JSON, которая не может быть проанализирована в классе PaymentRequest
RU.CBR.Resource.NotFound	400	Возвращается, когда ресурс с указанным идентификатором не существует (ресурс не может быть обработан)
RU.CBR.Resource.NotCreated	400	Возвращается, когда ресурс с указанным идентификатором еще не создан и не может быть передан в ответном сообщении. Для асинхронных вызовов. Например, получение выписки по счету, где сначала создается ресурс выписки (метод POST /statements/{accountId}) и в ответном сообщении приходит идентификатор созданного ресурса выписки, но для наполнения выписки данными ППУ требуется некоторое время. Соответственно, будет приходиться данное сообщение об ошибке
RU.CBR.Rules.AfterCutOffDateTime	400	Ресурс согласия или ресурс платежа запрашиваются после даты CutOffDateTime
RU.CBR.Signature.Invalid	400	Заголовок подписи x-jws-signature был проанализирован и имеет действительный заголовок JOSE, соответствующий спецификации. Но сама подпись не может быть проверена

Значение	HTTP-статус	Описание
RU.CBR.Signature.InvalidClaim	400	Заголовок JOSE в элементе x-jws-signature имеет одно или несколько утверждений (claim) с недопустимым значением (например, утверждение kid, которое не принимает сертификат). Наименование отсутствующего утверждения должно передаваться в поле path ответа об ошибке
RU.CBR.Signature.MissingClaim	400	Заголовок JOSE в элементе x-jws-signature имеет одно или несколько обязательных утверждений, которые не указаны. Имя пропущенного утверждения должно быть указано в поле path ответа об ошибке
RU.CBR.Signature.Malformed	400	x-jws-signature в заголовке запроса была искажена и не могла быть проанализирована как допустимый JWS
RU.CBR.Signature.Missing	400	Запрос API предполагает x-jws-signature в заголовке, но элемент отсутствовал
RU.CBR.Unsupported.AccountIdentifier	400	Идентификатор счета не поддерживается для данной схемы. Элемент path должен быть заполнен путем к элементу accountIdentifier
RU.CBR.Unsupported.LocalInstrument	400	Указанный localInstrument не поддерживается ППУ. Элемент path должен быть заполнен путем к элементу localInstrument. Элемент URL должен быть заполнен ссылкой на документацию ППУ со списком поддерживаемых localInstrument
RU.CBR.Reauthenticate	403	Данный код ошибки указывает, что для обработки запроса требуется повторная аутентификация Пользователя
RU.CBR.Rules.ResourceAlreadyExists	409	Данный код ошибки указывает, что ресурс с такими же параметрами уже существует
RU.CBR.UnexpectedError	5xx	Данный код ошибки можно использовать при возникновении непредвиденной ошибки. ППУ должен заполнить сообщение детальным описанием ошибки, не раскрывая конфиденциальную информацию

Таблица 71. Низкоуровневая классификация ошибок



СТАНДАРТ БАНКА РОССИИ

# ОТКРЫТЫЕ БАНКОВСКИЕ ИНТЕРФЕЙСЫ

Получение информации о счете  
клиента третьей стороной

Дата введения: 2020-10-23

Москва  
2020

# Оглавление

<b>1. История изменений</b>	<b>92</b>
<b>2. Введение</b>	<b>93</b>
2.1. Область применения	93
2.2. Термины и определения	93
2.3. Принципы архитектуры	94
2.3.1. Идентификаторы	94
2.3.2. Коды статусов	94
<b>3. Основы</b>	<b>95</b>
3.1. Роли участников	95
3.2. Кодировка символов	95
3.3. Формат даты	95
3.4. Структура пути URI-ресурса	96
3.5. Заголовки сообщений (headers)	97
3.5.1. Заголовки запросов	97
3.5.2. Заголовки ответов	99
3.6. Коды статусов HTTP	99
3.6.1. 400 (Bad Request) или 404 (Not Found)	100
3.6.2. 403 (Forbidden)	101
3.6.3. 401 (Unauthorized)	101
3.6.4. 429 (Too Many Requests)	101
3.7. Идемпотентность	101
3.8. Фильтрация	102
3.9. Нумерация страниц	102
3.10. Архивирование	102
3.11. Дополнительные данные	103
<b>4. Модель данных</b>	<b>104</b>
4.1. Справочники и перечисления	104
4.2. Общая структура полезной нагрузки	104
4.2.1. Структура запроса	104
4.2.2. Структура ответа	105
4.2.3. Структура ответа с ошибками	105
4.2.4. Необязательные поля	106
4.2.5. Ссылки	107
4.2.6. Метаданные	107
<b>5. Примеры использования</b>	<b>108</b>
5.1. Потоки с нумерацией страниц	108
<b>6. Счета и транзакции, спецификация API – v1.2.1</b>	<b>109</b>
6.1. Введение	109
6.2. Основы	109
6.2.1. Общее описание процесса	109
6.2.2. Идемпотентность	111
6.2.3. Сохранение обратной совместимости между мажорными версиями	111
6.3. Конечные точки	112
6.4. Безопасность и контроль доступа	113
6.4.1. Scopes	113
6.4.2. Grants Types	113
6.4.3. Авторизация согласия	113
6.4.4. Отмена согласия	116

6.4.5. Изменения информации о счетах . . . . .	116
6.4.6. Информация по оценке рисков . . . . .	116
6.5. Модель данных . . . . .	116
6.5.1. Использование метаданных для определения доступности транзакционного периода . . . . .	116
6.5.2. Связь со справочниками и стандартами . . . . .	117
6.5.3. Ресурсы . . . . .	117
6.5.4. Справочники . . . . .	117
6.6. Согласие на доступ к счету, спецификация API – v1.2.1 . . . . .	120
6.6.1. Конечные точки . . . . .	120
6.6.2. Модель данных . . . . .	122
6.6.3. Пример использования . . . . .	128
6.7. Счета, спецификация API – v1.2.1 . . . . .	130
6.7.1. Конечные точки . . . . .	130
6.7.2. Модель данных . . . . .	131
6.7.3. Примеры использования . . . . .	134
6.8. Баланс, спецификация API – v1.2.1 . . . . .	136
6.8.1. Конечные точки . . . . .	136
6.8.2. Модель данных . . . . .	137
6.8.3. Описание ресурсов . . . . .	137
6.8.4. Примеры использования . . . . .	140
6.9. Транзакции, спецификация API – v1.2.1 . . . . .	142
6.9.1. Конечные точки . . . . .	142
6.9.2. Модель данных . . . . .	142
6.9.3. Примеры использования . . . . .	152
6.10. Выписки, спецификация API – v1.2.1 . . . . .	156
6.10.1. Конечные точки . . . . .	156
6.10.2. Модель данных . . . . .	157
<b>7. Динамические справочники в пространствах имен – v1.2.1 . . . . .</b>	<b>163</b>
7.1. Введение . . . . .	163
7.2. Основы . . . . .	163
7.2.1. Принципы проектирования . . . . .	163
7.2.2. Управление версиями . . . . .	163
7.3. Общие справочники . . . . .	163

## 1. История изменений

Версия	Дата	Автор	Комментарий
1.0.0	21.08.2019	АФТ, Направление открытых API	Создана первая версия документа
1.0.1	27.11.2019	АФТ, Направление открытых API	Внесены минорные изменения
1.0.2	10.02.2020	АФТ, Направление открытых API	Внесены минорные изменения по замечаниям Центрального банка Российской Федерации
1.1.0	08.04.2020	АФТ, Направление открытых API	<p>Внесены изменения по замечаниям Банка России и участников Ассоциации развития финансовых технологий.</p> <p>Согласие на доступ к счету, спецификация API (раздел 6.6):</p> <ul style="list-style-type: none"> <li>• Добавлена конечная точка — GET /account-consents/{consentId}/retrieval-grant;</li> <li>• Добавлен класс — “RetrievalGrantResponse”.</li> </ul> <p>Выписки, спецификация API (раздел 6.10):</p> <ul style="list-style-type: none"> <li>• выполнена синхронизация с ресурсом “transactions” раздела “Транзакции, спецификация API”;</li> <li>• элементы “Party..” и “CounterParty..” заменены на “Debtor”, “DebtorAccount”, “DebtorAgent”, “Creditor”, “CreditorAccount” и “CreditorAgent”. В какие элементы попадает “Пользователь”, а в какие контрагент (“Debit..” или “Credit..”), определяется элементом “creditDebitIndicator”</li> </ul>
1.2.1	17.06.2020	АФТ, Направление открытых API	<p>Внесены изменения по замечаниям рабочей группы ТК122 ПКЗ.</p> <p>Добавлен динамичный справочник “OBRUErrorResponseErrorCode” для классификации низкоуровневых ошибок</p>

## 2. Введение

Настоящий Стандарт содержит описание элементов, которые являются общими для всех API на получение информации о счете клиента третьей стороной.

### 2.1. Область применения

Настоящий Стандарт рекомендован к использованию организациями при обмене финансовыми сообщениями, связанными с получением информации о банковском счете.

Настоящий Стандарт предназначен для:

- участников получения информации о банковском счете (банки и их клиенты, а также Сторонние поставщики<sup>1</sup>);
- участников перевода денежных средств (банки и их клиенты, а также Сторонние поставщики<sup>2</sup>);
- разработчиков информационного и программного обеспечения.

Положения настоящего Стандарта применяются на добровольной основе.

По предложениям участников настоящий Стандарт может дополняться принятыми в международной практике ролями и сценариями.

### 2.2. Термины и определения

В настоящем Стандарте применяются следующие термины и определения:

Наименование	Описание
API (Application Programming Interface)	Набор процедур, протоколов и инструментов для создания программных приложений. API определяет, как взаимодействуют программные компоненты
Открытые банковские интерфейсы	Общедоступные интерфейсы прикладного программирования (API), которые предоставляют разработчикам программный доступ к финансовым данным в финансовых сервисах
Пользователь	Физическое или юридическое лицо, являющееся плательщиком или получателем средств
Сторонний поставщик	Юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ)
ППУ	Кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы
СПИУ	Сторонний Поставщик Информационных Услуг о банковском счете — юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя
СППУ	Сторонний Поставщик Платежных Услуг — юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств
Среда Открытых банковских интерфейсов	Комплекс стандартов Открытых банковских интерфейсов, управление, системы, процессы, безопасность и процедуры, используемые для поддержки участников
Участники среды Открытых банковских интерфейсов	Пользователи, банки, поставщики финансовых услуг и разработчики программного обеспечения, которые участвуют в создании и развитии среды Открытых банковских интерфейсов
Плательщик	Пользователь, осуществляющий перевод денежных средств (либо от имени которого осуществляется перевод денежных средств)
Получатель средств	Пользователь, в пользу которого осуществляется перевод денежных средств
Стандарт ISO 20022	Международный стандарт обмена электронными сообщениями между организациями финансовой отрасли

<sup>1</sup> Получают доступ к информации с согласия владельца счета.

<sup>2</sup> Получают доступ к счету с согласия владельца счета.

Наименование	Описание
Многофакторная аутентификация Пользователя	Аутентификация, которая основана на использовании двух или более элементов, классифицированных как знания, владение и неотъемлемость. Эти пункты являются независимыми, поскольку нарушение одного не угрожает надежности других
Идемпотентность	Идемпотентность — это свойство объекта или операции при повторном применении операции к объекту давать тот же результат, что и при первом
Ресурс	Ресурсом является представление любой сущности (например, перевод денежных средств, счет, транзакция) в определенном формате (например, JSON). Каждый ресурс идентифицируется посредством постоянного идентификатора, который не меняется при изменении состояния ресурса
Полезная нагрузка	Часть пакета данных (сообщения) без служебной информации (без заголовка, битов синхронизации и т.п.). Детальное описание структуры полезной нагрузки находится в разделе 4.2

Таблица 1. Термины и определения

## 2.3. Принципы архитектуры

Архитектура среды Открытых банковских интерфейсов соответствует концепции RESTful API<sup>3</sup>.

Данная концепция была выбрана на основании отзывов участников рынка, а также согласно опыту мировых практик.

### 2.3.1. Идентификаторы

Ресурс REST имеет уникальный идентификатор, который используется для идентификации ресурса. Эти уникальные идентификаторы используются для создания URL-адресов для идентификации и адресации конкретных ресурсов.

### 2.3.2. Коды статусов

API используют два кода состояния, которые служат двум различным целям:

- Код состояния HTTP отражает результат вызова API (операция HTTP на ресурсе);
- Поле состояния в некоторых полезных нагрузках ресурса отражает состояние ресурсов.

<sup>3</sup> Детальное описание подхода можно посмотреть на сайте: <https://restfulapi.net/>.

## 3. Основы

### 3.1. Роли участников

Среда Открытых банковских интерфейсов определяет следующих участников:

**Пользователь** — физическое или юридическое лицо, являющееся плательщиком или получателем средств.

**Сторонний поставщик** — юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ).

**ППУ** (Поставщик Платежных Услуг) — кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы.

Среда Открытых банковских интерфейсов определяет следующие роли участников:

**СПИУ** (Сторонний Поставщик Информационных Услуг о банковском счете) — юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя.

**СППУ** (Сторонний Поставщик Платежных Услуг) — юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств.

Среда Открытых банковских интерфейсов определяет следующие взаимоотношения между участниками и ролями:

Роль \ Участник	Пользователь	ППУ	Сторонний поставщик
<b>СПИУ</b>	Нет	Да	Да
<b>СППУ</b>	Нет	Да	Да

Таблица 2. Роли участников

Открытые банковские интерфейсы регламентируют взаимодействие только между следующими участниками:

- **ППУ** предоставляет Открытые банковские интерфейсы для Стороннего поставщика. Получает сообщения запросов через свои Открытые банковские интерфейсы и отправляет соответствующие ответные сообщения Стороннему поставщику.
- **Сторонний поставщик** может получить доступ к счету Пользователя, управляемому ППУ через Открытые банковские интерфейсы, при согласии Пользователя. Сторонний поставщик отправляет сообщения запроса через Открытые банковские интерфейсы ППУ и получает соответствующие ответные сообщения от этого ППУ.

### 3.2. Кодировка символов

Запросы и ответы API используют кодировку UTF-8. Это кодировка символов по умолчанию для JSON (RFC 7158 — раздел 8.1).

Однако нисходящая система ППУ может не принимать некоторые символы UTF-8, такие как символы emoji (например, идеограммы и смайлики могут не быть приемлемой ссылкой на платеж). Если ППУ отклоняет сообщение с символом UTF-8, которое не может быть обработано, то ППУ отвечает кодом состояния HTTP 400 (неверный запрос).

### 3.3. Формат даты

ППУ принимает в запросах все действующие форматы даты стандарта ISO 8601, включая его разрешенные вариации.

Все даты в полезных нагрузках JSON представлены в формате dateTime стандарта ISO 8601. Все поля dateTime в ответах включают часовой пояс. Например:

```
2019-07-08T11:23:03+00:00
2019-05-03T18:13:23Z
```

Таблица 3. Поля dateTime с часовым поясом

Все даты в параметрах query имеют формат dateTime стандарта ISO 8601 и не включают часовой пояс. Например:

```
2019-07-01T09:23:01
2019-02-07
```

Таблица 4. Поля dateTime с обрезанием часового пояса

Все даты в заголовках HTTP представлены как полные даты RFC 7231. Пример:

```
Mon, 26 Aug 2019 14:23:51 GMT
```

Таблица 5. Формат представления полной даты

Все даты в параметрах claims JWT имеют формат number JSON, представляющего количество секунд с 1970-01-01T0:0:0Z, измеренное в GMT до текущей даты (dateTime).

```
//Mon, 26 Aug 2019 14:23:51 GMT
1987646700
```

Таблица 6. Формат представления полной даты с секундами

### 3.4. Структура пути URI-ресурса

Путь URI соответствует следующей структуре:

- [participant-path-prefix]/open-banking/[version]/[resource]/[resource-id]/[sub-resource]

Структура URI пути состоит из следующих элементов:

- [participant-path-prefix]  
Необязательный префикс ППУ.
  - open-banking  
Постоянное значение “open-banking”.
  - [version]  
Версия API, выраженная в виде /v[major-version].[minor-version]/.
  - [resource]/[resource-id]  
Наименование ресурса и его идентификатор.
  - [sub-resource]  
Наименование подресурса (ресурса 2-го уровня).
- ППУ использует один и тот же participant-path-prefix и host name для всех своих ресурсов.

Примеры:

```
https://bank.ru/oapi-channel/open-banking/v1.1/payments
https://bank.ru/oapi-channel/open-banking/v1.1/account-consents
https://bank.ru/oapi-channel/open-banking/v1.1/accounts
https://bank.ru/oapi-channel/open-banking/v1.1/accounts/1234
https://bank.ru/oapi-channel/open-banking/v1.1/accounts/1234/transactions
```

## 3.5. Заголовки сообщений (headers)

### 3.5.1. Заголовки запросов

Параметр header	Комментарий	POST-запрос	GET-запрос	DELETE-запрос	PUT-запрос
x-fapi-auth-date	Время последнего входа Пользователя в систему с TPP. Значение предоставляется в виде HTTP-date, как в разделе 7.1.1.1 [RFC7231]. Например, x-fapi-auth-date: Mon, 26 Aug 2019 12:23:11 GMT	Необязательно	Необязательно	Необязательно	Не используется
x-fapi-customer-ip-address	IP-адрес Пользователя, если Пользователь в данный момент подключен к Стороннему Поставщику (залогинен в приложении Стороннего Поставщика)	Необязательно	Необязательно	Необязательно	Не используется
x-fapi-interaction-id	RFC4122 UID, используемый в качестве идентификатора корреляции. Если необходимо, то ППУ передает обратно значение идентификатора корреляции в заголовке ответа x-fapi-interaction-id	Обязательно	Обязательно	Обязательно	Обязательно
Authorization	Стандартный заголовок HTTP. Позволяет предоставлять учетные данные серверу авторизации и (или) серверу ресурсов в зависимости от типа запрашиваемого ресурса. Для OAuth 2.0 / OIDC включает в себя Basic или Bearer схемы аутентификации	Обязательно	Обязательно	Обязательно	Обязательно
Content-Type	Стандартный заголовок HTTP. Представляет формат полезной нагрузки в запросе. Устанавливается значение application/json, за исключением конечных точек, которые поддерживают Content-Type, отличный от application/json (например, POST /file-payment-consents/{consentId}/file). Устанавливается значение application/jose+jwe для зашифрованных запросов. Сторонний Поставщик может предоставлять дополнительную информацию. Если установлено другое значение, то ППУ присылает ответ: 415 Unsupported Media Type	Обязательно	Не используется	Не используется	Обязательно
Accept	Стандартный HTTP заголовок, определяющий тип контента, который требуется от сервера.	Необязательно	Необязательно	Не используется	Необязательно

Параметр header	Комментарий	POST-запрос	GET-запрос	DELETE-запрос	PUT-запрос
	<p>Если Сторонний Поставщик ожидает незашифрованный ответ, то он указывает явно, что только ответ в формате JSON принимается (передавая значение application/json) в качестве заголовка контента для всех конечных точек, которые отвечают в формате JSON.</p> <p>Если Сторонний Поставщик ожидает зашифрованный ответ, то он указывает явно, что принимается только ответ JWT (передавая значение application/jose+jwe) в качестве заголовка контента для всех конечных точек, которые отвечают JSON.</p> <p>Для конечных точек, которые не отвечают в формате JSON (например, GET ../statements/{statementId}/file), ППУ указывает доступные параметры на своем портале для разработчиков.</p> <p>Сторонний Поставщик может предоставлять дополнительную информацию.</p> <p>Если установлено недопустимое значение, то ППУ отвечает: 406 (Not Acceptable).</p> <p>Если значение не указано, по умолчанию используется application/json</p>				
x-idempotency-key	<p>Нестандартный HTTP-заголовок. Уникальный идентификатор запроса для поддержки идемпотентности.</p> <p>Обязательно для запросов POST к конечным точкам идемпотентного ресурса.</p> <p>Для других запросов не указывается</p>	Необязательно	Не используется	Не используется	Не используется
x-jws-signature	<p>Указывает, что тело запроса подписано.</p> <p>В документации на ресурсы отдельно определяется, когда это поле в заголовке указывается.</p>	Условно (зависит от API)	Условно (зависит от API)	Условно (зависит от API)	Обязательно
x-customer-user-agent	<p>В заголовке указывается тип устройства (user-agent), который использует Пользователь.</p> <p>Сторонний Поставщик может заполнить это поле значением типа устройства (user-agent), указанным Пользователем.</p> <p>Если Пользователь использует мобильное приложение Стороннего Поставщика, Сторонний Поставщик проверяет, что строка типа устройства (user-agent) отличается от строки типа устройства (user-agent) на основе браузера</p>	Необязательно	Необязательно	Необязательно	Необязательно

Таблица 7. Заголовки запросов

Наличие или отсутствие Пользователя определяется через заголовок `x-fapi-customer-ip-address`. Если указан IP-адрес Пользователя, то предполагается, что Пользователь присутствует во время взаимодействия.

Последствием этого является то, что ППУ полагаются на информацию, предоставленную СПИУ.

### 3.5.2. Заголовки ответов

Параметр header	Комментарий	Обязательность
Content-Type	Стандартный параметр заголовка HTTP. Представляет формат полезной нагрузки, возвращаемой в ответе. ППУ возвращает значение Content-Type, равное <code>application/json</code> , в качестве заголовка для всех незашифрованных конечных точек. ППУ возвращает значение Content-Type, равное <code>application/jwe</code> , для всех зашифрованных конечных точек	Обязательно
Retry-After	Параметр заголовка, указывающий время (в секундах), в течение которого Сторонний Поставщик ожидает перед повторением операции. ППУ следует включать этот заголовок вместе с ответами с кодом состояния HTTP 429 (Too Many Requests)	Необязательно
x-fapi-interaction-id	RFC4122 UID, используемый в качестве идентификатора корреляции. ППУ заполняет параметр заголовка ответа <code>x-fapi-interaction-id</code> значением, полученным в соответствующем параметре заголовка запроса или значением UID RFC4122, если значение не было предоставлено в запросе для отслеживания взаимодействия	Обязательно
x-jws-signature	Указывает, что тело ответа подписано. В документации на ресурсы отдельно определяется, когда указывается это поле в заголовке	Условно (зависит от API)

Таблица 8. Заголовки ответов

## 3.6. Коды статусов HTTP

Ниже приведены коды ответов HTTP для различных методов HTTP всех конечных точек API.

Ситуация	HTTP статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
Запрос успешно выполнен	200 OK		нет	да	нет	да
Операция создания выполнена успешно	201 Created	Результатом операции является создание нового ресурса	да	нет	нет	нет
Операция удаления успешно завершена	204 No Content		нет	нет	да	нет
Запрос имеет неверные отсутствующее или несовместимое тело JSON, параметры URL или поля заголовка	400 Bad Request	Запрошенная операция не будет выполнена	да	да	да	да
Заголовок авторизации отсутствует или неверный токен	401 Unauthorized	Операции было отказано в доступе. Повторная аутентификация Пользователя может привести к созданию соответствующего токена, который может быть использован	да	да	да	да
Токен имеет неверную область действия или была нарушена политика безопасности	403 Forbidden	Операции было отказано в доступе. Повторная аутентификация Пользователя может привести к созданию соответствующего токена, который может быть использован	да	да	да	да
1. Сторонний Поставщик пытается получить ресурс, который указан в спецификации, но не реализован на стороне ППУ (например, ППУ решил не реализовывать конечную точку API статуса для внутренних запланированных платежей).	404 (Not Found)		да	да	да	да

Ситуация	HTTP статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
2. Сторонний Поставщик пытается получить ресурс, который не определен						
Сторонний Поставщик попытался получить доступ к ресурсу с помощью метода, который не поддерживается	405 Method Not Allowed		да	да	да	да
Запрос содержал параметр заголовка Ассерта, отличный от разрешенных media types, и набор символов, отличный от UTF-8	406 Not Acceptable		да	да	да	да
Операция была отклонена, поскольку полезная нагрузка находится в формате, не поддерживаемом этим методом на целевом ресурсе	415 Unsupported Media Type		да	нет	нет	да
Операция была отклонена, так как слишком много запросов было сделано в течение определенного периода времени	429 Too Many Requests	ППУ ограничивают запросы, если они сделаны сверх их политики добросовестного использования. ППУ документируют свои политики добросовестного использования на своих порталах для разработчиков. ППУ отвечают этим статусом, если количество запросов в единицу времени было превышено. ППУ следует включать заголовок Retry-After в ответ, указывающий, как долго Сторонний Поставщик ожидает перед повторением операции	да	да	да	да
Что-то пошло не так на стороне ППУ	500 Internal Server Error	Операция не удалась	да	да	да	да
Устаревшая версия сервиса	503 Service Unavailable	Если API устарел и больше не поддерживается ППУ, его путь URI все еще может быть активным и принимать запросы. В этом контексте рекомендуется вернуть 503 Service Unavailable, чтобы TPP знал, что версия API находится в offline режиме	да	да	да	да

Таблица 9. Коды статусов HTTP

ППУ возвращают другие стандартные коды состояния HTTP (например, от шлюзов и других периферийных устройств), как описано в RFC 7231 — раздел 6.

ППУ отвечают ошибкой в потоке OAuth / OIDC с обязательным выравниванием кодов ошибок, указанных в разделе 3.1.2.6 OpenID Connect Core Specification.

ППУ отвечают на все некорректные запросы общей структурой ошибок Открытых банковских интерфейсов.

### 3.6.1. 400 (Bad Request) или 404 (Not Found)

Если Сторонний Поставщик пытается запросить URL ресурса с идентификатором ресурса, который не существует, то ППУ отвечает 400 (неверный запрос), а не 404 (не найдено).

Например, если Сторонний Поставщик пытается выполнить запрос GET/payment/22289, где 22289 не является действительным paymentId, ППУ отвечает 400.

Если Сторонний Поставщик пытается получить доступ к URL-адресу ресурса, который не определен этими спецификациями (например, GET /card-accounts), то ППУ отвечает 404 (Not Found).

Если ППУ не реализовал конечную точку API, то он отвечает 404 (не найдено) для запросов к этому URL.

Таблица ниже иллюстрирует некоторые примеры предсказуемого поведения:

Ситуация	Запрос	Ответ
Сторонний Поставщик пытается получить платеж с несуществующим идентификатором paymentId	GET /payments/1001	400 (Bad Request)
Сторонний Поставщик пытается получить ресурс, который указан в спецификации, но не реализован на стороне ППУ. Например, ППУ решил не реализовывать конечную точку API для получения транзакций по счету	GET /accounts/{accountId}/transactions	404 (Not Found)
Сторонний Поставщик пытается получить ресурс, который не определен	GET /bulk	404 (Not Found)

Таблица 10. Возможные ситуации для ответов 400 (Bad Request) и 404 (Not Found)

### 3.6.2. 403 (Forbidden)

Когда Сторонний Поставщик пытается получить доступ к ресурсу, к которому у него нет разрешения, ППУ возвращает 403 (Forbidden) с необязательным телом ответа об ошибке.

Ситуация возможна в следующих случаях:

- Сторонний Поставщик использует токен доступа (access-token), который не имеет соответствующего scope для доступа к запрошенному ресурсу.
- Сторонний Поставщик попытался получить доступ к ресурсу с идентификатором, к которому у него нет доступа. Например, попытка получить доступ к GET /payments/1001, где платежный ресурс с идентификатором 1001 принадлежит другому Стороннему Поставщику.
- Сторонний Поставщик пытается получить доступ к ресурсу транзакции, а у Стороннего Поставщика нет согласия на авторизацию с правами доступа к запрашиваемому ресурсу.
- Сторонний Поставщик пытается получить доступ к ресурсу транзакции, а у Стороннего Поставщика нет согласия на авторизацию для accountId. Например, попытка получить доступ к GET /accounts/2001 или GET /accounts/2001/transactions, когда Пользователь не выбрал accountId 2001 для авторизации.
- Сторонний Поставщик пытается получить доступ к ресурсу, а ППУ решает повторно аутентифицировать Пользователя. ППУ отвечает соответствующим кодом ошибки, чтобы указать, что требуется повторная аутентификация.

### 3.6.3. 401 (Unauthorized)

Когда Сторонний Поставщик использует токен доступа с истекшим сроком, ППУ возвращает 401 (Unauthorized) без какого-либо сообщения об ошибке.

Ситуация возникает, если ППУ завершил срок действия токена доступа по любой из следующих причин:

1. Истек срок действия согласия.
2. Подозрительное использование токена доступа или подозрение в мошенничестве.
3. Плановая реализация многофакторной аутентификации.

Эта ошибка потенциально может быть исправлена, если Пользователь повторно пройдет аутентификацию или аутентифицируется с соответствующими разрешениями.

### 3.6.4. 429 (Too Many Requests)

Если Сторонний Поставщик пытается получить доступ к ресурсу слишком часто, то ППУ может вернуть 429 (Too Many Requests). Это нефункциональное требование, отдельные ППУ определяют метрику запросов в единицу времени.

Ситуация возникает, когда:

- Сторонний Поставщик решает реализовать функцию “Статус перевода денежных средств в реальном времени” для своих Пользователей и делает это некорректно, опрашивая конечную точку методом GET.
- Сторонний Поставщик решает использовать конечную точку для одиночного перевода денежных средств, как если бы она была конечной точкой пакетной оплаты, и отправляет большое количество запросов на оплату в очень короткий промежуток времени, так что это превышает политику использования ППУ.

## 3.7. Идемпотентность

Ключ идемпотентности используется для защиты от создания дубликатов ресурсов при использовании метода POST для конечных точек API.

Если для конечной точки API требуется ключ идемпотентности:

- Параметр заголовка x-idempotency-key содержит не более 40 символов. Если длина поля превышает 40 символов, то ППУ отклоняет запрос с кодом состояния 400 (Bad Request).

- Сторонний Поставщик не меняет тело запроса при использовании одинакового ключа `x-idempotency-key`. Если Сторонний Поставщик изменяет тело запроса, то ППУ не меняет конечный ресурс. ППУ рассматривает это как мошенническое действие.
- ППУ обрабатывает запрос как идемпотентный, если он получил запрос с существующим параметром `x-idempotency-key` от того же Стороннего Поставщика в течение 24 часов.
- ППУ не создает новый ресурс для запроса `POST`, если он определен как идемпотентный запрос.
- ППУ отвечает на запрос текущим статусом ресурса (или статусом, максимально близким к текущему, который можно получить в данный момент времени на существующем онлайн-канале) и кодом статуса `HTTP 201 (Created)`.
- Сторонний Поставщик не использует ключ идемпотентности при опросе состояния ресурсов.
- ППУ использует подпись сообщения вместе с ключом идемпотентности, чтобы гарантировать, что тело запроса не изменилось.

Если ключ идемпотентности не требуется для конечной точки API, но содержится в запросе, то ППУ игнорирует ключ идемпотентности.

### 3.8. Фильтрация

ППУ обеспечивает ограниченную поддержку фильтрации для операций `GET`, которые возвращают множественные записи.

Параметры фильтра всегда разные для конкретного поля (полей) ресурса и следуют правилам/форматам, определенным в справочниках для ресурса.

Для параметров фильтра типа `DateTime` значения соответствуют стандарту ISO 8601. Если значение поля типа `DateTime` содержит часовой пояс, то ППУ игнорирует эту информацию.

Предполагается, что значения фильтра относятся к тому же часовому поясу, что и часовой пояс, в котором поддерживается ресурс.

### 3.9. Нумерация страниц

ППУ предоставляет постраничный ответ для операций `GET`, которые возвращают множественные записи.

В такой ситуации ППУ:

- Если существует следующая страница записей ресурсов, то ППУ предоставляет ссылку на следующую страницу ресурсов в поле `Links.Next` ответа. Отсутствие следующей ссылки будет означать, что текущая страница является последней страницей результатов.
- Если предыдущая страница записей ресурсов существует, то ППУ предоставляет ссылку на предыдущую страницу ресурсов в поле `Links.Prev` ответа. Отсутствие предыдущей ссылки указывает на то, что текущая страница является первой страницей результатов.

Для разбитых на страницы ответов ППУ гарантирует, что количество записей на странице находится в разумных пределах, минимум 25 записей (кроме последней страницы, где больше нет записей) и максимум 1000 записей.

Дополнительно ППУ предоставляет:

- Ссылку на первую страницу результатов в поле `Links.First`.
- Ссылку на последнюю страницу результатов в поле `Links.Last`.
- Общее количество страниц в поле `Meta.TotalPages`.

ППУ включает `self` ссылку на ресурс в поле `Links.Self`, как описано в разделе “Ссылки”.

Этот стандарт не определяет, каким образом параметры перелистывания страниц передаются ППУ, и каждый ППУ может использовать свои собственные механизмы для разбивки ответа.

Если исходный запрос от СПИУ включал параметры фильтра, то в ответе возвращаются только те результаты (разбитые на страницы), которые соответствуют фильтру.

### 3.10. Архивирование

Архивация ресурсов будет определяться для ППУ на основе их внутренних требований.

В дополнение стоит отметить, что:

- ППУ удаляют просроченные идентификаторы согласия (`consentId`) только через 24 часа после создания.
- ППУ могут архивировать просроченные идентификаторы согласия.

### 3.11. Дополнительные данные

Ряд ресурсов в спецификации включает в себя раздел для дополнительных данных (Supplementary Data). Данный раздел позволит ППУ принимать или передавать данные, которые не предусмотрены основной структурой ресурса.

Раздел дополнительных данных определяется как пустой объект JSON в данной спецификации.

Если ППУ использует структуру с дополнительными данными (Supplementary Data), то он выкладывает детальное описание структуры у себя на портале для разработчиков.

ППУ не используют структуру Supplementary Data, если добавляемый туда элемент уже существует в текущей версии документа Открытых банковских интерфейсов.

## 4. Модель данных

### 4.1. Справочники и перечисления

Спецификации Открытых банковских интерфейсов содержат поля со справочными данными.

Справочники бывают двух видов:

- фиксированные;
- гибкие.

В случае фиксированных справочников — все возможные значения будут задаваться статично в стандарте Открытых банковских интерфейсов.

В случае гибких справочников — начальные возможные значения будут задаваться в стандарте Открытых банковских интерфейсов, но каждый ППУ может динамично управлять ими, расширяя своими кастомизированными значениями. Кастомизированные значения, которые используют ППУ, могут в будущем включаться в список начальных возможных значений для гибких справочников.

И фиксированные, и гибкие справочники находятся в разделе “Справочники и перечисления”.

### 4.2. Общая структура полезной нагрузки

В этом разделе приводится обзор структуры верхнего уровня для полезных нагрузок Открытых банковских интерфейсов.

Данные, которые содержатся в разделе “Data”, документируются для каждой отдельно взятой конечной точки API.

#### 4.2.1. Структура запроса

Структура верхнего уровня для запросов Открытых банковских интерфейсов имеет следующий вид:

```
{
  "Data": {
    ...
  },
  "Risk": {
    ...
  }
}
```

Таблица 11. Верхнеуровневая структура сообщений для запросов

#### **Data**

Раздел “Data” содержит данные запроса для конкретного запроса API.

Структура этого элемента отличается для каждой конечной точки API.

#### **Risk**

Раздел “Risk” содержит индикаторы риска для конкретного запроса API, предоставленного Сторонним Поставщиком.

Индикаторы риска, содержащиеся в этом элементе, могут отличаться для каждой конечной точки API.

### 4.2.2. Структура ответа

Структура верхнего уровня для ответов Открытых банковских интерфейсов имеет следующий вид:

```
{
  "Data": {
    ...
  },
  "Risk": {
    ...
  },
  "Links": {
    ...
  },
  "Meta": {
    ...
  }
}
```

Таблица 12. Верхнеуровневая структура сообщений для ответов

В соответствии с принципом API RESTful полный ресурс воспроизводится как часть ответа.

В ответ включаются следующие дополнительные разделы высокого уровня:

- Links
- Meta

### 4.2.3. Структура ответа с ошибками

Структура для ответов с ошибками Открытых банковских интерфейсов имеет следующий вид:

```
{
  "code": "...",
  "id": "...",
  "message": "...",
  "Errors": [
    {
      "errorCode": "...",
      "message": "...",
      "path": "...",
      "url": "..."
    }
  ]
}
```

Таблица 13. Структура сообщений для ответов с ошибками

#### UML-диаграмма

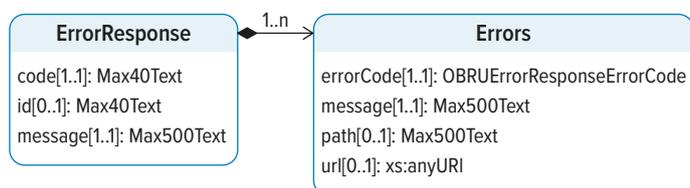


Рисунок 1. UML-диаграмма для структуры ответов с ошибками

## Спецификация данных

Наименование	Кратность	XPath	Подробное описание	Тип данных	Значения	Шаблон
OBRUErrorResponse		OBRUErrorResponse	Массив подробных кодов ошибок, сообщений и URL-адресов к документации для помощи в исправлении	OBRUErrorResponse		
code	1..1	OBRUErrorResponse/Code	Высокоуровневый текстовый код ошибки, необходимый для классификации	Max40Text		
id	0..1	OBRUErrorResponse/Id	Уникальный идентификатор ошибки для целей аудита в случае неизвестных / неклассифицированных ошибок	Max40Text		
message	1..1	OBRUErrorResponse/Message	Краткое сообщение об ошибке. Например, "что-то не так с предоставленными параметрами запроса"	Max500Text		
Errors	1..n	OBRUErrorResponse/Errors		OBRUError		
errorCode	1..1	OBRUErrorResponse/Errors/ErrorCode	Низкоуровневое текстовое описание ошибки. Например, RU.SBRF.Field.Missing	OBRUErrorResponse/ErrorCode		
message	1..1	OBRUErrorResponse/Errors/Message	Описание ошибки. Например, "Обязательное поле не указано".	Max500Text		
path	0..1	OBRUErrorResponse/Errors/Path	Путь к элементу с ошибкой в JSON объекте. Рекомендуемое, но не обязательное поле	Max500Text		
url	0..1	OBRUErrorResponse/Errors/Url	URL для помощи в устранении проблемы. Также через URL можно предоставлять дополнительную информацию	xs:anyURI		

Таблица 14. Детальное описание элементов ответа с ошибками

## 4.2.4. Необязательные поля

В объектах, где значение для необязательного поля не указано, поле исключается из полезной нагрузки JSON.

В объектах, где поле массива определено как имеющее значения 0..n, поле массива включается в полезную нагрузку с пустым массивом.

```
{
  "name": "", // Неправильно. Поле "Name" нужно исключить из полезной
  "age": 0, // Неправильно. Значение "0" не используется для
  "creditorAccount": {}, // Неправильно. Поле "CreditorAccount" нужно исключить.
  "balances": [] // Правильно. Таким образом передается пустой массив.
}
```

Таблица 15. Примеры передачи необязательных полей

#### 4.2.5. Ссылки

Раздел “Links” является обязательным и всегда будет содержать абсолютные URI для связанных ресурсов.

Ссылка “self” является обязательной.

```
"Links": {  
  "self": "https://api.bank.ru/open-banking/v3.1/payments/58923"  
}
```

Таблица 16. Пример передачи одинарной ссылки “Self”

При передаче большого количества данных раздел Links может также содержать элементы First, Prev, Next и Last.

```
"Links": {  
  "self": "http://rocks.ru/articles?page[number]=3&page[size]=25",  
  "first": "http://rocks.ru/articles?page[number]=1&page[size]=25",  
  "prev": "http://rocks.ru/articles?page[number]=2&page[size]=25",  
  "next": "http://rocks.ru/articles?page[number]=4&page[size]=25",  
  "last": "http://rocks.ru/articles?page[number]=6&page[size]=25"  
}
```

Таблица 17. Пример передачи всех элементов раздела Links

#### 4.2.6. Метаданные

Раздел Meta обязателен, но может быть пустым. Необязательный элемент TotalPages, указывает на количество передаваемых страниц. Если передается более одной страницы, то элемент TotalPages обязательно присутствует.

```
"Meta": {  
  "totalPages": 6  
}
```

Таблица 18. Пример передачи раздела Meta

## 5. Примеры использования

Примеры использования для отдельных API задокументированы на соответствующих страницах. В этом разделе приведены примеры использования некоторых шаблонов.

### 5.1. Поток с нумерацией страниц

Приведенный ниже пример иллюстрирует, как ППУ может возвращать многостраничный ответ.

```
GET /accounts/34566/transactions HTTP/1.1
Authorization: Bearer Az567A0Jtyue
x-fapi-auth-date: Mon, 2 Sep 2019 12:33:12 GMT
x-fapi-customer-ip-address: 10.5.412.45
x-fapi-interaction-id: 11bac543-d5de-3446-b687-880a5018434d
Accept: application/json
```

Таблица 19. Запрос списка транзакций

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 11bac543-d5de-3446-b687-880a5018434d
Content-Type: application/json
{
  "Data": {
    ...
  },
  "Links": {
    "self": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions/",
    "last": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions?pg=6",
    "first": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions/",
    "next": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions?pg=2"
  },
  "Meta": {
    "totalPages": 6,
    "firstAvailableDateTime": "2019-05-03T00:00:00+00:00",
    "lastAvailableDateTime": "2019-12-03T00:00:00+00:00"
  }
}
```

Таблица 20. Пример передачи всех элементов раздела Links

## 6. Счета и транзакции, спецификация API – v1.2.1

### 6.1. Введение

В данном разделе описываются потоки данных и полезные нагрузки для Открытых банковских интерфейсов по счетам и транзакциям.

Описанные здесь конечные точки API позволяют СПИУ:

- Создать согласие на доступ к информации о счете. Пользователь предоставляет доступ СПИУ к своим данным в ППУ. Для этого Пользователь:
  - Выбирает, к каким данным он готов предоставить доступ, проставив разрешения (permissions);
  - Выбирает срок действия разрешений на использование данных;
  - Выбирает даты начала и окончания, которые будут использоваться при построении отчетов по спискам транзакций и выпискам.
- Получать данные по счетам и транзакциям.

### 6.2. Основы

В данном разделе описываются потоки данных с позиции бизнес-процессов. Детальное описание потоков данных находится в стандарте “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.

#### 6.2.1. Общее описание процесса

##### 6.2.1.1. Пошаговое описание

Шаг 1 — Запрос информации о счете:

- Этот поток начинается с согласия Пользователя на предоставление СПИУ доступа к информации о счете.

Шаг 2 — Настройка согласия на доступ к информации о счете:

- Между СПИУ и сервером авторизации ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
- СПИУ с помощью потока Client Credentials Grant получает на сервере авторизации ППУ токен доступа (access-token).
- Между СПИУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
- СПИУ подключается к ППУ, который обслуживает счет (счета) Пользователя и создает ресурс согласия на доступ к информации о счете. ППУ сообщает о том, что один из его Пользователей предоставляет доступ к информации о счете и информации о транзакциях для СПИУ. ППУ отвечает идентификатором ресурса (consentId). Этот шаг выполняется с помощью запроса POST к конечной точке /account-consents.
- Ресурс согласия на доступ к информации о счете будет включать в себя следующие поля, которые описывают данные, на которые Пользователь дал согласие для СПИУ:
  - Permissions — список кластеров данных, которые были разрешены для доступа.
  - Expiration Date — обязательное поле, означающее срок действия доступа СПИУ к данным Пользователя.
  - Transaction Validity Period — обязательное поле, означающее диапазон дат, который определяет период для транзакций и выписок, к которым СПИУ может получить доступ.
- СПИУ может быть посредником для передачи данных другим сторонам, поэтому для Пользователя допустимо иметь несколько account-consents — ресурсов на доступ к информации о счете для одного и того же счета с разными параметрами согласия и (или) авторизации.

### Шаг 3 — Авторизация согласия:

- СПИУ запрашивает у Пользователя авторизацию согласия. ППУ выполняет это с использованием потока перенаправления (redirection flow):
  - СПИУ перенаправляет Пользователя на страницу к ППУ.
  - Перенаправление содержит consentId, созданный на предыдущем шаге, что позволяет ППУ сравнить с идентификатором имеющегося у него ресурса согласия на доступ к информации о счете (account-consent).
  - ППУ аутентифицирует Пользователя.
  - ППУ обновляет статус ресурса согласия на доступ к информации о счете (account-consent), фиксируя, что согласие было авторизовано.
  - Как только согласие было авторизовано, Пользователь перенаправляется обратно на сторону СПИУ с кодом авторизации (authorization-code).
  - Между СПИУ и сервером авторизации ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
  - СПИУ обменивает на сервере авторизации ППУ код авторизации (authorization-code) на токен доступа (access-token).
- Между Пользователем и СПИУ осуществляется управление согласием, поэтому на этом этапе не могут изменяться детали согласия на доступ к информации о счете (с ППУ). Пользователь может только полностью авторизовать или отклонить данные о согласии на доступ к информации о счете.
- Во время авторизации Пользователь выбирает счета, которые авторизованы для запросов СПИУ (в интерфейсе ППУ).

### Шаг 4 — Запрос данных:

- Между СПИУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта “Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID”.
- СПИУ выполняет запрос данных с помощью запроса GET для соответствующего ресурса.
- Уникальные accountId(s), которые соответствуют согласию на доступ к информации о счетах (account-consent), будут возвращены при вызове GET /accounts. Это всегда будет первый вызов при наличии у СПИУ действующего токена доступа (access-token).

### 6.2.1.2. Диаграмма последовательности

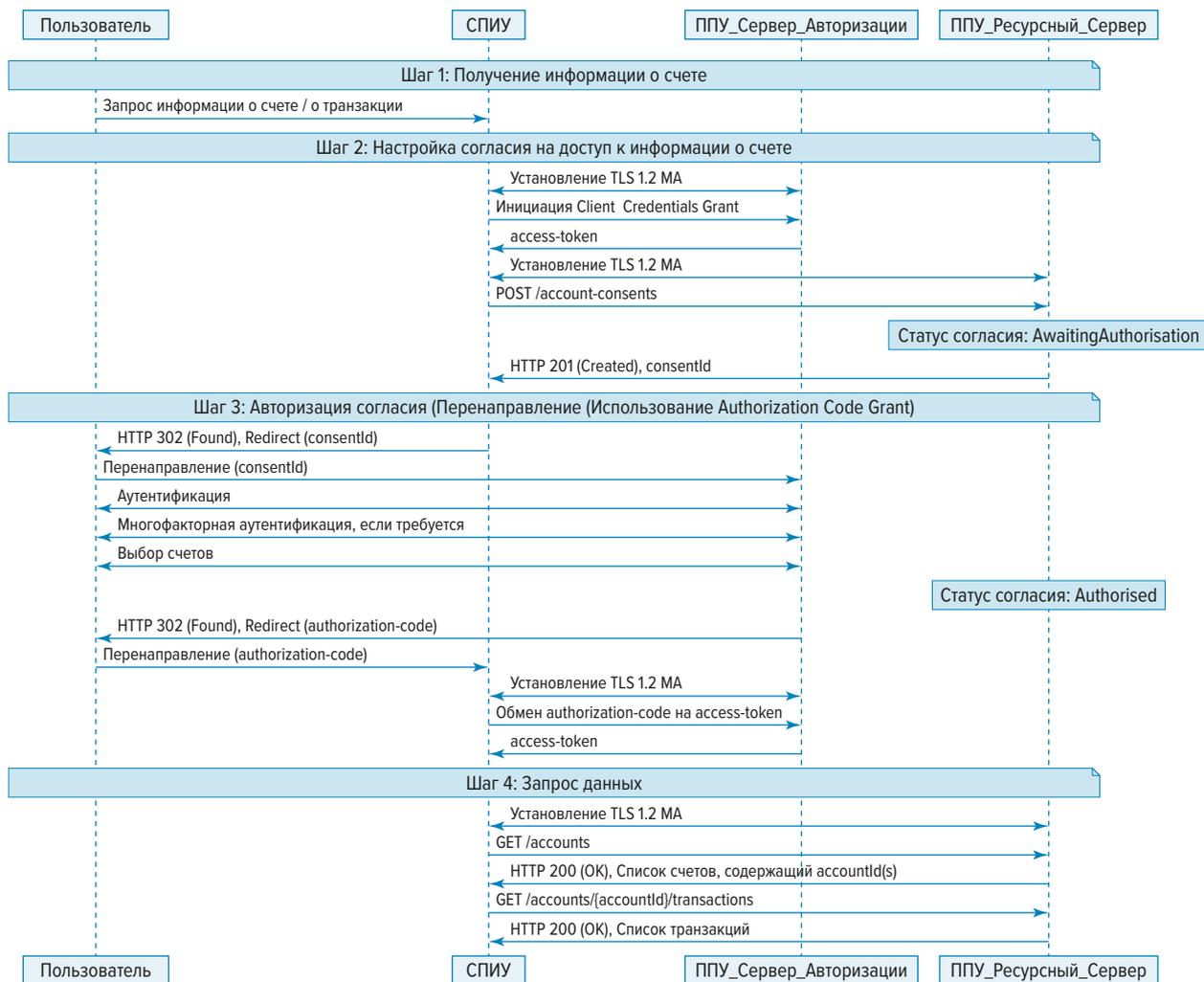


Рисунок 2. Диаграмма последовательности для получения информации о счете

### 6.2.2. Идемпотентность

Конечные точки Открытых банковских интерфейсов, предназначенные для создания согласия на доступ к информации о счете/транзакциях, не являются идемпотентными.

Если возникает ошибка тайм-аута, то ППУ создает новый ресурс согласия на доступ к информации о счете, а не тот же ресурс.

### 6.2.3. Сохранение обратной совместимости между мажорными версиями

В этом разделе дается обзор принципов сохранения обратной совместимости между версиями Открытых банковских интерфейсов на получение информации о счете/транзакциях.

#### 6.2.3.1. Ресурсы согласия на доступ к информации о счете

##### 6.2.3.1.1. POST

- СПИУ не создает согласие в более новой версии спецификации и не использует его для конечных точек предыдущей версии спецификации
  - То есть идентификатор consentId для созданного ресурса account-consent во 2-й версии не используется для доступа к конечным точкам 1-й версии.

##### 6.2.3.1.2. GET

- СПИУ не получает доступ к ресурсу согласия в более старой версии через Id для ресурса согласия, созданного в более новой версии:
- ППУ разрешает доступ к ресурсу согласия в более новой версии.

- ППУ гарантирует, что набор разрешений (permissions), связанный с ресурсом согласия, не изменяется при доступе в другой версии.
- ППУ гарантирует инвариантность полей ресурса согласия от версии API.
- ППУ разрешает доступ к ресурсам согласия с истекшей датой действия в более новой версии.
- ППУ выбирает заполнение новых полей, добавленных в ресурсе, из значений по умолчанию предыдущей версии (если установлена обязательность) или не заполняет (если обязательность не установлена).

#### 6.2.3.1.3. DELETE

- СПИУ не удаляет ресурс согласия в более старой версии с помощью идентификатора согласия, созданного в более новой версии:
  - То есть, если ресурс согласия создан во 2-й версии, а запрос на удаление осуществляется в 1-й версии конечной точки, то удаление не происходит.
- ППУ поддерживает удаление ресурса согласия из предыдущей версии через конечные точки более новой версии:
  - Например, ресурс согласия был создан в 1-й версии, а его удаление происходит через конечную точку 2-й версии.

#### 6.2.3.2. Ресурсы для предоставления информации о счете/транзакциях

##### 6.2.3.2.1. GET

- СПИУ может использовать токен, связанный с ресурсом согласия из предыдущей версии, для доступа к конечной точке более новой версии.
- СПИУ может использовать идентификатор ресурса согласия, созданного в предыдущей версии, для получения информации о счете в более новой версии.
- СПИУ не использует идентификатор ресурса согласия из более новой версии для доступа к ресурсам информации о счете в предыдущей версии.
- СПИУ не использует идентификатор ресурса согласия из предыдущей версии для доступа к ресурсу, представленному в более новой версии (согласие не будет иметь разрешений, необходимых для доступа к новому ресурсу).
- ППУ разрешает СПИУ использовать идентификатор ресурса согласия из предыдущей версии для доступа к конечным точкам ресурса информации о счете в более новой версии.
- ППУ отклоняет запрос на доступ к ресурсу, для которого набор разрешений согласия не соответствует.
- ППУ выбирает заполнение новых полей, добавленных в ресурсе, из значений по умолчанию из предыдущей версии (если установлена обязательность) или не заполняет.

## 6.3. Конечные точки

В этом разделе рассматривается список доступных конечных точек Открытых банковских интерфейсов для доступа к информации о счете/транзакциях.

Наличие ресурсов 2-го уровня (например, balances, transactions) поможет управлять этими ресурсами (у каждого ресурса 2-го уровня есть уникальный идентификатор).

Ссылка	Ресурс	Конечная точка
Согласие на доступ к счету, спецификация API — v1.2.1	account-consents	POST /account-consents GET /account-consents/{consentId} DELETE /account-consents/{consentId} GET /account-consents/{consentId}/retrieval-grant
Счета, спецификация API — v1.2.1	accounts	GET /accounts GET /accounts/{accountId}
Баланс, спецификация API — v1.2.1	balances	GET /accounts/{accountId}/balances GET /balances
Транзакции, спецификация API — v1.2.1	transactions	GET /accounts/{accountId}/transactions GET /transactions
Выписки, спецификация API — v1.2.1	statements	POST /statements/{accountId} GET /accounts/{accountId}/statements/{statementId} GET /statements

Таблица 21. Конечные точки для получения информации о счете

## 6.4. Безопасность и контроль доступа

### 6.4.1. Scopes

Токен доступа, требуемый для доступа к информации о счете, имеет как минимум следующий scope:

```
accounts: Возможность чтения информации о счете
```

Таблица 22. Score для получения информации о счете

### 6.4.2. Grants Types

СПИУ использует поток Client Credentials Grant для получения токена доступа к ресурсу account-consents. В спецификациях этот тип предоставления называется “Client Credentials”.

СПИУ использует Authorization Code Grant с потоком перенаправления (redirect flow) или с потоком разъединения (decoupled flow) для получения токена доступа ко всем другим ресурсам (кроме согласия). В спецификации этот тип предоставления называется “Authorization Code”.

### 6.4.3. Авторизация согласия

СПИУ создает ресурс account-consent с помощью POST операции. Пока данный ресурс не будет авторизован Пользователем, его создание указывает лишь на факт, что между Пользователем и СПИУ есть предварительная договоренность на использование данных, которыми Пользователь обладает в ППУ (с определенными permissions).

При создании ресурса account-consent в ответе ППУ содержится идентификатор ресурса consentId, поскольку данный идентификатор будет использоваться в дальнейшем для авторизации Пользователя (верификации permissions).

В рамках процесса авторизации согласия:

- ППУ аутентифицирует Пользователя.
  - ППУ передает согласие, зарегистрированное на стороне СПИУ, Пользователю для получения от него авторизации напрямую.
  - Пользователь может подтвердить или отклонить согласие.
  - ППУ предоставляет Пользователю список счетов, который соответствует согласию Пользователя.
- После этого ресурс согласия является авторизованным Пользователем.

#### 6.4.3.1. Элементы согласия

Ресурс account-consent состоит из следующих элементов:

- **Permissions:** Набор кластеров данных, к которым СПИУ имеет доступ после согласия Пользователя.
- **ExpirationDateTime:** Дата, до которой согласие действительно. Срок действия согласия.
- **TransactionFromDateTime:** Начальная дата, с которой СПИУ имеет доступ к транзакциям Пользователя на стороне ППУ.
- **TransactionToDateTime:** Конечная дата, до которой СПИУ имеет доступ к транзакциям Пользователя на стороне ППУ.

##### 6.4.3.1.1. Permissions (разрешения)

Коды permissions будут использоваться для ограничения данных, возвращаемых в ответ на запрос ресурса.

Разрешения (permissions) бывают базовыми и детальными. Если Пользователь дал доступ к детальным разрешениям, то соответствующие базовые разрешения считаются доступными по умолчанию.

Массив разрешений состоит как минимум из ReadAccountsBasic или ReadAccountsDetail разрешений.

Следующие комбинации разрешений являются недопустимыми, и ППУ отклонит такие ресурсы account-consents с 400 кодом в ответе:

- Согласие содержит пустой массив разрешений. Если ППУ не получил ни одного permission, то такой запрос отклоняется.
- Согласие содержит код permission, который не поддерживается ППУ (ожидается, что ППУ опубликуют, какие конечные точки API поддерживаются).
- Массив разрешений в согласии содержит ReadTransactionsBasic, но не содержит ни ReadTransactionsCredits, ни ReadTransactionsDebits.
- Массив разрешений в согласии содержит ReadTransactionsDetail, но не содержит ни ReadTransactionsCredits, ни ReadTransactionsDebits.

- Массив разрешений в согласии содержит ReadTransactionsCredits, но не содержит ни ReadTransactionsBasic, ни ReadTransactionsDetail.
- Массив разрешений в согласии содержит ReadTransactionsDebits, но не содержит ни ReadTransactionsBasic, ни ReadTransactionsDetail.

Разрешение (permission)	Конечная точка	Бизнес-логика	Описание кластера данных
ReadAccountsBasic	/accounts /accounts/{accountId}		Возможность чтения основной информации о счете
ReadAccountsDetail	/accounts /accounts/{accountId}	Доступ к дополнительным элементам в полезной нагрузке	Возможность чтения детальной информации о счете
ReadBalances	/balances /accounts/{accountId}/balances		Возможность чтения информации о балансе счета
ReadTransactionsBasic	/transactions /accounts/{accountId}/transactions	Массив разрешений также включает один из permission: <ul style="list-style-type: none"> <li>• ReadTransactionsCredits</li> <li>• ReadTransactionsDebits</li> </ul>	Возможность чтения основной информации о транзакциях
ReadTransactionsDetail	/transactions /accounts/{accountId}/transactions	Доступ к дополнительным элементам в полезной нагрузке. Массив разрешений также включает один из permission: <ul style="list-style-type: none"> <li>• ReadTransactionsCredits</li> <li>• ReadTransactionsDebits</li> </ul>	Возможность чтения детальной информации о транзакциях
ReadTransactionsCredits	/transactions /accounts/{accountId}/transactions	Доступ к кредитовым транзакциям. Массив разрешений также включает один из permission: <ul style="list-style-type: none"> <li>• ReadTransactionsBasic</li> <li>• ReadTransactionsDetail</li> </ul>	Возможность чтения только кредитовых транзакций
ReadTransactionsDebits	/transactions /accounts/{accountId}/transactions	Доступ к дебетовым транзакциям. Массив разрешений также включает один из permission: <ul style="list-style-type: none"> <li>• ReadTransactionsBasic</li> <li>• ReadTransactionsDetail</li> </ul>	Возможность чтения только дебетовых транзакций

Таблица 23. Базовые разрешения на доступ к кластерам данных

#### 6.4.3.1.1.1. Детальные разрешения (permissions)

В этом разделе перечислены разрешения, которые являются детальными. Все остальные разрешения являются базовыми (основными).

Код детального разрешения	Наименование элемента данных (комплексного типа)	Кратность	Путь к элементу (XPath)
ReadAccountsDetail	Account	0..1	Account/Data/Account/Account
ReadAccountsDetail	Servicer	0..1	Account/Data/Account/Servicer
ReadTransactionsDetail	TransactionInformation	0..1	Transaction/Data/Transaction/TransactionInformation
ReadTransactionsDetail	Balance	0..1	Transaction/Data/Transaction/Balance
ReadTransactionsDetail	MerchantDetails	0..1	Transaction/Data/Transaction/MerchantDetails
ReadTransactionsDetail	CreditorAgent	0..1	Transaction/Data/Transaction/CreditorAgent
ReadTransactionsDetail	CreditorAccount	0..1	Transaction/Data/Transaction/CreditorAccount
ReadTransactionsDetail	DebtorAgent	0..1	Transaction/Data/Transaction/DebtorAgent
ReadTransactionsDetail	DebtorAccount	0..1	Transaction/Data/Transaction/DebtorAccount

Таблица 24. Детальные разрешения на доступ к кластерам данных

Пример применения разрешений (permission) для кодов ReadAccountsBasic и ReadAccountsDetail выглядит следующим образом:

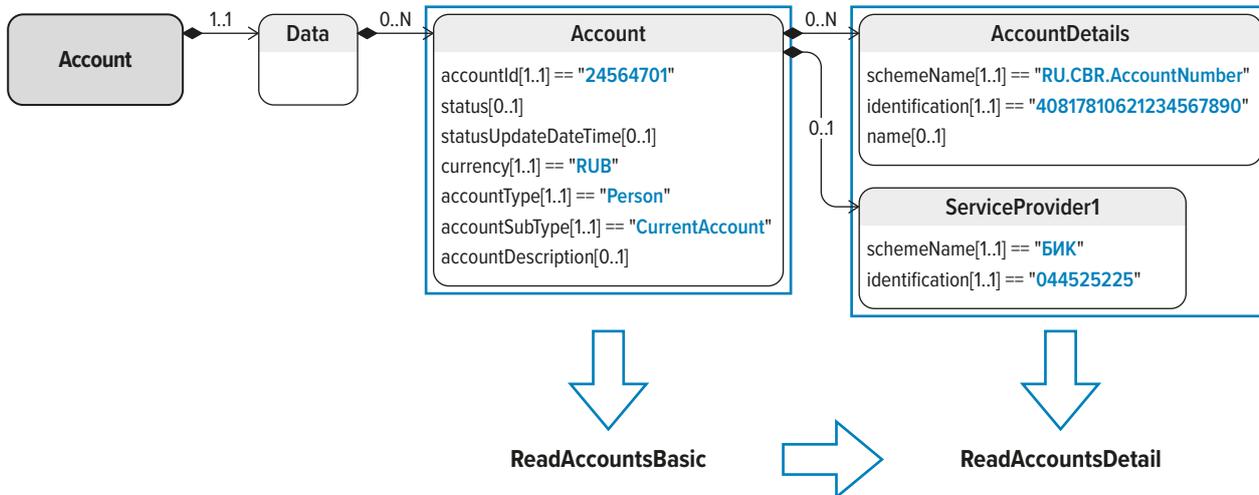


Рисунок 3. Пример применения базового и детального разрешений на доступ к кластерам данных

**6.4.3.1.2. Отмененные записи**

Если Пользователь дал разрешение ReadTransactionsCredits, то ППУ включает все кредитовые транзакции, включая отмененные дебетовые.

Если Пользователь дал разрешение ReadTransactionsDebits, то ППУ включает все дебетовые транзакции, включая отмененные кредитовые.

**6.4.3.1.2. Срок действия согласия**

ExpirationDateTime является необязательным полем, которое определяет срок истечения действия согласия для СПИУ на доступ к данным Пользователя в ППУ.

Поле является необязательным, поскольку согласие СПИУ на доступ к данным Пользователя может быть бессрочным. На данный момент под бессрочным доступом понимается 90 дней.

ExpirationDateTime применяется ко всем разрешениям (кластерам данных), которые находятся в массиве ресурса согласия при его авторизации.

**6.4.3.1.3. Начальная/конечная дата доступа к транзакциям**

TransactionToDateTime и TransactionFromDateTime определяют согласие Пользователя на доступ к транзакциям в определенный период времени. Оба поля являются необязательными, и одно поле может быть указано без другого.

ППУ может ограничить доступ к транзакциям за рамками указанного временного периода при запросах к ресурсу transactions.

**6.4.3.2. Статус согласия на доступ к информации о счетах**

Ресурс согласия на доступ к информации о счете имеет один из следующих кодов состояния после авторизации:

N°	Status	Description
1	Authorised	Согласие на доступ к информации о счете было успешно авторизовано
2	Rejected	Согласие на доступ к информации о счете было отклонено
3	Revoked	Согласие на доступ к информации о счете было отозвано через интерфейс ППУ

Таблица 25. Статус согласия на доступ к информации о счете

**6.4.3.3. Повторная аутентификация согласия**

Согласие на доступ к информации о счете является долгосрочным согласием.

Пользователь может повторно авторизовать согласие на доступ к информации о счете, если выполняются оба условия:

- Согласие имеет статус Authorised.
  - Указан и не истек срок действия согласия (ExpirationDateTime).
- Счета, связанные с ресурсом согласия, выбираются на стороне ППУ.

ППУ может позволить Пользователю изменить выбранные счета при повторной аутентификации.

#### 6.4.4. Отмена согласия

Пользователь может отозвать согласие на доступ к информации о счете в любой момент времени.

Пользователь может отозвать авторизацию ресурса согласия напрямую на стороне ППУ. Механизмы для этого процесса для каждого отдельно взятого ППУ могут отличаться (реализация на стороне ППУ). Если Пользователь отозвал авторизацию согласия на доступ к информации о счете на стороне ППУ, то статус ресурса account-consents меняется на Revoked.

Пользователь может потребовать от СПИУ отозвать согласие. Если согласие будет отозвано на стороне СПИУ:

- СПИУ перестает обращаться к этой конечной точке.
- СПИУ выполняет операцию DELETE для ресурса account-consent (до подтверждения отзыва согласия Пользователя) для информирования ППУ об отзыве согласия.

#### 6.4.5. Изменения информации о счетах

Пользователь выбирает счета, к которым применяется согласие, в момент авторизации согласия.

Кроме того, список выбранных счетов может меняться из-за внешних факторов. Например:

- Счет был закрыт.
- Счет был заморожен.
- Пользователь изменяет выбранные счета во время повторной аутентификации согласия.

В таких случаях только затронутый счет удаляется из списка. ППУ не отменяет авторизацию для других счетов.

#### 6.4.6. Информация по оценке рисков

Информация для оценки рисков будет доступна:

- В заголовках FAPI.
- В дополнительных полях, передаваемых в разделе Risk полезной нагрузки объекта JSON.

### 6.5. Модель данных

#### 6.5.1. Использование метаданных для определения доступности транзакционного периода

Для API, передающих данные о транзакциях и счетах, раздел "Meta" в ответе может содержать два дополнительных поля, определяющих диапазон дат, в пределах которых находятся возвращаемые данные.

Ответы API по транзакциям и выпискам могут не содержать данные в определенном ранее диапазоне дат, потому что:

- ППУ не предоставляет данные о транзакциях и (или) выписки по счету в указанном диапазоне дат.
- Пользователь не давал согласия на получение данных о транзакциях и (или) выписки по счету в указанном диапазоне дат.

Отсутствие данных о транзакциях / о выписках по счету в полезной нагрузке не означает, что не совершалось операций со счетом в указанный период.

Для гарантии правильной интерпретации данных ППУ может предоставить даты первой и последней доступных транзакций, как часть ответа в разделе "Meta" в полях FirstAvailableDateTime и LastAvailableDateTime.

```
"Meta": {  
  "totalPages": 15,  
  "firstAvailableDateTime": "2019-05-03T00:00:00+00:00",  
  "lastAvailableDateTime": "2019-12-03T00:00:00+00:00"  
}
```

Таблица 26. Пример заполнения раздела Meta

## 6.5.2. Связь со справочниками и стандартами

API для получения информации о счете там, где это возможно, брались из объекта `camt.052` стандарта ISO 20022. Однако они были адаптированы под принципы архитектуры, указанные в стандарте Открытых банковских интерфейсов.

Отклонения от `camt.052 XML` объекта:

- Раздел заголовка и раздел с дополнительными данными объекта `camt.052` были удалены, поскольку они не требуются для RESTful API.
- Ресурсам были присвоены идентификаторы, и для этих ресурсов были разработаны структуры полезной нагрузки, а не бралось все сообщение `camt.052`, которое охватывает все ресурсы в формате отчета. Это означает, что были разработаны отдельные конечные точки и полезные нагрузки для покрытия следующих ресурсов (1-го и 2-го уровня):
  - `accounts`
  - `balances`
  - `statements`
  - `transactions`
- Для ресурсов, не включенных в стандарт ISO 20022, разрабатывались новые полезные нагрузки.
- Элемент `DateTime` был использован вместо сложного элемента выбора `Date` и `DateTime` (для всех конечных точек API). Если элементы времени не существуют в системах ППУ, ожидается, что элемент `DateTime` по умолчанию будет иметь значение формата `00:00:00+00:00`.
- Структура ресурса `accounts`:
  - Стандартизирована в соответствии со структурами счетов API переводов денежных средств.
  - Содержит элемент наименования счета (задается Пользователем).
- Структура ресурса `balances`:
  - Добавлено поле `type` в раздел `CreditLine` для учета нескольких типов кредитных линий на доступный баланс.
  - Элемент `DateTime` был указан вместо сложного выбора `Date` и `DateTime`.
- Структура ресурса `transactions` включает:
  - Поле `entry` переименовано в `transaction` согласно практикам международного опыта: CMA Order и PSD2.
  - Элемент `DateTime` был указан вместо сложного выбора `Date` и `DateTime`.
  - Оптимизированы наборы кодов для `BankTransactionCode` и `ProprietaryBankTransactionCode`.
  - Дополнительную информацию для разделов `AddressLine`, `MerchantDetails`.

## 6.5.3. Ресурсы

Каждый из ресурсов задокументирован на отдельной странице.

Для каждого ресурса в документации есть:

- Список конечных точек API, доступных для ресурса.
- Модель данных:
  - Определение ресурса.
  - UML-диаграмма, отображающая структуру ресурса (состав данных).
  - Разрешения (`permissions`), необходимые для доступа к ресурсу.
  - Спецификация данных, содержащая подробное описание элементов, кратность элементов и используемые справочные значения.
- Примеры использования

## 6.5.4. Справочники

### 6.5.4.1. Статические справочники

#### **AccountStatusStaticType**

Статус ресурса счета.

№	Значение	Описание
1	Enabled	Ресурс счета доступен и может использоваться
2	Disabled	Ресурс счета не доступен и не может использоваться, временно или навсегда
3	Deleted	Ресурс счета был удален и не может использоваться
4	Pending	Изменения ресурса счета ожидают одобрения

Таблица 27. Статус ресурса счета

**AccountTypeStaticType**

Тип ресурса банковского счета.

№	Значение	Описание
1	Business	Тип ресурса счетов для юридических лиц
2	Personal	Тип ресурса счетов для физических лиц

Таблица 28. Тип ресурса банковского счета

**AccountSubTypeStaticType**

Подтип ресурса банковского счета.

№	Значение	Описание
1	CreditCard	Кредитные карты
2	CurrentAccount	Текущие счета
3	Loan	Потребительские кредиты
4	Mortgage	Ипотека
5	PrePaidCard	Карты предоплаты
6	Savings	Депозиты

Таблица 29. Подтип ресурса банковского счета

**CreditDebitIndicatorStaticType**

Индикатор дебетовой/кредитовой операции.

№	Значение	Описание
1	Credit	Кредитовая операция
2	Debit	Дебетовая операция

Таблица 30. Индикатор дебетовой/кредитовой операции

**BalanceTypeStaticType**

Тип баланса.

№	Значение	Описание
1	ClosingAvailable	Конечный остаток суммы денег, которая находится в распоряжении владельца счета на указанную дату
2	ClosingBooked	Остаток по счету на конец предварительно согласованного отчетного периода. Это сумма начального зарегистрированного баланса в начале периода и всех записей, зачисленных на счет в течение предварительно согласованного отчетного периода
3	ClosingCleared	Конечный остаток суммы денег, которая очищается в указанную дату
4	Expected	Баланс, состоящий из забронированных записей и отложенных позиций, известных на момент расчета, который прогнозирует остаток на конец дня, если все зарезервировано на счете и никакая другая запись не публикуется
5	OpeningAvailable	Начальный баланс суммы денег, которая находится в распоряжении владельца счета на указанную дату
6	OpeningBooked	Баланс счета в начале отчетного периода. Он всегда равен балансу из предыдущего отчета
7	OpeningCleared	Начальный баланс, который очищается в указанную дату
8	PreviouslyClosedBooked	Остаток по счету за ранее закрытый отчетный период. Начальный зарегистрированный баланс на новый период равен этому балансу

Таблица 31. Тип баланса

**TransactionStatusStaticType**

Статус записи транзакции.

№	Значение	Описание
1	Booked	Забронировано
2	Pending	В ожидании, бронирование еще не произошло

Таблица 32. Статус записи транзакции

**AddressTypeStaticCode**

Тип адреса.

№	Значение	Описание
1	Business	Юридический адрес
2	Registration	Адрес регистрации
3	DeliveryTo	Адрес доставки
4	Actual	Фактический адрес
6	Postal	Почтовый адрес
7	Residential	Адрес прописки

Таблица 33. Тип адреса

**CardSchemeNameStaticType**

Наименование схемы карты.

№	Значение	Описание
1	AmericanExpress	AmericanExpress
2	Diners	Diners
3	Discover	Discover
4	MasterCard	MasterCard
6	VISA	VISA
7	MIR	MIR

Таблица 34. Наименование схемы карты

**CardAuthorisationTypeStaticType**

Тип авторизации карты.

№	Значение	Описание
1	ConsumerDevice	Устройство пользователя
2	Contactless	Бесконтактный
3	None	Отсутствует
4	PIN	PIN код

Таблица 35. Тип авторизации карты

**6.5.4.2. Справочники ISO**

Следующие справочники ISO используются в API счетов и транзакций

№	Значение	Описание
1	ActiveOrHistoricCurrencyCode	<a href="https://www.iso20022.org/external_code_list.page">https://www.iso20022.org/external_code_list.page</a>
2	CountryCode	Стандарт ГОСТ 7.67.2003 (ИСО 3166-1:1997), alpha-2 code — <a href="http://docs.cntd.ru/document/1200035671">http://docs.cntd.ru/document/1200035671</a> . Таблица 1, двухзначный буквенный код на латинице
3	ExternalBankTransactionFamily1Code	<a href="https://www.iso20022.org/external_code_list.page">https://www.iso20022.org/external_code_list.page</a>
4	ExternalBankTransactionSubFamily1Code	<a href="https://www.iso20022.org/external_code_list.page">https://www.iso20022.org/external_code_list.page</a>
5	Min3Max4Text	<a href="https://www.iso.org/standard/33365.html">https://www.iso.org/standard/33365.html</a>

Таблица 36. Международные и общероссийские справочники

## 6.6. Согласие на доступ к счету, спецификация API – v1.2.1

### 6.6.1. Конечные точки

№	Ресурс	Метод HTTP	Конечная точка	Scope	Grant Type	Ключ идемпотентности	Подписание	Объект запроса	Объект ответа
1	account-consents	POST	POST /account-consents	accounts	Client Credentials	No		Consent	ConsentResponse
2	account-consents	GET	GET /account-consents/{consentId}	accounts	Client Credentials	No	Подписывается ответ		ConsentResponse
3	account-consents	DELETE	DELETE /account-consents/{consentId}	accounts	Client Credentials	No			
4	account-consents	GET	GET /account-consents/{consentId}/retrieval-grant	accounts	Client Credentials	No	Подписывается ответ		RetrievalGrantResponse

Таблица 37. Конечные точки для ресурса согласия на доступ к счету

**6.6.1.1. POST /account-consents**

Для конечной точки должны выполняться следующие требования:

- Конечная точка позволяет СПИУ на стороне ППУ создавать новый ресурс согласия на доступ к информации о счете (ресурс account-consents).
- ППУ создает ресурс account-consents и отвечает сообщением, содержащим его уникальный идентификатор consentId.
- Для вызова конечной точки СПИУ получает токен доступа, выданный ППУ с использованием потока Client Credentials Grant.

**6.6.1.1.1. Статус согласия на доступ к информации о счете**

Для получения доступа к информации о счете Пользователь аутентифицируется на стороне ППУ и авторизует ресурс account-consents.

Успешно созданный ресурс account-consents имеет следующий статус:

№	Статус	Описание
1	AwaitingAuthorisation	Согласие на доступ к счету ожидает авторизации

Таблица 38. Статус созданного ресурса согласия на доступ к счету

После авторизации ресурс account-consents имеет следующие статусы:

№	Статус	Описание
1	Rejected	Согласие на доступ к информации о счете было отклонено
2	Authorised	Согласие на доступ к информации о счете успешно авторизовано
3	Revoked	Согласие на доступ к информации о счете было отозвано через интерфейс ППУ

Таблица 39. Возможные статусы ресурса согласия на доступ к счету

**6.6.1.1.2. Смена состояний статуса**

Диаграмма состояний для статуса согласия на доступ к информации о счете имеет следующий вид:

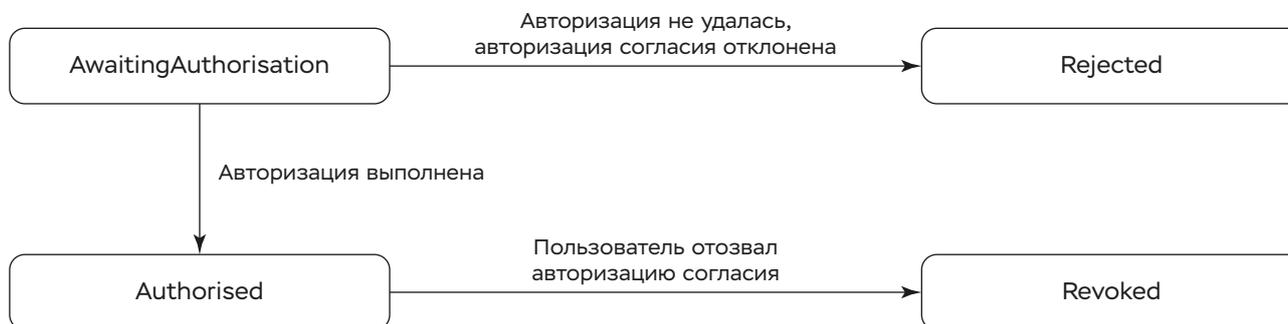


Рисунок 4. Диаграмма состояний ресурса согласия на доступ к счету

**6.6.1.2. GET /account-consents/{consentId}**

Конечная точка позволяет СПИУ получать созданный им ресурс согласия на доступ к информации о счете.

Конечная точка позволяет СПИУ проверять статус ресурса согласия на доступ к информации о счете.

Перед вызовом конечной точки СПИУ получает токен доступа на стороне ППУ с использованием потока Client Credentials Grant.

**6.6.1.3. DELETE /account-consents/{consentId}**

Если Пользователь отзывает согласие на доступ к данным для СПИУ, СПИУ удаляет ресурс account-consents на стороне ППУ до подтверждения отзыва согласия Пользователем.

- Для этого необходимо сделать вызов DELETE ресурса account-consents.
- Перед вызовом API СПИУ получает токен доступа, выдаваемый ППУ с использованием потока Client Credentials Grant.

**6.6.1.4. GET /account-consents/{consentId}/retrieval-grant**

Данный ресурс создается на ресурсном сервере ППУ после авторизации согласия на получение информации о счете Пользователем (после авторизации ресурса account-consents).

Период действия поручения на извлечение (ресурса retrieval-grant) совпадает с периодом действия согласия на получение информации о счете (ресурса account-consents).

## 6.6.2. Модель данных

### 6.6.2.1. Согласие на доступ к информации о счете – Запрос

Объект Consent используется в запросе к конечной точке:

- POST /account-consents

#### 6.6.2.1.1. Диаграмма UML

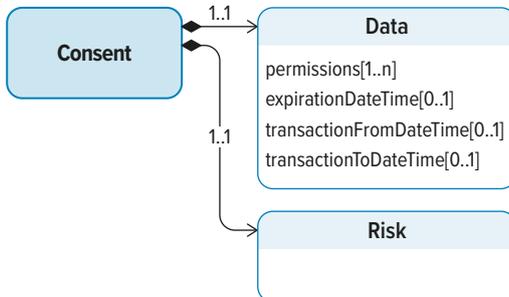


Рисунок 5. Диаграмма объекта Consent

## 6.6.2.1.2. Состав данных объекта Consent:

Наименование	Кратность	Путь	Описание	Тип	Значения
Consent		Consent		ConsentType	
Data	1..1	Consent/Data		DataType	
permissions	1..n	Consent/Data/permissions	Указание типов данных доступа к счетам Пользователя. Это список доменов данных, которые были одобрены Пользователем и запрошены для авторизации на стороне ППУ	PermissionsType	ReadAccountsBasic ReadAccountsDetail ReadBalances ReadTransactionsBasic ReadTransactionsCredits ReadTransactionsDebits ReadTransactionsDetail
expirationDateTime	0..1	Consent/Data/expirationDateTime	Дата и время истечения срока действия разрешений. Если он не заполнен, разрешения будут с открытой датой	ISODatetime	
transactionFromDateTime	0..1	Consent/Data/transactionFromDateTime	Дата и время начала периода запроса транзакции. Если он не заполнен, дата начала будет с открытой датой, и данные будут возвращены с самой ранней из доступных транзакций	ISODatetime	
transactionToDateTime	0..1	Consent/Data/transactionToDateTime	Дата и время окончания периода запроса транзакции. Если он не заполнен, дата окончания будет открытой, и данные будут возвращены в самую последнюю доступную транзакцию	ISODatetime	
Risk	1..1	Consent/Risk	Risk направляется иницилирующей стороной в ППУ. Используется для указания дополнительной информации для скоринга рисков	RiskType	

Таблица 40. Состав данных объекта Consent

### 6.6.2.2. Согласие на доступ к информации о счете – Ответ

Объект ConsentResponse используется в ответах при обращениях к следующим конечным точкам:

- GET /account-consents/{consentId}
- POST /account-consents

#### 6.6.2.2.1. Диаграмма UML

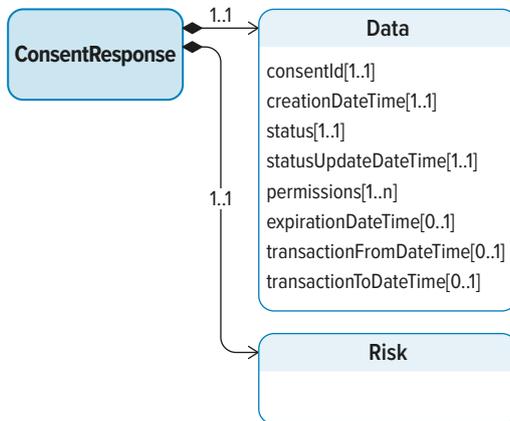


Рисунок 6. Диаграмма объекта ConsentResponse

## 6.6.2.2.2. Состав данных объекта ConsentResponse:

Наименование	Кратность	Путь	Описание	Тип	Значения
ConsentResponse		ConsentResponse		ConsentResponseType	
Data	1..1	ConsentResponse/Data		DataConsentResponseType	
consentId	1..1	ConsentResponse/Data/consentId	Уникальный идентификатор, предназначенный для идентификации ресурса согласия на доступ к счету	Max128Text	
creationDateTime	1..1	ConsentResponse/Data/creationDateTime	Дата и время создания ресурса	ISODateTime	
status	1..1	ConsentResponse/Data/status	Статус согласия	ConsentStatusType	Authorised AwaitingAuthorisation Rejected Revoked
statusUpdateDateTime	1..1	ConsentResponse/Data/statusUpdateDateTime	Дата и время обновления статуса ресурса	ISODateTime	
permissions	1..n	ConsentResponse/Data/permissions	Указание типов данных доступа к счетам Пользователя. Это список доменов данных, которые были одобрены Пользователем и запрошены для авторизации на стороне ППУ	PermissionsType	ReadAccountsBasic ReadAccountsDetail ReadBalances ReadTransactionsBasic ReadTransactionsCredits ReadTransactionsDebits ReadTransactionsDetail
expirationDateTime	0..1	ConsentResponse/Data/expirationDateTime	Дата и время истечения срока действия разрешений. Если он не заполнен, разрешения будут с открытой датой	ISODateTime	
transactionFromDateTime	0..1	ConsentResponse/Data/transactionFromDateTime	Дата и время начала периода запроса транзакции. Если он не заполнен, дата начала будет с открытой датой, и данные будут возвращены с самой ранней из доступных транзакций	ISODateTime	
transactionToDateTime	0..1	ConsentResponse/Data/transactionToDateTime	Дата и время окончания периода запроса транзакции. Если он не заполнен, дата окончания будет открытой, и данные будут возвращены в самую последнюю доступную транзакцию	ISODateTime	
Risk	1..1	ConsentResponse/Risk	Risk направляется иницилирующей стороной на сторону ППУ. Используется для указания дополнительной информации для скоринга рисков	RiskType	

Таблица 41. Состав данных объекта ConsentResponse

### 6.6.2.3. Поручение на извлечение информации о счете – Ответ

Объект RetrievalGrantResponse используется в ответах при обращениях к следующим конечным точкам:

- GET /account-consents/{consentId}/retrieval-grant

#### 6.6.2.3.1. Диаграмма UML

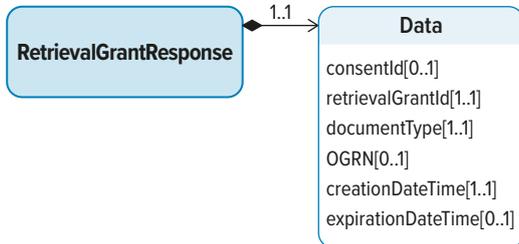


Рисунок 7. Диаграмма объекта RetrievalGrantResponse

### 6.6.2.3.2. Состав данных объекта RetrievalGrantResponse:

Наименование	Кратность	Путь	Описание	Тип	Значения
<b>RetrievalGrantResponse</b>		<b>RetrievalGrantResponse</b>		<b>RetrievalGrantResponseType</b>	
<b>Data</b>	<b>1.1</b>	<b>RetrievalGrantResponse/Data</b>		<b>DataConsentResponseType</b>	
consentId	0..1	RetrievalGrantResponse/Data/consentId	Уникальный идентификатор, предназначенный для идентификации ресурса согласия на доступ к счету	Max128Text	
retrievalGrantId	1.1	RetrievalGrantResponse/Data/retrievalGrantId	Уникальный идентификатор, предназначенный для идентификации ресурса поручения на доступ к счету	Max128Text	
documentType	1.1	RetrievalGrantResponse/Data/documentType	Тип документа. Пока передается фиксированное значение "Поручение на извлечение"	Max128Text	
OGRN	0..1	RetrievalGrantResponse/Data/OGRN	ОГРН Стороннего поставщика. Поле может не передаваться, поскольку у Стороннего поставщика известен его ОГРН	Max13Text	
creationDateTime	1.1	RetrievalGrantResponse/Data/creationDateTime	Дата и время создания ресурса	ISODateTime	
expirationDateTime	0..1	RetrievalGrantResponse/Data/expirationDateTime	Дата и время истечения срока действия поручения на извлечение информации о счете. Дата и время совпадают с датой и временем для согласия на получение информации о счете	ISODateTime	

Таблица 42. Состав данных объекта RetrievalGrantResponse

### 6.6.3. Пример использования

#### 6.6.3.1. Установка согласия на доступ к информации о счете – все разрешения (permissions)

СПИУ создает на стороне ППУ ресурс согласия с разрешениями на доступ к кластерам данных “ReadAccountsDetail”, “ReadBalances”, “ReadBeneficiariesDetail”, “ReadTransactionsCredits”, “ReadTransactionsDebits”, “ReadTransactionsDetail” со сроком действия до 3 сентября 2019 года. Также задается диапазон дат с 3 мая 2019 года по 3 сентября 2019 года, в рамках которого доступны транзакции Пользователя.

Все параметры задаются Пользователем заранее на стороне СПИУ.

СПИУ создает на стороне ППУ ресурс согласия на доступ к информации о счете с разрешениями кластерных данных “ReadAccountsDetail”, “ReadBalances”, “ReadBeneficiariesDetail”, “ReadTransactionsCredits”, “ReadTransactionsDebits”, “ReadTransactionsDetail” со сроком действия до 3 сентября 2019 года вызовом конечной точки API POST /accounts-consent.

В ответ от ППУ СПИУ получает согласие с идентификатором согласия (consentId) “urn-alphabank-intent-88379” со статусом ожидания авторизации “AwaitingAuthorisation” и с перечислением предоставленных ресурсов и сроков, в рамках которого доступны транзакции Пользователя.

```
POST /account-consents HTTP/1.1
Authorization: Bearer 5YotnWEFEqrlzDsicMwPSS
x-fapi-auth-date: Sun, 02 May 2019 19:43:31 GMT
x-fapi-customer-ip-address: 18.25.212.99
x-fapi-interaction-id: 21bac548-d2de-1237-b106-880a5018460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "permissions": [
      "ReadAccountsDetail",
      "ReadBalances",
      "ReadBeneficiariesDetail",
      "ReadTransactionsCredits",
      "ReadTransactionsDebits",
      "ReadTransactionsDetail",
    ],
    "expirationDateTime": "2019-09-03T00:00:00+00:00",
    "transactionFromDate": "2019-05-03T00:00:00+00:00",
    "transactionToDate": "2019-09-03T00:00:00+00:00"
  },
  "Risk": {}
}
```

Таблица 43. Запрос на установку согласия на доступ к счету

```
HTTP/1.1 201 Created
x-fapi-interaction-id: 21bac548-d2de-1238-b106-880a5018460d
Content-Type: application/json
{
  "Data": {
    "consentId": "urn-alphabank-intent-88379",
    "status": "AwaitingAuthorisation",
    "statusUpdateDateTime": "2019-05-02T00:00:00+00:00",
    "creationDateTime": "2019-05-02T00:00:00+00:00",
    "permissions": [
      "ReadAccountsDetail",
      "ReadBalances",
      "ReadBeneficiariesDetail",
      "ReadTransactionsCredits",
      "ReadTransactionsDebits",
      "ReadTransactionsDetail",
    ],
    "expirationDateTime": "2019-08-02T00:00:00+00:00",
    "transactionFromDateDateTime": "2019-05-03T00:00:00+00:00",
    "transactionToDateDateTime": "2019-09-03T00:00:00+00:00"
  },
  "Risk": {},
  "Links": {
    "self": "https://bank.ru/open-banking/v1.0/aisp/account-consents/bank-consent-88379"
  },
  "Meta": {
    "totalPages": 1
  }
}
```

Таблица 44. Ответ на установку согласия на доступ к счету

## 6.7. Счета, спецификация API – v1.2.1

### 6.7.1. Конечные точки

№	Ресурс	Метод HTTP	Конечная точка	Scope	Grant Type	Ключ идемпотентности	Параметры	Объект запроса	Объект ответа
1	accounts	GET	GET /accounts	accounts	Authorization Code	Нет	Нумерация страниц		AccountResponse
2	accounts	GET	GET /accounts/{accountId}	accounts	Authorization Code	Нет			AccountResponse

Таблица 45. Конечные точки ресурса счета

#### 6.7.1.1. GET /accounts

СПИУ предоставляется полный список счетов (с идентификаторами accountId), которые были авторизованы Пользователем на стороне ППУ.

#### 6.7.1.2. GET /accounts/{accountId}

СПИУ получает детальную информацию о счете по идентификатору accountId (который был получен при вызове конечной точки списка счетов GET /accounts).

## 6.7.2. Модель данных

Объект AccountResponse используется в ответах при вызове конечных точек:

- GET /accounts
- GET /accounts/{accountId}

### 6.7.2.1. Описание ресурсов

Ресурс представляет собой банковский счет, на котором делаются кредитовые и дебетовые записи. Каждый ресурс счета будет иметь уникальный и неизменный идентификатор accountId.

### 6.7.2.2. UML-диаграмма

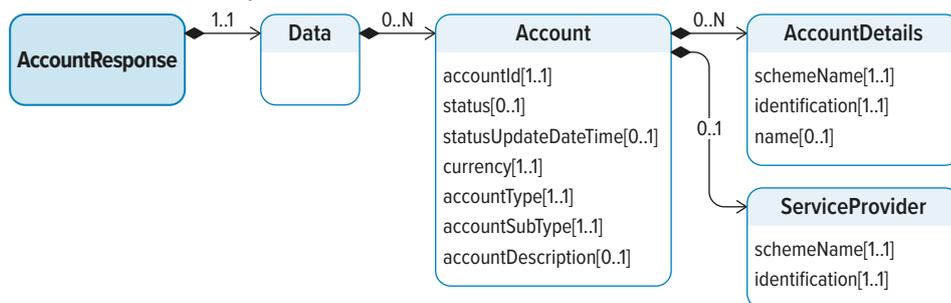


Рисунок 8. Диаграмма объекта AccountResponse

Структуры AccountDetails и ServiceProvider были спроектированы для соответствия структурам идентификации банковского счета Плательщика и идентификации банка Плательщика при использовании СПИУ.

Наличие элемента schemeName для деталей счета и поставщика сервиса (организация, в которой открыт счет) позволяет использовать разные типы идентификаторов. Например, операции могут осуществляться по номеру счета, номеру карты, номеру телефона.

### 6.7.2.3. Описание доступа к кластерам данных

Состав данных ресурса зависит от разрешений (ReadAccountsBasic and ReadAccountsDetail), которые используются для доступа к ресурсу. В случае, когда доступ к ресурсу выдан с обоими разрешениями ReadAccountsBasic и ReadAccountsDetail, то более детализированные разрешения будут использоваться (ReadAccountsDetail).

При работе с ресурсом accounts выполняются правила:

- Следующие объекты не возвращаются без разрешения **ReadAccountsDetail**:
  - AccountResponse/Data/Account/AccountDetails
  - AccountResponse/Data/Account/ServiceProvider
- Если Пользователь дал разрешение **ReadAccountsDetail**, то:
  - Объект AccountResponse/Data/Account/AccountDetails возвращается (1..n)
  - Объект AccountResponse/Data/Account/ServiceProvider может возвращаться (0..1)

#### 6.7.2.4. Состав данных объекта AccountResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
AccountResponse		AccountResponse		AccountResponseComplexType		
Data	1..1	AccountResponse/Data		DataAccountResponseComplexType		
Account	0..N	AccountResponse/Data/Account	Комплексный объект	AccountComplexType		
accountId	1..1	AccountResponse/Data/Account/accountId	Уникальный и неизменный идентификатор, используемый для идентификации ресурса "accounts"	Max40Text		
status	0..1	AccountResponse/Data/Account/status	Статус счета в форме кода	AccountStatusStaticType	Enabled Disabled Deleted Pending	
statusUpdateDateTime	0..1	AccountResponse/Data/Account/statusUpdateDateTime	Дата и время изменения статуса счета. Используется стандарт ISO 8601	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
currency	1..1	AccountResponse/Data/Account/currency	Валюта ведения счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
accountType	1..1	AccountResponse/Data/Account/accountType	Тип счета (физическое или юридическое лицо)	AccountTypeStaticType	Business Personal	
accountSubType	1..1	AccountResponse/Data/Account/accountSubType	Подтип счета (классификация банковских продуктов)	AccountSubTypeStaticType	CreditCard CurrentAccount Loan Mortgage PrePaidCard Savings	
accountDescription	0..1	AccountResponse/Data/Account/accountDescription	Детальное описание продукта, привязанного к счету	Max35Text		
AccountDetails	0..N	AccountResponse/Data/Account/AccountDetails	Подробная информация для идентификации счета	AccountDetailsComplexType		
schemeName	1..1	AccountResponse/Data/Account/AccountDetails/schemeName	Наименование схемы идентификации счета	AccountIdentificationDynamicType		
identification	1..1	AccountResponse/Data/Account/AccountDetails/identification	Номер счета, присваивается организацией для идентификации счета. Эта идентификация известна владельцу счета	Max256Text		
name	0..1	AccountResponse/Data/Account/AccountDetails/name	Название идентификатора счета	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
ServiceProvider	0..1	AccountResponse/Data/ Account/ServiceProvider	Организация, которая управляет счетом от имени владельца счета, то есть управляет регистрацией и бронированием записей на счете, рассчитывает остатки на счете и предоставляет информацию о счете	ServiceProviderComplexType		
schemeName	1..1	AccountResponse/Data/ Account/ServiceProvider/ schemeName	Наименование схемы организации	FinancialInstitutionIdentification DynamicType		
identification	1..1	AccountResponse/Data/ Account/ServiceProvider/ identification	Идентификатор организации	Max35Text		

Таблица 46. Состав данных объекта AccountResponse

### 6.7.3. Примеры использования

#### 6.7.3.1. Список – разрешение на детали

Вызов конечной точки API GET /accounts осуществляется первым после авторизации согласия. Это позволяет СПИУ узнать, какие счета соответствуют авторизованному согласию Пользователя.

СПИУ получает от ППУ текущий рублевый счет Пользователя с разрешением **ReadAccountsDetail**. Также ППУ возвращает идентификаторы ресурсов “23489” и “31820”.

После авторизации согласия на доступ к счету СПИУ осуществляет запрос получения списка счетов вызовом конечной точки API GET /accounts.

В ответе от ППУ СПИУ получает список счетов Пользователя с идентификаторами ресурсов (AccountId) “23489” и “31820”, соответствующие данному им авторизационному согласию с разрешением **ReadAccountsDetail**.

```
GET /accounts HTTP/1.1
Authorization: Bearer Az34SAOJkldf
x-fapi-auth-date: Sun, 8 Sep 2019 23:43:12 GMT
x-fapi-customer-ip-address: 134.21.122.22
x-fapi-interaction-id: 93bac548-f5fe-6780-b106-880a5018460d
Accept: application/json
```

Таблица 47. Запрос получения списка счетов

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 93bac548-f5fe-6780-b106-880a5018460d
Content-Type: application/json

{
  "Data": {
    "Account": [
      {
        "accountId": "23489",
        "status": "Enabled",
        "statusUpdateDateTime": "2019-01-01T06:06:06+00:00",
        "currency": "RUB",
        "accountType": "Personal",
        "accountSubType": "CurrentAccount",
        "accountDetails": [
          {
            "schemeName": "RU.CBR.AccountNumber",
            "identification": "40817810621234567890",
            "name": "Основной текущий счет",
          }
        ]
      },
      {
        "accountId": "31820",
        "status": "Enabled",
        "statusUpdateDateTime": "2019-01-01T06:06:06+00:00",
        "currency": "RUB",
        "accountType": "Personal",
        "accountSubType": "CurrentAccount",
        "accountDetails": [
          {
            "schemeName": "RU.CBR.AccountNumber",
            "identification": "40817810621234562345",
            "name": "Дополнительный текущий счет"
          }
        ]
      }
    ]
  },
  "Links": {
    "self": "https://bank.ru/open-banking/v3.1/aisp/accounts/"
  },
  "Meta": {
    "totalPages": 1
  }
}
```

Таблица 48. Ответ получения списка счетов

## 6.8. Баланс, спецификация API – v1.2.1

### 6.8.1. Конечные точки

№	Ресурс	Метод HTTP	Конечная точка	Scope	Grant Type	Ключ идемпотентности	Параметры	Объект запроса	Объект ответа
1	balances	GET	GET /accounts/{accountId}/balances	accounts	Authorization Code	No			BalanceResponse
2	balances	GET	GET /balances	accounts	Authorization Code	No	Pagination		BalanceResponse

Таблица 49. Конечные точки ресурса баланс

#### 6.8.1.1. GET /accounts/{accountId}/balances

Конечная точка используется для получения информации о балансе банковского счета с идентификатором accountId.

#### 6.8.1.2. GET /balances

Конечная точка используется для получения баланса по нескольким счетам Пользователя.

Баланс передается по всем банковским счетам, которые Пользователь авторизовал с помощью согласия с балансовыми разрешениями (permissions).

## 6.8.2. Модель данных

Объект `BalanceResponse` используется в ответах от конечных точек:

- GET `/accounts/{accountId}/balances`
- GET `/balances`

## 6.8.3. Описание ресурсов

Данный ресурс представляет собой отображение уменьшения или увеличения баланса счета с уникальным идентификатором (`accountId`) в определенный момент времени.

Банковский счет может иметь несколько типов баланса (выбранные типы баланса соответствуют пересечению стандарта ISO 20022 и потребностей российского рынка банковских услуг). Если ПТТУ включает кредитную линию в доступный баланс (`available balance`), то в структуре объекта баланса будет специальный раздел для суммы и типа кредитной линии.

### 6.8.3.1. UML-диаграмма

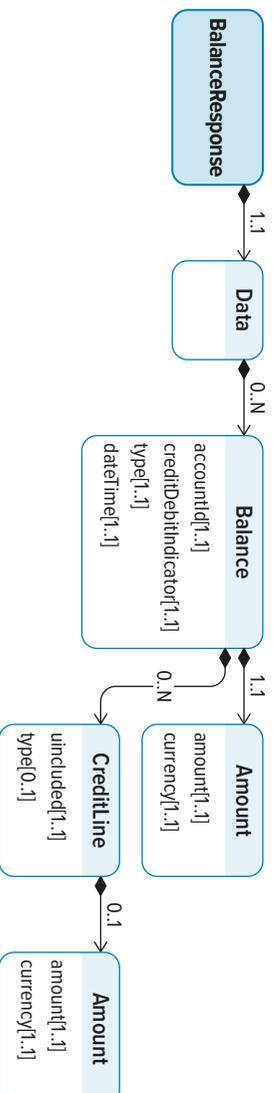


Рисунок 9. Диаграмма объекта `BalanceResponse`

Для банковского счета может быть возвращено несколько значений баланса, поскольку баланс разного типа может иметь разные значения.

У баланса с типом “доступный” (`available balance`) может быть несколько кредитных линий, соответственно объект баланса будет содержать несколько разделов `CreditLine`.

Если элементы времени не существуют в системах ПТТУ, часть времени элемента `DateTime` по умолчанию будет иметь значение `00:00:00+00:00`.

### 6.8.3.2. Описание доступа к кластерам данных

Ресурс имеет разрешение `ReadBalances`.

## 6.8.3.3. Состав данных объекта BalanceResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
BalanceResponse		BalanceResponse		BalanceResponseComplexType		
Data	1..1	BalanceResponse/Data		DataBalanceResponseComplexType		
Balance	0..N	BalanceResponse/Data/Balance		BalanceComplexType		
accountId	1..1	BalanceResponse/Data/Balance/accountId	Уникальный и неизменный идентификатор, используемый для идентификации ресурса 'accounts'	Max40Text		
creditDebitIndicator	1..1	BalanceResponse/Data/Balance/creditDebitIndicator	Определяет, является баланс кредитовым или дебетовым	CreditDebitIndicatorStaticType	Credit Debit	
type	1..1	BalanceResponse/Data/Balance/type	Тип баланса, заполняется согласно ISO 20022	BalanceTypeStaticType	ClosingAvailable ClosingBooked ClosingCleared Expected ForwardAvailable Information InterimAvailable InterimBooked InterimCleared OpeningAvailable OpeningBooked OpeningCleared PreviouslyClosedBooked	
dateTime	1..1	BalanceResponse/Data/Balance/dateTime	Дата и время построения отчета. Используется стандарт ISO 8601	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
Amount	1..1	BalanceResponse/Data/Balance/Amount	Детали баланса	AmountComplexType		
amount	1..1	BalanceResponse/Data/Balance/Amount/amount	Сумма	ActiveOrHistoricCurrencyAndAmount_SimpleType		^d{1,13}\.d{1,5}\$
currency	1..1	BalanceResponse/Data/Balance/Amount/currency	Валюта счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
CreditLine	0..N	BalanceResponse/Data/Balance/CreditLine	Подробная информация о кредитной линии	CreditLineComplexType		
included	1..1	BalanceResponse/Data/Balance/CreditLine/included	Указывает, включена ли кредитная линия в баланс счета. Если отсутствует, кредитная линия не включается в сумму баланса счета	xs:boolean		
type	0..1	BalanceResponse/Data/Balance/CreditLine/type	Тип кредитного лимита	creditLineStaticType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Amount	0..1	BalanceResponse/Data/ Balance/CreditLine/Amount	Детали баланса	AmountComplexType		
amount	1..1	BalanceResponse/Data/ Balance/CreditLine/Amount/ amount	Сумма	ActiveOrHistoricCurrencyAnd Amount_SimpleType		^\\d{1,13}\\.\d{1,5}\$
currency	1..1	BalanceResponse/Data/ Balance/CreditLine/Amount/ currency	Валюта счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$

Таблица 50. Состав данных объекта BalanceResponse

#### 6.8.4. Примеры использования

Получение баланса по идентификатору счета.

Запрос получения баланса. СПИУ выполняет к ППУ запрос информации о балансе на счете с идентификатором (accountId) "11139" вызовом конечной точки GET /accounts/11139/balances без параметров.

Ответ получения баланса. ППУ возвращает СПИУ информацию о положительном балансе на указанном счете в размере 13 430,00 рублей и наличии включенной в него кредитной линии в размере 4000,00 рублей.

```
GET /accounts/11139/balances HTTP/1.1
Authorization: Bearer Sd6753A0Jweae
x-fapi-auth-date: Sun, 15 Sep 2019 11:22:31 GMT
x-fapi-customer-ip-address: 10.11.21.39
x-fapi-interaction-id: 23bac548-d2de-4567-b106-880a5018460d
Accept: application/json
```

Таблица 51. Запрос получения баланса

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 23bac548-d2de-4567-b106-880a5018460d
Content-Type: application/json

{
  "Data": {
    "Balance": [
      {
        "accountId": "11139",
        "Amount": {
          "amount": "13430.00",
          "currency": "RUB"
        },
        "CreditDebitIndicator": "Credit",
        "Type": "OpeningAvailable",
        "DateTime": "2019-09-15T14:33:07+00:00",
        "CreditLine": [
          {
            "Included": true,
            "Amount": {
              "Amount": "4000.00",
              "Currency": "RUB"
            },
            "Type": "Pre-Agreed"
          }
        ]
      }
    ]
  },
  "Links": {
    "Self": "https://api.alphabank.com/open-banking/v3.1/aisp/accounts/22289/balances/"
  },
  "Meta": {
    "TotalPages": 1
  }
}
```

Таблица 52. Ответ получения баланса

Получение списка объектов баланса по идентификатору Пользователя (по всем счетам, к которым Пользователь авторизовал согласие).

Запрос получения баланса по списку счетов. СПИУ выполняет запрос к ППУ информации о балансе по всем счетам, к которым Пользователь авторизовал согласие вызовом конечной точки GET /balances без параметров.

Ответ получения баланса по списку счетов. ППУ возвращает СПИУ список счетов с информацией о балансе:

“accountId”: “11139” имеет положительный баланс в размере 2345,00 рублей и наличие включенной в него кредитной линии в размере 1000,00 рублей;

“accountId”: “76533” имеет отрицательный баланс в размере 257,00.

```
GET /balances HTTP/1.1
Authorization: Bearer Sd6753A0Jweae
x-fapi-auth-date: Sun, 15 Sep 2019 11:22:31 GMT
x-fapi-customer-ip-address: 10.11.21.39
x-fapi-interaction-id: 23bac548-d2de-4567-b106-880a5018460d
Accept: application/json
```

Таблица 53. Запрос получения баланса по списку счетов

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 23bac548-d2de-4567-b106-880a5018460d
Content-Type: application/json

{
  "Data": {
    "Balance": [
      {
        "accountId": "11139",
        "Amount": {
          "amount": "2345.00",
          "currency": "GBP"
        },
        "creditDebitIndicator": "Credit",
        "type": "OpeningAvailable",
        "dateTime": "2017-04-05T10:43:07+00:00",
        "CreditLine": [
          {
            "included": true,
            "Amount": {
              "amount": "1000.00",
              "currency": "GBP"
            },
            "type": "Pre-Agreed"
          }
        ]
      },
      {
        "AccountId": "76533",
        "Amount": {
          "Amount": "257.00",
          "Currency": "GBP"
        },
        "CreditDebitIndicator": "Debit",
        "Type": "OpeningAvailable",
        "DateTime": "2019-09-15T14:22:09+00:00"
      }
    ]
  },
  "Links": {
    "Self": "https://bank.ru/open-banking/v1.0/aisp/balances/"
  },
  "Meta": {
    "TotalPages": 1
  }
}
```

Таблица 54. Ответ получения баланса по списку счетов

## 6.9. Транзакции, спецификация API – v1.2.1

### 6.9.1. Конечные точки

№	Ресурс	Метод HTTP	Конечная точка	Scope	Grant Type	Ключ идемпотентности	Параметры	Объект запроса	Объект ответа
1	transactions	GET	GET /accounts/{accountId}/transactions	accounts	Authorization Code	Нет	Нумерация страниц Фильтрация		TransactionResponse
2	transactions	GET	GET /transactions	accounts	Authorization Code	Нет	Нумерация страниц Фильтрация		TransactionResponse

Таблица 55. Конечные точки ресурса транзакция

#### 6.9.1.1. GET /accounts/{accountId}/transactions

Конечная точка извлекает ресурс транзакции с идентификатором accountId, который получается с помощью вызова конечной точки GET /accounts.

#### 6.9.1.2. GET /transactions

Конечная точка позволяет получить список транзакций по всем счетам, которые авторизованы Пользователем с помощью согласия.

### 6.9.2. Модель данных

Объект TransactionResponse используется в ответах при вызове следующих конечных точек:

- GET /accounts/{accountId}/transactions
- GET /transactions

#### 6.9.2.1. Описание ресурсов

Ресурс описывает проводку банковского счета, которая приводит к увеличению или уменьшению баланса.

6.9.2.2. UML-диаграмма

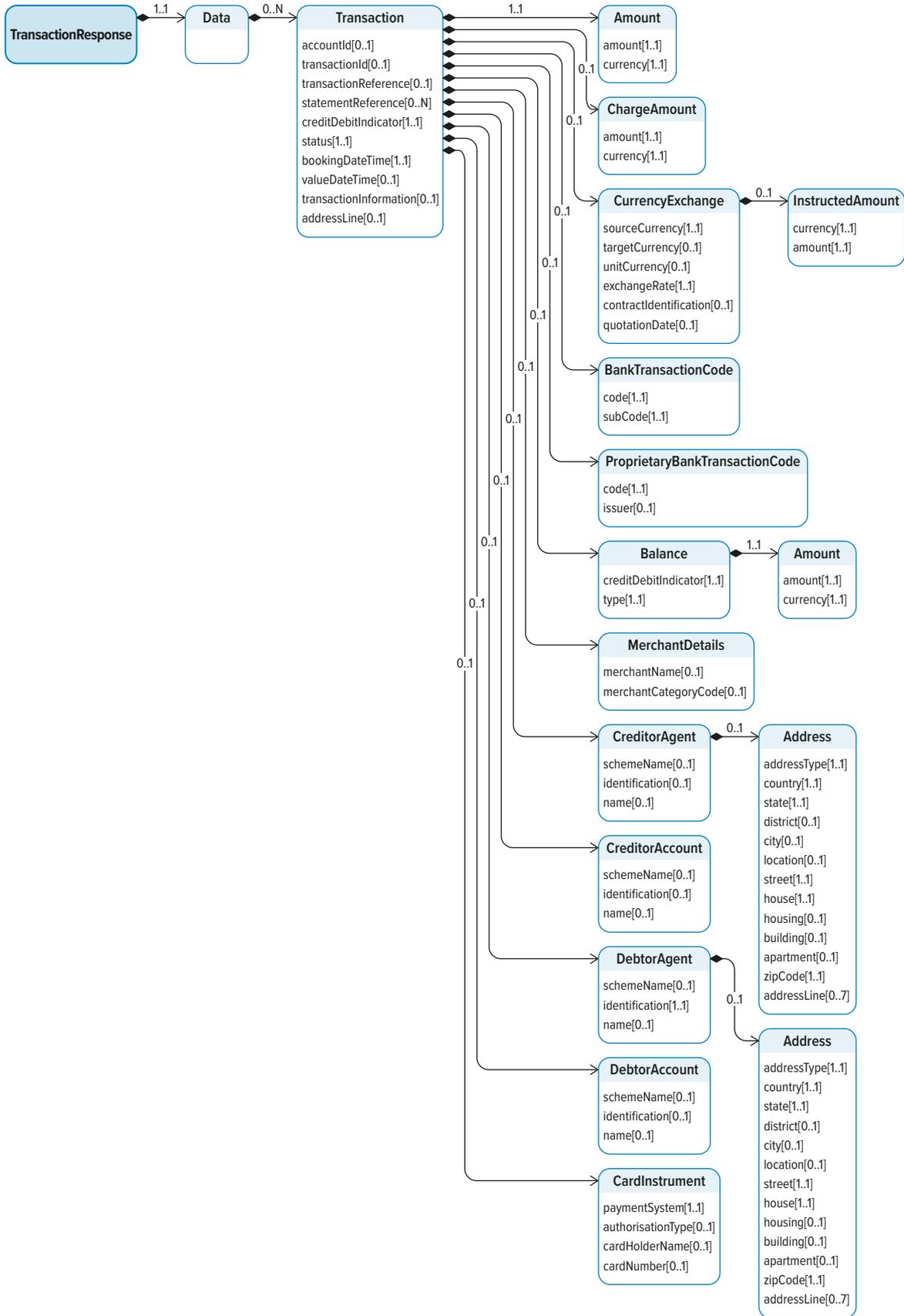


Рисунок 10. Диаграмма объекта TransactionResponse

Вместо элемента комплексного типа “Entrty”, который используется для транзакций в стандарте ISO 20022, в Открытых банковских интерфейсах используется элемент “Transaction” комплексного типа.

Вместо элемента сложного типа выбора Date и DateTime в Открытых банковских интерфейсах используется элемент DateTime. Если элементы времени не существуют в системах ППУ, часть времени элемента DateTime по умолчанию будет иметь значение 00:00:00+00:00.

Все ППУ предоставляют поле “BookingDateTime” для разбивки на страницы и для фильтрации, поэтому поле является обязательным для пересылки. BookingDateTime — это дата, когда транзакция забронирована (или проведена) и становится неизменной, а не дата совершения транзакции.

### 6.9.2.3. Фильтрация

Для ограничения списка транзакций, которые возвращает ППУ, используются следующие параметры запроса при обращении к ресурсу:

Наименование	Кратность	Описание	Тип данных
fromBookingDateTime	0..1	Дата и время начала фильтрации списка транзакций	ISODateTime
toBookingDateTime	0..1	Дата и время окончания фильтрации списка транзакций	ISODateTime

Таблица 56. Фильтрация транзакций

ППУ рассматривает следующее как допустимый ввод:

- Нерабочие дни (например, воскресенье или праздничные дни) или любые другие дни, когда транзакции не регистрируются.
- Даты, выходящие за пределы диапазона, для которого информация о транзакциях предоставляется через API.
- Даты, выходящие за пределы диапазона, для которого Пользователь авторизовал согласие.
- Часовой пояс может быть включен в запрос фильтра, но игнорируется ППУ. ППУ использует локальный часовой пояс.

В вышеуказанных ситуациях ППУ возвращает данные за оставшийся действительный период, указанный фильтром.

```
// Все транзакции начиная с января 2019 года (по всем счетам пользователя)
GET /transactions?fromBookingDateTime=2019-01-01T00:00:00

// Все транзакции за 2020 (по всем счетам пользователя)
GET /transactions?fromBookingDateTime=2020-01-01T00:00:00&toBookingDateTime=2020-12-31T23:59:59

// Все транзакции для определенного счета до 31 марта 2019 года
GET /accounts/87562/transactions?toBookingDateTime=2019-03-31T23:59:59
```

Таблица 57. Пример фильтрации транзакций

### 6.9.2.4. Описание доступа к кластерам данных

Объект с данными, который ППУ возвращает, зависит от разрешений (ReadTransactionsBasic и ReadTransactionsDetail), используемых для доступа к ресурсу. В случае если к ресурсу обращаются как с помощью ReadTransactionsBasic, так и ReadTransactionsDetail, то используется самый подробный уровень ReadTransactionsDetail.

- Следующие разделы (комплексные типы) и элементы (простые типы) не возвращаются без разрешения

#### ReadTransactionsDetail:

- TransactionResponse/Data/Transaction/transactionInformation
- TransactionResponse/Data/Transaction/Balance
- TransactionResponse/Data/Transaction/MerchantDetails
- TransactionResponse/Data/Transaction/CreditorAgent
- TransactionResponse/Data/Transaction/CreditorAccount
- TransactionResponse/Data/Transaction/DebtorAgent
- TransactionResponse/Data/Transaction/DebtorAccount

### 6.9.2.5. Состав данных объекта TransactionResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
TransactionResponse		TransactionResponse		TransactionResponseComplexType		
Data	1..1	TransactionResponse/Data		DataTransactionResponseComplexType		
Transaction	0..n	TransactionResponse/Data/Transaction		TransactionComplex		
accountId	1..1	TransactionResponse/Data/Transaction/accountId	Уникальный и неизменный идентификатор, используемый для идентификации ресурса "accounts"	Max40Text		
transactionId	0..1	TransactionResponse/Data/Transaction/transactionId	Уникальный и неизменный идентификатор, используемый для идентификации ресурса "transactions"	Max210Text		
transactionReference	0..1	TransactionResponse/Data/Transaction/transactionReference	Уникальная ссылка на сделку	Max35Text		
creditDebitIndicator	1..1	TransactionResponse/Data/Transaction/creditDebitIndicator	Определяет, является баланс кредитовым или дебетовым	CreditDebitIndicatorStaticType	Credit Debit	
status	1..1	TransactionResponse/Data/Transaction/status	Статус транзакции	TransactionStatusStaticType	Booked Pending	
bookingDateTime	1..1	TransactionResponse/Data/Transaction/bookingDateTime	Дата и время, когда запись о транзакции публикуется на счете в бухгалтерской книге обслуживающей организации. Используется стандарт ISO 8601	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
valueDateTime	0..1	TransactionResponse/Data/Transaction/valueDateTime	Дата и время, когда активы становятся доступными владельцу счета в случае ввода кредита или перестают быть доступными владельцу счета в случае ввода дебетовой транзакции. Используется стандарт ISO 8601	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
transactionInformation	0..1	TransactionResponse/Data/Transaction/transactionInformation	Детали транзакции	Max500Text		
addressLine	0..1	TransactionResponse/Data/Transaction/addressLine	Информация, которая находит и идентифицирует конкретный адрес для записи транзакции, который представлен в тексте в произвольном формате	Max70Text		
Amount	1..1	TransactionResponse/Data/Transaction/Amount	Информация о сумме и валюте транзакции	AmountComplexType		
amount	1..1	TransactionResponse/Data/Transaction/Amount/amount	Сумма	ActiveCurrencyAndAmountSimpleType		^d{1,13}\.d{1,5}\$

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
currency	1.1	TransactionResponse/Data/Transaction/Amount/currency	Валюта счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
ChargeAmount	0..1	TransactionResponse/Data/Transaction/ChargeAmount	Комиссионные за транзакцию	AmountComplexType		
amount	1.1	TransactionResponse/Data/Transaction/ChargeAmount/amount	Сумма	ActiveCurrencyAndAmount_SimpleType		^\d{1,13}\.\d{1,5}\$
currency	1.1	TransactionResponse/Data/Transaction/ChargeAmount/currency	Валюта счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
CurrencyExchange	0..1	TransactionResponse/Data/Transaction/CurrencyExchange	Подробная информация об обмене валюты	CurrencyExchangeComplexType		
sourceCurrency	1.1	TransactionResponse/Data/Transaction/CurrencyExchange/sourceCurrency	Валюта, из которой необходимо конвертировать сумму. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
targetCurrency	0..1	TransactionResponse/Data/Transaction/CurrencyExchange/targetCurrency	Валюта, в которую необходимо конвертировать сумму. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
unitCurrency	0..1	TransactionResponse/Data/Transaction/CurrencyExchange/unitCurrency	Валюта, в которой обменный курс выражен. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
exchangeRate	1.1	TransactionResponse/Data/Transaction/CurrencyExchange/exchangeRate	Коэффициент, используемый для перевода суммы из одной валюты в другую. Это отражает цену, по которой одна валюта была куплена за другую валюту	BaseOneRate		
contractIdentification	0..1	TransactionResponse/Data/Transaction/CurrencyExchange/contractIdentification	Идентификатор для однозначного определения валютного контракта	Max35Text		
quotationDate	0..1	TransactionResponse/Data/Transaction/CurrencyExchange/quotationDate	Дата и время обменного курса. Используется стандарт ISO 8601	ISODateTime		YYYY-MM-DDThh:mm:ss+03:00
InstructedAmount	0..1	TransactionResponse/Data/Transaction/CurrencyExchange/InstructedAmount	Сумма денег, подлежащая переводу между плательщиком и получателем средств до вычета расходов, выраженная в валюте обозначенной иницирующей стороной	AmountComplexType		
amount	1.1	TransactionResponse/Data/Transaction/CurrencyExchange/InstructedAmount/amount	Сумма	ActiveCurrencyAndAmount_SimpleType		^\d{1,13}\.\d{1,5}\$

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
currency	1.1	TransactionResponse/Data/Transaction/CurrencyExchange/InstructedAmount/currency	Валюта счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
BankTransactionCode	0..1	TransactionResponse/Data/Transaction/BankTransactionCode	Подробная информация для полной идентификации транзакции	BankTransactionCodeStructureComplexType		
code	1.1	TransactionResponse/Data/Transaction/BankTransactionCode/code	Множество внутри домена	ExternalBankTransactionFamily1Code		
subCode	1.1	TransactionResponse/Data/Transaction/BankTransactionCode/subCode	Подмножество внутри множества	ExternalBankTransactionSubFamily1Code		
ProprietaryBankTransactionCode	0..1	TransactionResponse/Data/Transaction/ProprietaryBankTransactionCode	Подробная информация для полной идентификации собственного банковского кода транзакции	ProprietaryBankTransactionCodeStructureComplexType		
code	1.1	TransactionResponse/Data/Transaction/ProprietaryBankTransactionCode/code	Собственный банковский код транзакции	Max35Text		
issuer	0..1	TransactionResponse/Data/Transaction/ProprietaryBankTransactionCode/issuer	Идентификация эмитента собственного банковского кода транзакции	Max35Text		
Balance	0..1	TransactionResponse/Data/Transaction/Balance	Детальная информация о балансе счета	TransactionCashBalanceComplexType		
creditDebitIndicator	1.1	TransactionResponse/Data/Transaction/Balance/creditDebitIndicator	Определяет, является баланс кредитовым или дебетовым	CreditDebitIndicatorStaticType	Credit Debit	
type	1.1	TransactionResponse/Data/Transaction/Balance/type	Тип баланса	BalanceTypeStaticType	ClosingAvailable ClosingBooked ClosingCleared Expected ForwardAvailable Information InterimAvailable InterimBooked InterimCleared OpeningAvailable OpeningBooked OpeningCleared PreviouslyClosedBooked	

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Amount	1..1	TransactionResponse/Data/Transaction/Balance/Amount	Детали баланса	AmountComplexType		
amount	1..1	TransactionResponse/Data/Transaction/Balance/Amount/amount	Сумма	ActiveCurrencyAndAmount_SimpleType		^d{1,13}\.d{1,5}\$
currency	1..1	TransactionResponse/Data/Transaction/Balance/Amount/currency	Валюта счета. Используется стандарт ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
MerchantDetails	0..1	TransactionResponse/Data/Transaction/MerchantDetails	Детали продавца, участвующего в сделке	MerchantDetailsComplexType		
merchantName	0..1	TransactionResponse/Data/Transaction/MerchantDetails/merchantName	Наименование продавца	Max350Text		
merchantCategoryCode	0..1	TransactionResponse/Data/Transaction/MerchantDetails/merchantCategoryCode	Код категории относится к типу услуг или товаров, которые продавец предоставляет	Min3Max4Text		
CreditorAgent	0..1	TransactionResponse/Data/Transaction/CreditorAgent	Финансовая организация, обслуживающая счет получателя средств	BranchAndFinancialInstitutionIdentificationComplexType		
schemeName	0..1	TransactionResponse/Data/Transaction/CreditorAgent/schemeName	Название схемы	FinancialInstitutionIdentificationDynamicType		
identification	0..1	TransactionResponse/Data/Transaction/CreditorAgent/identification	Идентификатор	Max35Text		
name	0..1	TransactionResponse/Data/Transaction/CreditorAgent/name	Название	Max140Text		
Address	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress	Адрес финансовой организации, обслуживающей счет получателя средств	PostalAddressComplexType		
addressType	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/addressType	Тип адреса	AddressTypeStaticType	Business Correspondence DeliveryTo MailTo POBox Postal Residential Statement	
department	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/department		Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
subDepartment	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/subDepartment		Max70Text		
streetName	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/streetName		Max70Text		
buildingNumber	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/buildingNumber		Max16Text		
postCode	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/postCode		Max16Text		
townName	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/townName		Max35Text		
countrySubDivision	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/countrySubDivision		Max35Text		
country	0..1	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/country	Страна. Справочник кодов ISO 3166, Alpha-3 code	CountryCode		^[A-Z]{2,2}\$
addressLine	0..7	TransactionResponse/Data/Transaction/CreditorAgent/PostalAddress/addressLine		Max70Text		
CreditorAccount	0..1	TransactionResponse/Data/Transaction/CreditorAccount	Идентификация счета получателя средств в случае дебетовой транзакции	CashAccountComplexType		
schemeName	0..1	TransactionResponse/Data/Transaction/CreditorAccount/schemeName	Название схемы	AccountIdentificationDynamicType		
identification	0..1	TransactionResponse/Data/Transaction/CreditorAccount/identification	Идентификатор	Max256Text		
name	0..1	TransactionResponse/Data/Transaction/CreditorAccount/name	Название	Max70Text		
DebtorAgent	0..1	TransactionResponse/Data/Transaction/DebtorAgent	Финансовая организация, обслуживающая счет плательщика	BranchAndFinancialInstitutionIdentificationComplexType		
schemeName	0..1	TransactionResponse/Data/Transaction/DebtorAgent/schemeName	Название схемы	FinancialInstitutionIdentificationDynamicType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
identification	0..1	TransactionResponse/Data/Transaction/DebtorAgent/identification	Идентификатор	Max35Text		
name	0..1	TransactionResponse/Data/Transaction/DebtorAgent/name	Название	Max140Text		
Address	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress	Адрес финансовой организации, обслуживающей счет плательщика	PostalAddressComplexType		
addressType	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/addressType		AddressTypeStaticType	Business Correspondence DeliveryTo MailTo POBox Postal Residential Statement	
department	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/department		Max70Text		
subDepartment	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/subDepartment		Max70Text		
streetName	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/streetName		Max70Text		
buildingNumber	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/buildingNumber		Max16Text		
postCode	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/postCode		Max16Text		
townName	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/townName		Max35Text		
countrySubDivision	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/countrySubDivision		Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
country	0..1	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/country		CountryCode		^[A-Z]{2,2}\$
addressLine	0..7	TransactionResponse/Data/Transaction/DebtorAgent/PostalAddress/addressLine		Max70Text		
DebtorAccount	0..1	TransactionResponse/Data/Transaction/DebtorAccount	Идентификация счета плательщика в случае кредитной операции	CashAccountComplexType		
schemeName	0..1	TransactionResponse/Data/Transaction/DebtorAccount/schemeName	Название схемы	AccountIdentificationDynamicType		
identification	0..1	TransactionResponse/Data/Transaction/DebtorAccount/identification	Идентификатор	Max256Text		
name	0..1	TransactionResponse/Data/Transaction/DebtorAccount/name	Название	Max70Text		
CardInstrument	0..1	TransactionResponse/Data/Transaction/CardInstrument	Детальное описание карточного инструмента, использованного в транзакции	TransactionCardInstrumentComplexType		
cardSchemeName	1..1	TransactionResponse/Data/Transaction/CardInstrument/cardSchemeName	Название схемы	CardSchemeNameStaticType	AmericanExpress Diners Discover MasterCard VISA	
authorisationType	0..1	TransactionResponse/Data/Transaction/CardInstrument/authorisationType	Тип авторизации	CardAuthorisationTypeStaticType	ConsumerDevice Contactless None PIN	
name	0..1	TransactionResponse/Data/Transaction/CardInstrument/name	Имя владельца	Max70Text		
identification	0..1	TransactionResponse/Data/Transaction/CardInstrument/identification	Идентификационный номер	Max34Text		

Таблица 58. Состав данных объекта TransactionResponse

### 6.9.3. Примеры использования

#### 6.9.3.1. Получение списка транзакции по идентификатору счета

Запрос получения списка транзакций. СПИУ выполняет к ППУ запрос получения списка транзакций на счете с идентификатором (accountId) “87659” вызовом конечной точки GET /accounts/87659/transactions без параметров.

Ответ получения списка транзакций. ППУ возвращает СПИУ информацию о транзакциях на указанном счете (одна кредитовая транзакция “transactionId”: “234” в размере 1000,00 рублей).

```
GET /accounts/87659/transactions HTTP/1.1
Authorization: Bearer Lz667AOJsdfg
x-fapi-auth-date: Sun, 15 Sep 2019 21:22:31 GMT
x-fapi-customer-ip-address: 10.5.212.11
x-fapi-interaction-id: 23bac548-d2de-7633-b106-880a5018460d
Accept: application/json
```

Таблица 59. Запрос получения списка транзакций

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 23bac548-d2de-7633-b106-880a5018460d
Content-Type: application/json

{
  "Data": {
    "Transaction": [
      {
        "accountId": "87659",
        "transactionId": "234",
        "transactionReference": "Ref 1",
        "Amount": {
          "amount": "1000.00",
          "currency": "RUB"
        },
        "creditDebitIndicator": "Credit",
        "status": "Booked",
        "bookingDateTime": "2019-09-15T7:33:07+00:00",
        "valueDateTime": "2019-09-15T7:35:07+00:00",
        "transactionInformation": "Деньги от Льва",
        "BankTransactionCode": {
          "code": "ReceivedCreditTransfer",
          "subCode": "DomesticCreditTransfer"
        },
        "ProprietaryBankTransactionCode": {
          "code": "Transfer",
          "issuer": "AlphaBank"
        },
        "Balance": {
          "Amount": {
            "amount": "10000.00",
            "currency": "RUB"
          },
          "CreditDebitIndicator": "Credit",
          "Type": "OpeningAvailable"
        }
      }
    ]
  },
  "Links": {
    "Self": "https://bank.ru/open-banking/v3.1/aisp/accounts/87659/transactions/"
  },
  "Meta": {
    "TotalPages": 1,
    "FirstAvailableDateTime": "2019-09-15T00:00:00+00:00",
    "LastAvailableDateTime": "2019-09-15T21:00:00+00:00"
  }
}
```

Таблица 60. Ответ получения списка транзакций

Получение списка транзакций по всем авторизованным счетам Пользователя.

В данном примере транзакции приходят по всем счетам, к которым Пользователь авторизовал согласие.

Запрос получения списка транзакций. СПИУ выполняет к ППУ запрос получения списка транзакций по всем счетам, к которым Пользователь авторизовал согласие вызовом конечной точки GET /transactions без параметров.

Ответ получения списка транзакций. ППУ возвращает СПИУ список счетов с информацией о транзакциях:

“accountId”: “12345” одна кредитовая транзакция “transactionId”: “123” в размере 100,00 рублей;  
“accountId”: “98765” одна дебетовая транзакция “transactionId”: “345” в размере 100,00 рублей.

```
GET /accounts/87659/transactions HTTP/1.1
Authorization: Bearer Lz667AOJsdfg
x-fapi-auth-date: Sun, 15 Sep 2019 21:22:31 GMT
x-fapi-customer-ip-address: 10.5.212.11
x-fapi-interaction-id: 23bac548-d2de-7633-b106-880a5018460d
Accept: application/json
```

Таблица 61. Запрос получения списка транзакций

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 23bac548-d2de-7633-b106-880a5018460d
Content-Type: application/json
```

```
{
  "Data": {
    "Transaction": [
      {
        "AccountId": "12345",
        "TransactionId": "123",
        "TransactionReference": "Ref 123",
        "Amount": {
          "Amount": "100.00",
          "Currency": "RUB"
        },
        "CreditDebitIndicator": "Credit",
        "Status": "Booked",
        "BookingDateTime": "2019-09-15T10:43:07+00:00",
        "ValueDateTime": "2019-09-15T10:45:22+00:00",
        "TransactionInformation": "Деньги от друга",
        "BankTransactionCode": {
          "Code": "ReceivedCreditTransfer",
          "SubCode": "DomesticCreditTransfer"
        },
        "ProprietaryBankTransactionCode": {
          "Code": "Transfer",
          "Issuer": "Bank"
        },
        "Balance": {
          "Amount": {
            "Amount": "230.00",
            "Currency": "RUB"
          },
          "CreditDebitIndicator": "Credit",
          "Type": "OpeningAvailable"
        }
      },
      {
        "AccountId": "98765",
        "TransactionId": "345",
        "TransactionReference": "Ref 124",
        "Amount": {
          "Amount": "100.00",
          "Currency": "GBP"
        },
        "CreditDebitIndicator": "Debit",
        "Status": "Booked",
        "BookingDateTime": "2019-09-15T14:22:09+00:00",
```

```

    "ValueDateTime": "2019-09-15T14:22:09+00:00",
    "TransactionInformation": "Оплата коммунальных услуг",
    "AddressLine": "Шлюзовая",
    "BankTransactionCode": {
      "Code": "IssuedCreditTransfer",
      "SubCode": "AutomaticTransfer"
    },
    "ProprietaryBankTransactionCode": {
      "Code": "DirectDebit",
      "Issuer": "Bank"
    },
    "Balance": {
      "Amount": {
        "Amount": "1257.36",
        "Currency": "RUB"
      },
      "CreditDebitIndicator": "Debit",
      "Type": "OpeningAvailable"
    }
  }
]
},
"Links": {
  "Self": "https://bank.ru/open-banking/v3.1/aisp/transactions/"
},
"Meta": {
  "TotalPages": 1,
  "FirstAvailableDateTime": "2019-09-15T00:00:00+00:00",
  "LastAvailableDateTime": "2019-09-15T21:00:00+00:00"
}
}

```

Таблица 62. Ответ получения списка транзакций

**6.9.3.2. Отказано в доступе**

СПИУ не имеет доступа для вызова конечной точки списка транзакций.

Запрос получения списка транзакций. СПИУ выполняет к ППУ запрос получения списка транзакций на счете с идентификатором (accountId) "87659" вызовом конечной точки GET /accounts/87659/transactions без параметров.

Ответ получения списка транзакций. ППУ возвращает СПИУ ошибку 403 – отказ в доступе.

```

GET /accounts/87659/transactions HTTP/1.1
Authorization: Bearer Lz667AOJsdfg
x-fapi-auth-date: Sun, 15 Sep 2019 21:22:31 GMT
x-fapi-customer-ip-address: 10.5.212.11
x-fapi-interaction-id: 23bac548-d2de-7633-b106-880a5018460d
Accept: application/json

```

Таблица 63. Запрос получения списка транзакций

```

HTTP/1.1 403 Forbidden
x-fapi-interaction-id: 23bac548-d2de-7633-b106-880a5018460d

```

Таблица 64. Ответ получения списка транзакций

## 6.10. Выписки, спецификация API – v1.2.1

### 6.10.1. Конечные точки

№	Ресурс	Метод HTTP	Конечная точка	Scope	Grant Type	Ключ идиempотентности	Параметры	Объект запроса	Объект ответа
1	statements	POST	POST /statements/{accountId}	accounts	Authorization Code	Да		StatementInitRequest	StatementInitResponse
2	statements	GET	GET /accounts/{accountId}/statements/{statementId}	accounts	Authorization Code	Нет	Нумерация страниц Фильтрация		StatementResponse
3	statements	GET	GET /statements	accounts	Authorization Code	Нет	Нумерация страниц Фильтрация		StatementResponse

Таблица 65. Конечные точки ресурса выписка

#### 6.10.1.1. POST /statements

ППУ предоставляет конечную точку СПИУ для получения идентификатора выписки statementId по идентификатору счета accountId, который предоставляется при вызове конечной точки GET /accounts.

#### 6.10.1.2. GET /accounts/{accountId}/statements/{statementId}

ППУ предоставляет конечную точку СПИУ для получения выписки по идентификатору счета и идентификатору выписки.

#### 6.10.1.3. GET /statements

ППУ предоставляет конечную точку СПИУ для получения выписок по всем авторизованным Пользователем счетам.

## 6.10.2. Модель данных

Объект `StatementInitRequest` используется в запросе конечной точки:

- POST /statements

Объект `StatementInitResponse` используется в ответе конечной точки:

- POST /statements

Объект `StatementResponse` используется в ответах конечных точек:

- GET /accounts/{accountId}/statements/{statementId}
- GET /statements

### 6.10.2.1. Описание ресурсов

Данный ресурс представляет собой выписку по счету с уникальным идентификатором (`accountId`) в определенный момент времени или выписки по всем авторизованным Пользователем счетам в определенный момент времени.

### 6.10.2.2. UML-диаграмма

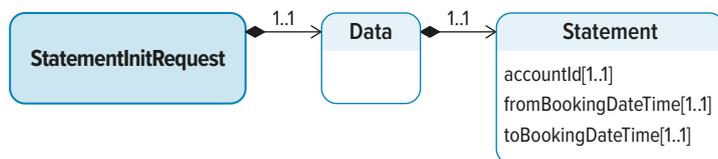


Рисунок 11. Диаграмма объекта `StatementInitRequest`

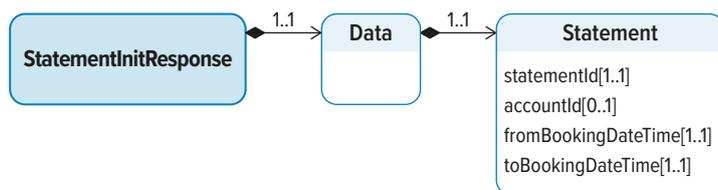


Рисунок 12. Диаграмма объекта `StatementInitResponse`

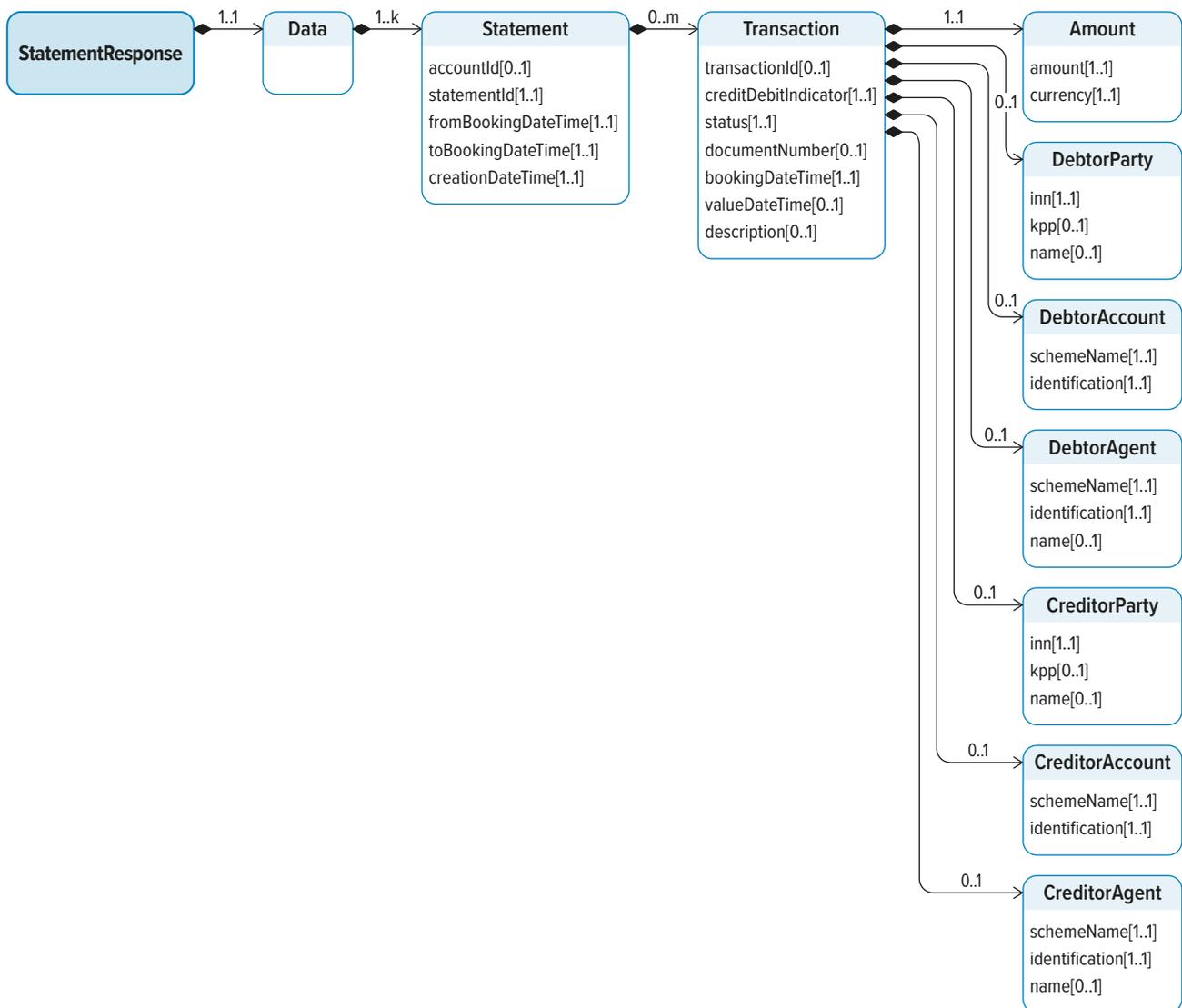


Рисунок 13. Диаграмма объекта StatementResponse

Все ППУ предоставляют поле “BookingDateTime” для разбивки на страницы и для фильтрации, поэтому поле является обязательным для пересылки. BookingDateTime — это дата, когда транзакция забронирована (или проведена) и становится неизменной, а не дата совершения транзакции.

### 6.10.2.3. Фильтрация

Для ограничения списка транзакций, которые возвращает ППУ, используются следующие параметры запроса при обращении к ресурсу:

Наименование	Кратность	Описание	Тип данных
fromBookingDateTime	0..1	Дата и время начала фильтрации списка транзакций	ISODateTime
toBookingDateTime	0..1	Дата и время окончания фильтрации списка транзакций	ISODateTime

Таблица 66. Фильтрация транзакций для выписки

ППУ рассматривает следующее как допустимый ввод:

- Нерабочие дни (например, воскресенье или праздничные дни) или любые другие дни, когда транзакции не регистрируются.
- Даты, выходящие за пределы диапазона, для которого информация о транзакциях предоставляется через API.
- Даты, выходящие за пределы диапазона, для которого Пользователь авторизовал согласие.
- Часовой пояс может быть включен в запрос фильтра, но игнорируется ППУ. ППУ использует локальный часовой пояс.

#### 6.10.2.4. Состав данных объекта StatementInitRequest

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
<b>StatementInitRequest</b>		<b>StatementInitRequest</b>		<b>StatementInitRequestComplexType</b>		
<b>Data</b>	<b>1.1</b>	<b>StatementInitRequest/Data</b>		<b>DataStatementInitRequestComplexType</b>		
<b>Statement</b>	<b>1.1</b>	<b>StatementInitRequest/Data/Statement</b>		<b>StatementInitReqComplexType</b>		
accountId	1.1	StatementInitRequest/Data/Statement/accountId	Идентификатор ресурса счета	Max40Text		
fromBookingDateTime	1.1	StatementInitRequest/Data/Statement/fromBookingDateTime	Дата и время начала выписки	ISODateTime		YYYY-MM-DDThh:mm:ss+03:00
toBookingDateTime	1.1	StatementInitRequest/Data/Statement/ toBookingDateTime	Дата и время окончания выписки	ISODateTime		YYYY-MM-DDThh:mm:ss+03:00

Таблица 67. Состав данных объекта StatementInitRequest

#### 6.10.2.5. Состав данных объекта StatementInitResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
<b>StatementInitResponse</b>		<b>StatementInitResponse</b>		<b>StatementInitResponseComplex Type</b>		
<b>Data</b>	<b>1.1</b>	<b>StatementInitResponse/Data</b>		<b>DataStatementInitResponseComplexType</b>		
<b>Statement</b>	<b>1.1</b>	<b>StatementInitResponse/Data/Statement</b>		<b>StatementInitRespComplex</b>		
accountId	0..1	StatementInitResponse/Data/Statement/accountId	Идентификатор ресурса счета	Max40Text		
statementId	1.1	StatementInitResponse/Data/Statement/statementId	Идентификатор ресурса выписки	Max40Text		
fromBookingDateTime	1.1	StatementInitResponse/Data/Statement/fromBookingDateTime	Дата и время начала выписки	ISODateTime		YYYY-MM-DDThh:mm:ss+03:00
toBookingDateTime	1.1	StatementInitResponse/Data/Statement/ toBookingDateTime	Дата и время окончания выписки	ISODateTime		YYYY-MM-DDThh:mm:ss+03:00

Таблица 68. Состав данных объекта StatementInitResponse

### 6.10.2.6. Состав данных объекта StatementResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
<b>StatementResponse</b>		<b>StatementResponse</b>		<b>StatementResponseComplexType</b>		
<b>Data</b>	<b>1.1</b>	<b>StatementResponse/Data</b>		<b>DataStatementResponseComplexType</b>		
<b>Statement</b>	<b>1..k</b>	<b>StatementResponse/Data/Statement</b>		<b>StatementComplexType</b>		
accountId	0..1	StatementResponse/Data/Statement/accountId	Идентификатор ресурса счета	Max40Text		
statementId	1..1	StatementResponse/Data/Statement/statementId	Идентификатор ресурса выписки	Max40Text		
fromBookingDateTime	1..1	StatementResponse/Data/Statement/fromBookingDateTime	Дата и время начала выписки	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
toBookingDateTime	1..1	StatementResponse/Data/Statement/toBookingDateTime	Дата и время окончания выписки	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
creationDateTime	1..1	StatementResponse/Data/Statement/creationDateTime	Дата и время создания ресурса	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
<b>Transaction</b>	<b>0..m</b>	<b>StatementResponse/Data/Statement/Transaction</b>		<b>TransactionComplexType</b>		
transactionId	0..1	StatementResponse/Data/Statement/Transaction/transactionId	Уникальный идентификатор транзакции	Max210Text		
creditDebitIndicator	1..1	StatementResponse/Data/Statement/Transaction/creditDebitIndicator	Приход/Уход	CreditDebitIndicatorStaticType	Credit Debit	
status	1..1	StatementResponse/Data/Statement/Transaction/status	Статус транзакции	TransactionStatusStaticType	Booked Pending	
documentNumber	0..1	StatementResponse/Data/Statement/Transaction/documentNumber	Номер платежного документа	Max6Text		
bookingDateTime	1..1	StatementResponse/Data/Statement/Transaction/bookingDateTime	Дата и время, когда запись о транзакции публикуется на счете в бухгалтерской книге обслуживающей организации. Используется стандарт ISO 8601	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
valueDateTime	0..1	StatementResponse/Data/Statement/Transaction/valueDateTime	Дата и время, когда активы становятся доступными владельцу счета в случае ввода кредита или перестают быть доступными владельцу счета в случае ввода дебетовой транзакции. Используется стандарт ISO 8601	ISODatetime		YYYY-MM-DDThh:mm:ss+03:00
description	0..1	StatementResponse/Data/Statement/Transaction/description	Назначение перевода денежных средств	Max300Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
<b>Amount</b>	<b>1.1</b>	<b>StatementResponse/Data/Statement/Transaction/Amount</b>	<b>Информация о сумме и валюте транзакции</b>	<b>AmountComplexType</b>		
amount	1.1	StatementResponse/Data/Statement/Transaction/Amount/amount	Сумма транзакции по счету запроса в рублях по курсу Банка России на дату транзакции	ActiveCurrencyAndAmount_SimpleType		^\d{1,13}\.\d{1,5}\$
currency	1.1	StatementResponse/Data/Statement/Transaction/Amount/currency	Валюта счета ISO 4217	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
<b>DebtorParty</b>	<b>0..1</b>	<b>StatementResponse/Data/Statement/Transaction/DebtorParty</b>	<b>Информация о контрагенте в случае кредитной операции</b>	<b>PartyIdentificationComplexType</b>		
inn	1.1	StatementResponse/Data/Statement/Transaction/DebtorParty/inn	ИНН контрагента	Max12Text		
name	0..1	StatementResponse/Data/Statement/Transaction/DebtorParty/name	Наименование контрагента	Max160Text		
kpp	0..1	StatementResponse/Data/Statement/Transaction/DebtorParty/kpp	КПП контрагента	Max9Text		
<b>DebtorAccount</b>	<b>0..1</b>	<b>StatementResponse/Data/Statement/Transaction/DebtorAccount</b>	<b>Идентификация счета плательщика в случае кредитной операции</b>	<b>CashAccountComplexType</b>		
schemeName	1.1	StatementResponse/Data/Statement/Transaction/DebtorAccount/schemeName	Название схемы	AccountIdentificationDynamicType		
identification	1.1	StatementResponse/Data/Statement/Transaction/DebtorAccount/identification	Идентификатор счета	Max256Text		
<b>DebtorAgent</b>	<b>0..1</b>	<b>StatementResponse/Data/Statement/Transaction/DebtorAgent</b>	<b>Финансовая организация, обслуживающая счет плательщика</b>	<b>BranchAndFinancialInstitutionIdentificationComplexType</b>		
schemeName	0..1	StatementResponse/Data/Statement/Transaction/DebtorAgent/schemeName	БИК/SWIFT банка агента	FinancialInstitutionIdentificationDynamicType		
identification	0..1	StatementResponse/Data/Statement/Transaction/DebtorAgent/identification	БИК/SWIFT банка агента	Max35Text		
name	0..1	StatementResponse/Data/Statement/Transaction/DebtorAgent/name	Наименование банка агента	Max160Text		
<b>CreditorParty</b>	<b>0..1</b>	<b>StatementResponse/Data/Statement/Transaction/CreditorParty</b>	<b>Информация о контрагенте в случае дебетовой транзакции</b>	<b>PartyIdentificationComplexType</b>		
inn	1.1	StatementResponse/Data/Statement/Transaction/CreditorParty/inn	ИНН контрагента	Max12Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
name	0..1	StatementResponse/Data/Statement/Transaction/CreditorParty/name	Наименование контрагента	Max160Text		
kpp	0..1	StatementResponse/Data/Statement/Transaction/CreditorParty/kpp	КПП контрагента	Max9Text		
<b>CreditorAccount</b>	<b>0..1</b>	<b>StatementResponse/Data/Statement/Transaction/CreditorAccount</b>	<b>Идентификация счета получателя средств в случае дебетовой транзакции</b>	<b>CashAccountComplexType</b>		
schemeName	1..1	StatementResponse/Data/Statement/Transaction/CreditorAccount/schemeName	Название схемы контрагента	AccountIdentificationDynamicType		
identification	1..1	StatementResponse/Data/Statement/Transaction/CreditorAccount/identification	Идентификатор счета контрагента	Max256Text		
<b>CreditorAgent</b>	<b>0..1</b>	<b>StatementResponse/Data/Statement/Transaction/CreditorAgent</b>	<b>Финансовая организация, обслуживающая счет получателя средств</b>	<b>BranchAndFinancialInstitutionIdentificationComplexType</b>		
schemeName	0..1	StatementResponse/Data/Statement/Transaction/CreditorAgent/schemeName	БИК/SWIFT банка контрагента	FinancialInstitutionIdentificationDynamicType		
identification	0..1	StatementResponse/Data/Statement/Transaction/CreditorAgent/identification	БИК/SWIFT банка контрагента	Max35Text		
name	0..1	StatementResponse/Data/Statement/Transaction/CreditorAgent/name	Наименование банка контрагента	Max160Text		

Таблица 69. Состав данных объекта StatementResponse

## 7. Динамические справочники в пространствах имен – v1.2.1

### 7.1. Введение

Спецификация определяет поля только с фиксированным набором возможных значений, а дальнейшее добавление значений требует изменения спецификации.

В рамках текущей версии определены новые типы данных, представляющие собой расширяемый список значений. Любые расширения этого стандартного списка значений могут быть сделаны ППУ с соответствующей документацией на их порталах для разработчиков.

Значения расширяемых типов данных располагаются в соответствующих пространствах имен, чтобы помочь идентифицировать источник значения и соответствующее значение.

### 7.2. Основы

Элементы справочников, определенные участниками Ассоциации развития финансовых технологий, задокументированы здесь и будут иметь префикс RU.CBR.

#### 7.2.1. Принципы проектирования

При добавлении собственных значений в справочники участники пространства Открытых банковских интерфейсов с ролью ППУ помещают такие значения в пространство имен, состоящее из двухбуквенного кода страны (код ISO 3166-1 Alpha-2), после которого ставится точка, после которой следует наименование организации. Например:

- RU.SBRF.Int-payments
- RU.VTB.Int-payments
- KE.Safaricom.M-Pesa

#### 7.2.2. Управление версиями

В данном разделе будут описываться требования к обратной совместимости при разработке последующих версий Стандарта.

### 7.3. Общие справочники

#### AccountIdentificationDynamicType

Значение	Описание
RU.CBR.PAN	Primary Account Number — схема идентификатора, используемая для идентификации карточного счета
RU.CBR.CellphoneNumber	Схема для осуществления перевода денежных средств по номеру телефона
RU.CBR.BBAN	Схема для осуществления перевода денежных средств по номеру счета

Таблица 70. Идентификатор счета

#### FinancialInstitutionIdentificationDynamicType

Значение	Описание
RU.CBR.BICFI	BIC для финансовых учреждений согласно ISO 9362.
RU.CBR.BIK	Уникальный идентификатор банка, используемый в платежных документах на территории Российской Федерации

Таблица 71. Идентификатор финансового учреждения

**OBRUErrorResponseErrorCode**

Тип данных, который дает низкоуровневый текстовый код ошибки для ее классификации. Применяется также код ответа HTTP.

Значение	HTTP-статус	Описание
RU.CBR.Field.Expected	400	Если поля передаются парой (ключ-значение) и значение не было передано. В поле path должен передаваться путь к ожидаемому полю (например, ErrorResponse.Errors.path == "AccountResponse.Data.Account.AccountDetails.identification"). Например, для допустимого значения поля "schemeName" должно передаваться соответствующее значение идентификатора в поле "identification"
RU.CBR.Field.Invalid	400	В поле указано недопустимое значение или длина предоставленного значения превышает соответствующую максимальную длину поля в домене ППУ. Ссылка на недопустимое поле должна быть указана в поле path (например, ErrorResponse.Errors.path == "AccountResponse.Data.Account.AccountDetails.schemeName"). В поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение. Проблема должна быть подробно описана в сообщении об ошибке (поле ErrorResponse.Errors.message)
RU.CBR.Field.InvalidDate	400	Указана неверная дата. Например, когда ожидается будущая дата, а указана дата в прошлом или текущая дата. В сообщении можно указать актуальную проблему с датой. Ссылка на недопустимое поле должна быть указана в поле path, а в поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение
RU.CBR.Field.Missing	400	Обязательное поле, необходимое для API, отсутствует в полезной нагрузке. Данный код ошибки можно использовать, если ошибка еще не определена при проверке RU.CBR.Resource.InvalidFormat. Ссылка на отсутствующее поле должна быть указана в поле path, а в поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение
RU.CBR.Header.Invalid	400	В элементе заголовка HTTP указано неверное значение. Элемент заголовка HTTP должен быть указан в элементе пути
RU.CBR.Header.Missing	400	Обязательный элемент HTTP-заголовка не был предоставлен. Элемент заголовка HTTP должен быть указан в элементе path
RU.CBR.Resource.ConsentMismatch	400	Несоответствие ресурсов "payment-consent" и "payment". Например, если элемент в разделе "Initiation" или "Risk" ресурса платежа не совпадает с одноименным элементом в соответствующем разделе ресурса согласия. Элемент пути должен быть заполнен элементом ресурса платежа, который не соответствует согласию
RU.CBR.Resource.InvalidConsentStatus	400	Согласие, соответствующее ресурсу, находится в некорректном статусе, который бы позволил создать ресурс или выполнить запрос. Например, если ресурс согласия имеет статус AwaitingAuthorisation или Rejected, то ресурс не может быть создан с таким статусом соответствующего ему согласия. Элемент пути должен быть заполнен элементом ресурса согласия, который является недопустимым
RU.CBR.Resource.InvalidFormat	400	Когда json-схема полезной нагрузки не соответствует конечной точке. Например, конечная точка POST /payments вызывается с полезной нагрузкой JSON, которая не может быть проанализирована в классе PaymentRequest
RU.CBR.Resource.NotFound	400	Возвращается, когда ресурс с указанным идентификатором не существует (ресурс не может быть обработан)
RU.CBR.Resource.NotCreated	400	Возвращается, когда ресурс с указанным идентификатором еще не создан и не может быть передан в ответном сообщении. Для асинхронных вызовов. Например, получение выписки по счету, где сначала создается ресурс выписки (метод POST /statements/{accountId}) и в ответном сообщении приходит идентификатор созданного ресурса выписки, но для наполнения выписки данными ППУ требуется некоторое время. Соответственно будет приходиться данное сообщение об ошибке
RU.CBR.Rules.AfterCutOffDateTime	400	Ресурс согласия или ресурс платежа запрашивается после даты CutOffDateTime
RU.CBR.Signature.Invalid	400	Заголовок подписи x-jws-signature был проанализирован и имеет действительный заголовок JOSE, соответствующий спецификации. Но сама подпись не может быть проверена

Значение	HTTP-статус	Описание
RU.CBR.Signature.InvalidClaim	400	Заголовок JOSE в элементе x-jws-signature имеет одно или несколько утверждений (claim) с недопустимым значением (например, утверждение kid, которое не принимает сертификат). Наименование отсутствующего утверждения должно передаваться в поле path ответа об ошибке
RU.CBR.Signature.MissingClaim	400	Заголовок JOSE в элементе x-jws-signature имеет одно или несколько обязательных утверждений, которые не указаны. Имя пропущенного утверждения должно быть указано в поле path ответа об ошибке
RU.CBR.Signature.Malformed	400	x-jws-signature в заголовке запроса была искажена и не могла быть проанализирована как допустимый JWS
RU.CBR.Signature.Missing	400	Запрос API предполагает x-jws-signature в заголовке, но элемент отсутствовал
RU.CBR.Unsupported.AccountIdentifier	400	Идентификатор счета не поддерживается для данной схемы. Элемент path должен быть заполнен путем к элементу accountIdentifier
RU.CBR.Unsupported.LocalInstrument	400	Указанный localInstrument не поддерживается ППУ. Элемент path должен быть заполнен путем к элементу localInstrument. Элемент URL должен быть заполнен ссылкой на документацию ППУ со списком поддерживаемых localInstrument
RU.CBR.Reauthenticate	403	Данный код ошибки указывает, что для обработки запроса требуется повторная аутентификация Пользователя
RU.CBR.Rules.ResourceAlreadyExists	409	Данный код ошибки указывает, что ресурс с такими же параметрами уже существует
RU.CBR.UnexpectedError	5xx	Данный код ошибки можно использовать при возникновении непредвиденной ошибки. ППУ должен заполнить сообщение детальным описанием ошибки, не раскрывая конфиденциальную информацию

Таблица 72. Низкоуровневая классификация ошибок