

Единая система приема платежей (ЕСПП)

**СПЕЦИФИКАЦИЯ ПРОТОКОЛА ВЗАИМОДЕЙСТВИЯ
ЕДИНОЙ СИСТЕМЫ ПРИЕМА ПЛАТЕЖЕЙ
ОАО «РОСТЕЛЕКОМ» С ИНФОРМАЦИОННОЙ СИСТЕМОЙ АГЕНТА**

действует с момента подписания

Редакция 1.7

Оглавление

АННОТАЦИЯ 3

ИСТОРИЯ ИЗМЕНЕНИЙ 3

1 ВВЕДЕНИЕ 4

1.1	Термины и сокращения.....	4
1.1.1	Термины.....	4
1.1.2	Сокращения.....	4
1.2	Назначение ЕСПП.....	5
1.3	Задачи протокола взаимодействия.....	5
1.4	Состав спецификации протокола.....	5
1.5	Содержание спецификации.....	6

2 ОРГАНИЗАЦИЯ ВЗАИМОДЕЙСТВИЯ И ПОДКЛЮЧЕНИЯ 7

2.1	Организация взаимодействия.....	7
2.2	Организация подключения.....	7

3 ВИДЫ ВЗАИМОДЕЙСТВИЯ И УПРАВЛЕНИЕ ПЛАТЕЖАМИ 8

3.1	Виды взаимодействия.....	8
3.2	Жизненный цикл платежа.....	9
3.3	Отмена платежей.....	11
3.4	Идентификация платежей.....	11
3.5	Идентификация лицевого счета.....	11
3.6	Валюты и денежные единицы.....	11
3.7	Время платежа.....	12
3.8	Назначение платежа.....	12
3.9	Отложенная обработка.....	12
3.10	Сверки.....	12
3.11	Повтор платежа.....	13
3.12	Детализация платежа.....	13
3.13	Счета учета платежей.....	13

4 ФОРМАТ ЗАПРОСОВ И ОТВЕТОВ 14

4.1	Использование HTTP.....	14
4.2	Формирование ответа сервера.....	14
4.3	URL-кодирование.....	14
4.4	Передача массивов.....	14
4.5	Передача табличных данных.....	15
4.6	Типы данных.....	15
4.7	Поля ответа на запросы.....	16
4.8	Статусы выполнения запросов.....	17
4.9	Статусы платежей.....	17

5 ОПИСАНИЕ ЗАПРОСОВ И ОТВЕТОВ 18

5.1	Запрос на проверку параметров зачисления.....	18
5.2	Запрос на зачисление средств.....	19
5.3	Запрос на отмену зачисления.....	20
5.4	Чтение статуса.....	21
5.5	Пакетное чтение статуса.....	22
5.6	Запрос на получение информации о ЛС/Абоненте.....	24
5.7	Запрос на поиск информации о ЛС абонента по адресу.....	26

Аннотация

Данный документ является спецификацией протокола взаимодействия Платежного агента с Единой системой приема платежей ОАО "Ростелеком" (ЕСПП) – протокол ПА-ЕСПП. Протокол включает в себя функции обмена сообщениями в формате «запрос-ответ» при проведении платежей, а так же инструменты для проведения сверок.

История изменений

Версия	Изменения	Дата
1.7	Добавлен возврат поле acceptedTime, abandonedTime, payTime в запрос getPaymentStatus, см.п. 5.4	25-07-2013
1.6	Добавлено поле agentAccount в запросы abandonPayment и getPaymentStatus. Добавлено поле dstDepCode, см.пп.5.2, 5.4, 5.5.	12-04-2013
1.5	Внесены уточнения по тексту	31-01-2013
1.4	Добавлено дополнительное значение для поля dupFlag, добавлены коды возврата, убран контроль времени	12-10-2012
1.3	Добавлено уточнение по правилам анализа статусов запросов, см. п.4.7, уточнен content-type для синтаксиса JSON, см.п.4.1, уточнен состав полей операций зачисления и отмены зачисления, см. пп.5.2, 5.3., поиска по адресу, см. п.5.7 Расширен состав дат, возвращаемых функцией пакетного чтения статуса платежей, см.п.5.5 Внесены уточнения	09-08-2012
1.2	Добавлено поле agentAccount, см. п.3.13	30-07-2012
1.1	Внесены уточнения	19-07-2012
1.0	Первая редакция	26-06-2012

1 Введение

1.1 Термины и сокращения

1.1.1 Термины

Термин	Описание
Оператор связи, Оператор	ОАО «Ростелеком»
Клиент	Юридическое или физическое лицо, которому оказываются услуги Оператора или планируется предоставление таких услуг.
Абонент	Клиент, заключивший с оператором связи абонентский договор на приобретение и использование услуг.
АСР	Автоматизированная система расчетов, биллинг. В контексте данного документа, если не оговорено иного, под АСР понимается любая система, подключенная к ЕСПП и ведущая лицевые счета, на которые могут быть зачислены или списаны средства по команде ЕСПП.
Платеж	Зачисление средств на лицевой счет. В контексте данного документа, если не оговорено иного, подразумевается не сам платеж, а лишь информация о факте его совершения, передаваемая через ЕСПП. Фактические взаиморасчеты сторон не входят в зону ответственности ЕСПП.
Платежный агент	Внешняя по отношению к ЕСПП система, уполномоченная принимать платежи за услуги связи в пользу ОАО «Ростелеком»
Зачисление	Вид платежа: операция пополнения лицевого счета в АСР, выполняемая по команде ЕСПП.
Транзакция	В контексте данного документа, если не оговорено иного, под транзакцией понимается операция по выполнению зачисления или по отмене зачисления.

1.1.2 Сокращения

Сокращение	Полное название
АСР	Автоматизированная система расчетов
ПА	Платежный агент
ЕСПП	Единая система приема платежей

Сокращение	Полное название
ПС	Подключенная система – источник или получатель платежей
ЛС	Лицевой счет

1.2 Назначение ЕСПП

Единая система приема платежей (ЕСПП) позволяет передавать информацию о платежах, осуществляемых в счет оплаты услуг связи, как между различными АСР, так и получать такую информацию из «внешних систем» - Агентов, уполномоченных принимать платежи в пользу «Ростелеком». С помощью передачи информации о платежах между различными системами, Абонентам «Ростелеком» предоставляется как возможность оплачивать услуги связи в любых пунктах приема вне зависимости от их расположения, так и возможность оплачивать услуги связи большим числом способов – через карты оплаты, «внешние» по отношению к «Ростелеком» Платежные системы, банковские системы. Фактически, ЕСПП выступает интегрирующим звеном сети различных систем, как внутренних, так и внешних, единообразно передающим по этой сети информацию о платежах.

1.3 Задачи протокола взаимодействия

Функции протокола разделяются на следующие логические группы:

- **Информационные:**
 - получение информации о ЛС/Абоненте;
 - получение списка лицевых счетов по адресу абонента.
- **Зачисление средств на ЛС:**
 - проверка допустимости зачисления;
 - выполнение зачисления;
 - отмена зачисления.
- **Служебные:**
 - чтение текущего статуса одной транзакции;
 - получение списка транзакций по фильтру.

Все перечисленные функции реализуются на стороне ЕСПП, которая выступает по отношению к Агенту в роли сервера, и вызываются со стороны Агента, который выступает по отношению к ЕСПП в роли клиента.

1.4 Состав спецификации протокола

Спецификация описывает взаимодействия ЕСПП и Агента, передаваемые данные и вызываемые функции. В состав спецификации не входят справочники, согласуемые отдельно с каждым Агентом на этапе интеграции:

- справочник областей именования (svcTypeId, см. п. 3.5);
- справочник кодов субсчетов (svcSubNum, см. п. 3.5);
- справочник допустимых назначений платежа (payPurpose, см. п. 3.8);
- справочник дополнительных кодов валют (см. п. 3.6);
- справочник кодов регионов для функции поиска лицевых счетов по адресу.

1.5 Содержание спецификации

Глава 1 содержит вводную информацию;

Глава 2 описывает принципы организации взаимодействия и подключения Агента к ЕСПП;

Глава 3 описывает процедуры взаимодействия, жизненный цикл платежей;

Глава 4 описывает правила кодирования запросов и ответов, коды возврата, коды статусов;

Глава 5 описывает запросы и ответы.

2 Организация взаимодействия и подключения

2.1 Организация взаимодействия

Взаимодействие осуществляется через HTTPS соединения с авторизацией по X.509 сертификатам. Данные запросов и ответов передаются в HTTP POST url encoded в теле запроса/ответа, либо в JSON формате. Подробно формат запросов описан ниже в п.3.13.

Принципиально, схему взаимодействия систем, участвующих в проведении платежей, можно представить следующим образом:

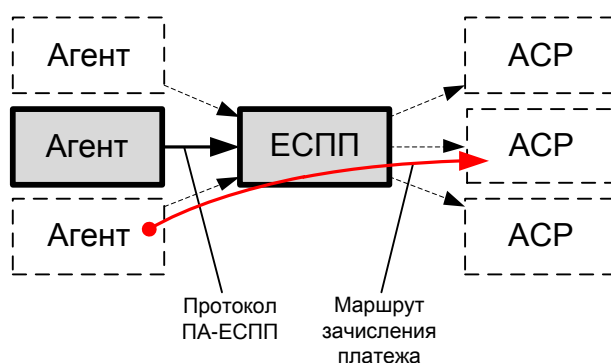


Рис. 1

Источником платежей может выступать как внешняя по отношению к Оператору система («Агент»), так и внутренняя (собственные пункты приема платежей, формы оплаты на сайте и в Едином личном кабинете и т.п.). Команда на проведение платежа передается от источника в ЕСПП, где на основании реквизитов платежа выполняется маршрутизация: определяется АСР, в которой ведется ЛС получателя. После определения АСР, ЕСПП передает команду в нее. Результаты выполнения платежа передаются обратно по цепочке.

Все запросы от Агента к ЕСПП передаются на один и тот же URL, при этом вид запроса (имя вызываемой функции) передается в качестве параметра в теле запроса. Ответ сервера формируется так же в формате HTTP-POST или JSON (выбирается тот же формат, в котором был сделан запрос).

Запросы логически подразделяются на два основных класса: запросы на получение информации и запросы на проведение платежей (подробнее см. п. 3.1).

Для того, чтобы совместить удобство on-line взаимодействия, при котором ответ на запрос возвращается в рамках одной HTTP-транзакции с запросом, и возможность продолжительной обработки запроса, превышающей по времени разумные рамки одной HTTP-транзакции, протокол использует концепцию «отложенной обработки»: если ЕСПП может исполнить запрос в течение 30 секунд, то она возвращает ответ в рамках текущей HTTP-транзакции. В противном случае не позднее чем через 30 секунд после получения запроса, ЕСПП возвращает результат «OK» (ERROR_OK) и продолжает исполнение запроса off-line (подробнее см. п.3.9). «Отложенная обработка» используется только для запросов на проведение платежей, запросы на получение информации всегда исполняются on-line. Сервер ЕСПП параллельно обслуживает несколько запросов от каждого Агента. Количество одновременных запросов настраивается на этапе подключения и может меняться в дальнейшем службой эксплуатации. Типичное значение — 16 одновременных запросов и более.

2.2 Организация подключения

Взаимодействие подключенной системы с ЕСПП осуществляется по открытым каналам связи сети общего пользования Internet. Для обмена используется стек протоколов IP/TCP/HTTPS (HTTP over SSL). Защита передаваемых данных от несанкционированного доступа обеспечивается средствами криптографической библиотеки SSL (secure socket layer).

При подключении производится обмен X.509 сертификатами, которые в дальнейшем используются для организации SSL-соединения. При этом использовать заверенные центром сертификации (Certificate Authority, CA) сертификаты нет необходимости, допустимо использовать самоподписанные (self-signed) сертификаты.

Используются 2 сертификата:

1. Серверный сертификат ЕСПП

2. Клиентский сертификат Агента, выданный с помощью серверного сертификата ЕСПП и используемый Агентом при выполнении запросов ПА→ЕСПП

При подключении Агента к ЕСПП определяются необходимые параметры взаимодействия:

1. IP-адреса и URL. Параметры для подключения к серверу Оператора сообщаются Агенту отдельным документом и включают в себя полный URL ([https://<domain>\[:<port>\]/path](https://<domain>[:<port>]/path)), например: https://domainname.ru/_path01/. С Агентом согласуется пул IP адресов, с которых допускается обращение к серверу Оператора. Обращение с других IP не допускается и может привести к автоматической блокировке интерфейса на стороне Оператора.
2. На стороне ЕСПП настраивается ограничение на количество одновременных запросов, которые Агент может выполнять к ЕСПП.
3. На стороне ЕСПП настраивается допустимое отклонение времени от времени Агента (см. п.3.7).

При подключении Агент получает документацию, содержащую следующие справочники:

- Справочник областей именования (svcTypeId, см. п. 3.5);
- Справочник кодов субсчетов (svcSubNum, см. п.3.5);
- Справочник допустимых назначений платежа (payPurpose, см. п.3.8);
- Справочник дополнительных кодов валют (см. п.3.6);
- Максимальное время на отмену платежей (см. п.3.3);
- Справочник счетов учета платежей и счет по умолчанию (см. п.3.13).

3 Виды взаимодействия и управление платежами

3.1 Виды взаимодействия

Рассматриваются следующие виды взаимодействия ЕСПП и Агента:

1. Запрос информации в ЕСПП со стороны Агента; логически данные операции не меняют состояние системы на стороне ЕСПП: их исполнение не модифицирует прикладные объекты, управляемые ЕСПП; такие запросы не имеют индивидуальных идентификаторов и исполняются on-line:
 - Чтение статуса (getPaymentStatus);
 - Пакетное чтение статуса (getPaymentsStatus) – данная функция используется в том числе для проведения сверок;
 - Запрос на проверку параметров зачисления (checkPaymentParams);
 - Запрос на получение информации о ЛС/Абоненте (queryPayeeInfo);
 - Запрос на поиск ЛС по адресу (findPayeeInfoByAddr).
2. Запрос на управление платежом, подаваемый Агентом в ЕСПП; все такие запросы порождают транзакции, идентифицируемые по номеру платежа, назначаемому Агентом, могут исполняться off-line и над ними определяется набор возможных операций: запрос статуса, отмена, повторный запрос.
 - Запрос на зачисление средств (createPayment);
 - Запрос на отмену зачисления (abandonPayment).

На все запросы от Агента ЕСПП должна отвечать за время, не превышающее 30 секунд. Если время, необходимое на выполнение запроса, превышает 30 секунд, то:

- в случае запросов информации, сервер ЕСПП может прекратить обработку запроса, вернув ошибку ERROR_BUSY; если потребность в запрошенных данных сохранится, Агент должен повторить запрос через некоторое время.
- в случае команды на управление платежом начинается «отложенная обработка»: сервер ЕСПП возвращает в ответ на запрос ERROR_OK, текущий статус платежа и продолжает обработку транзакции. Агент может повторить запрос, запросить его текущий статус, подать команду на отмену, подробнее см. п.3.3.

Если ЕСПП не отвечает в течение 30 секунд:

- в случае запросов информации, если потребность в запрошенных данных сохранится, Агент должен повторить запрос через некоторое время.
- в случае команды на управление платежом, Агент может повторить запрос, запросить его текущий статус, подать команду на отмену.

3.2 Жизненный цикл платежа

Жизненный цикл платежа изображен на рисунке 2.

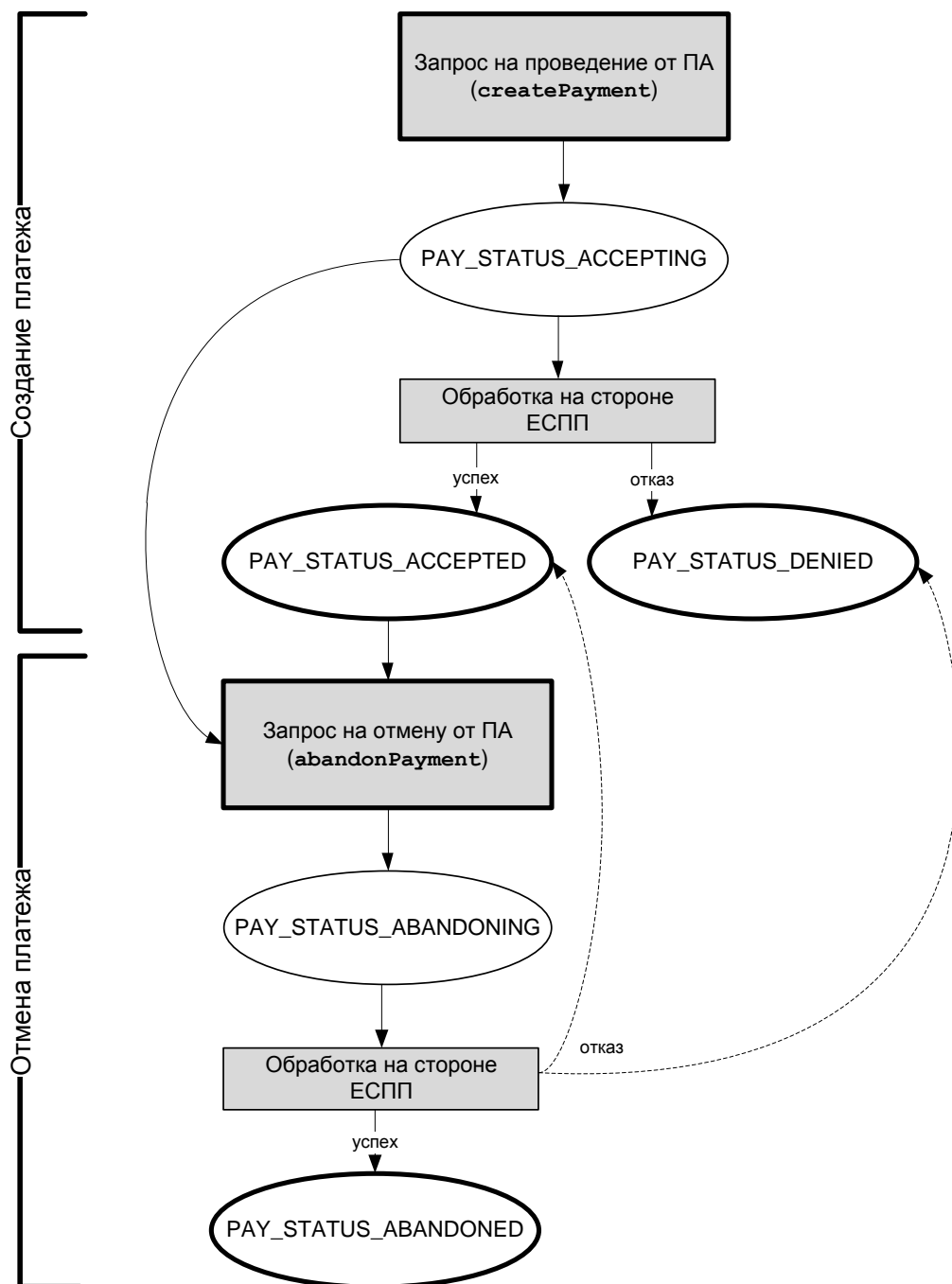


Рис. 2

Обработка платежа начинается в момент получения ЕСПП запроса на создание (зачисления – createPayment) со стороны Агента и заканчивается моментом, когда платеж примет один из финальных статусов: «выполнен», «отменен» или «отклонен» (PAY_STATUS_ACCEPTED, PAY_STATUS_ABANDONED и PAY_STATUS_DENIED, см. рис. 2).

По платежу в статусе «выполнен» и «обрабатывается» Агент может отправить запрос на отмену в ЕСПП и ЕСПП должна будет отменить его. Платеж в этом случае переходит в статус «отменен», подробнее см. п.3.3.

Если ЕСПП успевает исполнить запрос за 30 секунд, то ЕСПП возвращает результат исполнения сразу же в рамках одной on-line транзакции. Если обработка продолжительная, то ЕСПП не позднее чем через 30 секунд от получения запроса возвращает результат «OK» (ERROR_OK) и продолжает обработку off-line. Для получения результата обработки команды Агент периодически запрашивает статус платежа. Схематически это можно изобразить так (см. рис. 3):

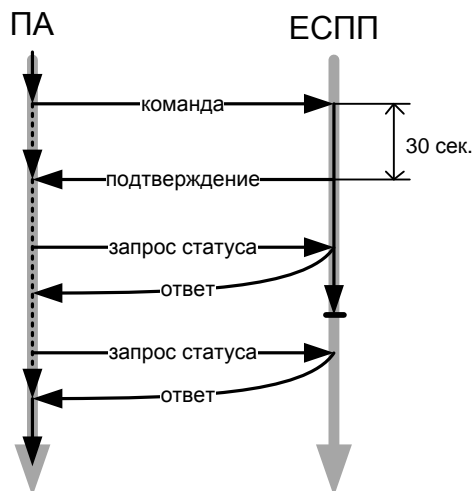


Рис. 3

Технически возможна ситуация, когда 30 секунд еще не истекли, но Агент запросит статус платежа в параллельном соединении, не дожидаясь завершения первого запроса (это возможно при различных нарушениях сетевого взаимодействия). ЕСПП корректно обрабатывает такие ситуации и возвращает текущий статус платежа – «обрабатывается» или другой, который он принял на момент выполнения запроса статуса.

Если ЕСПП не ответила в течение 30 секунд, Агент в зависимости от ситуации может повторить запрос (при этом Агент повторяет запрос на создание платежа с тем же идентификатором платежа), запросить его текущий статус, подать команду на отмену платежа.

Если в момент, когда платеж находится в состоянии обработки, Агент повторно передает запрос на создание платежа с тем же идентификатором платежа (такое возможно, например, если по каким-то причинам Агент не получил ответа на первый запрос), ЕСПП трактует его как повторный запрос и возвращает в ответе текущий статус платежа, который он имеет на данный момент. Схематично это можно изобразить на рисунке 4.

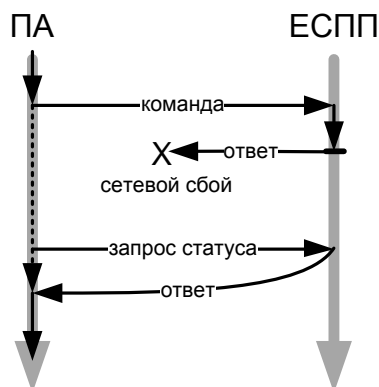


Рис. 4

Повторные запросы определяются на стороне ЕСПП только по значению идентификатора платежа (поле srcPayId), другие поля платежа игнорируются.

3.3 Отмена платежей

Агент может в любой момент подать команду на отмену платежа. Если платеж при этом находится в статусах «обрабатывается» или «выполнен», то ЕСПП инициирует предусмотренную для данного типа платежа процедуру отмены. Для платежа в состоянии «отклонен» ЕСПП возвращает ее текущий статус – «отклонен»¹. Если на стороне ЕСПП невозможно отменить платеж, ЕСПП возвращает отказ отмены с причинами.

Если процедура отмены завершилась успешно – отменой платежа, то платеж принимает статус «отменен». Если отмена не удалась, то платеж принимает предыдущий статус – «выполнен» или «отклонен» (переход обратно в «обрабатывается» запрещен: ЕСПП безусловно прекращает обработку еще незавершенной транзакции). Допускается только временная невозможность отмены по техническим причинам либо отказ от отмены в связи с истечением срока отмены.

На стороне Оператора могут быть настроены ограничения на срок отмены платежа от даты его совершения. Например, может быть установлено ограничение в 60 дней: Агент может отменять платежи в течение последующих 60-и дней, отмена более старых платежей завершается отказом (ошибка ERROR_ABANDON_DENIED, см. п.4.8). Отмена таких платежей должна проводиться обращением Агента в соответствующие службы Оператора.

3.4 Идентификация платежей

Для идентификации платежей в запросах со стороны Агента используется пара полей: srcPayId (идентификатор платежа на стороне Агента) и agentAccount (номер счета для учета платежей).

Идентификаторы платежных транзакций (srcPayId) назначаются на стороне Агента и уникальны в пределах платежной системы Агента, «ротация» идентификаторов не предусматривается. Идентификаторы не несут смысловой нагрузки и генерируются псевдослучайно. В идентификаторах допускается использование любых символов с кодами больше 32 (знак пробела) и меньшими 128. Длина идентификатора не может превышать 64 символа.

Номер счета для учета платежей (agentAccount) - номер счета, на котором должен быть учтен данный платеж, см. п.3.13. Данное поле не является обязательным. В случае, если данное поле в запросе не указывается, или равно нулю, используется номер счета по умолчанию.

Для упрощения сверок и претензионной работы ЕСПП сохраняет идентификатор srcPayId, но в свою очередь присваивает каждому платежу, поступившему от Агентов, свой идентификатор esppPayId, который уникален в пределах ЕСПП, «ротация» идентификаторов не предусматривается. При этом ЕСПП возвращает каждому Агенту идентификатор esppPayId, присвоенный платежу. Поле esppPayId так же уникально, в нем допускается использование любых символов с кодами больше 32 (знак пробела) и меньшими 128. Длина идентификатора не может превышать 64 символа.

3.5 Идентификация лицевого счета

Лицевой счет, для которого выполняется платеж, идентифицируется парой полей: svcTypeId и svcNum. При необходимости номер субсчета передается в поле svcSubNum (см. ниже).

Поле svcTypeId задает пространство имен, в котором svcNum указывает на конкретный лицевой счет. Если svcTypeId в запросе не указан или содержит значение «0» (ноль), то svcNum содержит федеральный телефонный номер², связанный с оплачиваемым ЛС, и состоит из 10-и цифр от «0» до «9» (разделители и другие знаки удалены).

Для идентификации непосредственно самих ЛС с каждой АСР утверждается наименование одного или более пространств имен и синтаксис номеров ЛС для каждого из пространств.

В случае, если на одном счете располагается несколько услуг, и при необходимости указать конкретную оплачиваемую услугу, оплачиваемая услуга рассматривается как субсчет в пределах лицевого счета и ее код передается в поле svcSubNum. Услуги МГМН связи так же могут указываться с помощью субсчета.

Если один платеж делается в счет нескольких субсчетов, перечень субсчетов с соответствующими частными суммами передается в поле payDetails (см. п.3.12), а поле svcSubNum не используется.

3.6 Валюты и денежные единицы

Все денежные суммы передаются только в виде целых чисел в минимальных единицах соответствующей валюты, дробные числа не используются.

¹ Допускается также вернуть статус «отменен», т.к. оба статуса подразумевают, что платеж не исполнен на стороне ЕСПП и с точки зрения Агента нет принципиальной разницы почему – из-за его ошибочности или успешного исполнения команды на отмену. Так же эквивалентность отсутствия платежа и платежа в статусе «отменен» учитывается при сверках, см. п.3.11

² Или номер ЛС в случае МРФ «Урал», где номера ЛС не пересекаются с абонентскими номерами и составляют с ними единое пространство имен.

Все поля с денежным значением сопровождаются полями с указанием кода валюты, даже если текущая версия протокола предполагает работу только с одной predetermined валюты. Валюты кодируются трехбуквенными аббревиатурами. Коды основных валют соответствуют кодам ISO-4217, дополнительно к ним могут быть использованы коды валют, специфичные для АСР. Перечень допустимых для использования кодов валют передается Агенту в составе документации к подключению.

В данной версии все платежи зачисляются в рублях (код валюты RUB, так же допускается написание RUR). Соответствующее поле кода валюты (payCurrId) является обязательным и всегда содержит или RUB или один из допустимых кодов.

3.7 Время платежа

Агент передает в ЕСПП в поле payTime дату фактического приема средств в пункте приема. Эта дата имеет юридическое значение как дата возникновения обязательств Оператора перед Абонентом и не связана с взаиморасчетами других сторон, участвующих в приеме платежа.

3.8 Назначение платежа

Агент передает в ЕСПП в поле payPurpose информацию, которая может быть использована для правильного учета платежа на счетах бухгалтерского учета. Значения поля payPurpose могут быть индивидуальными для каждой АСР и передаются Агенту вместе с другими справочниками.

3.9 Отложенная обработка

Статус платежа «отложенная обработка» (PAY_STATUS_ACCEPTING) предполагает ситуацию, когда сервер ЕСПП определил АСР Абонента (получателя платежа) и возможность выполнения операции, однако по каким либо причинам выполнить операцию своевременно (в течение отведенного на обслуживание запроса времени, 30 секунд) невозможно. Такая ситуация может возникнуть из-за проблем связи или из-за слишком долгого отсутствия ответа со стороны АСР. Возможна ситуация когда АСР отвечает своевременно без задержек, но сама возвращает результат «отложенная обработка». Для Агента обе эти ситуации выглядят одинаково. В таких случаях Агент должен эпизодически выполнять «уточнения» статуса в ЕСПП, вызывая getPaymentStatus. Агент не должен вызывать эту функцию для одного и того же платежа чаще раза в 60 секунд. Следует заметить, что за счет контроля реквизитов Абонента и платежа сервером ЕСПП, платеж в «отложенной обработке» с очень значительной вероятностью в итоге будут успешно принят. Тем не менее, не смотря на то, что за счет контроля сервером ЕСПП, «отложенные» платежи и можно трактовать как «вероятно успешные», окончательный ответ будет получен только когда платеж будет обработан АСР Абонента.

Фактически, статус «отложенная обработка» означает техническую задержку в обработке платежа на стороне ЕСПП-АСР.

3.10 Сверки

Процедура сверки выполняется согласно регламентам, выходящим за рамки протокола ПА-ЕСПП. Для выполнения сверок протокол предоставляет возможность получения от ЕСПП списка платежей за период (см. пакетное чтение статуса, п.5.5), который затем используется самим Агентом для сверки. Агент может запрашивать реестр платежей от ЕСПП как в автоматическом режиме с заданной периодичностью, так и по требованию пользователя.

При сверке сравнение статусов должно проводиться следующей таблице:

Таблица 1. Соответствие статусов платежа на сторонах Агента и ЕСПП

Агент	Отсутствует	ACCEPTING	ACCEPTED	DENIED	ABANDONING	ABANDONED
ЕСПП						
Отсутствует	ok	ok ¹	BAD	ok ¹	BAD	ok ¹
ACCEPTING	BAD	ok	BAD	BAD	ok ⁴	ok ⁴
ACCEPTED	BAD	BAD	ok	BAD	BAD	BAD
DENIED	ok ²	BAD	BAD	ok	ok ³	ok ³
ABANDONING	ok ²	ok ⁴	ok ⁴	ok	ok	ok
ABANDONED	ok ²	BAD	BAD	ok ³	ok	ok

¹) ЕСПП может не включать в реестр отложенные, не принятые и отмененные платежи.

²) Допускается, что на стороне ЕСПП есть платежи, отмеченные ей как неуспешные, а на стороне Агента их нет.

³) Расхождения в статусах PAY_DENIED и PAY_ABANDONED на стороне Агента и ЕСПП допустимы.

⁴) Допускается, что на стороне ЕСПП ожидается завершение проведения платежа, который на стороне Агента находится в состоянии отмены, либо на стороне Агента начата операция отмены, которая еще не начата на стороне ЕСПП. Если в результате сверки статусов получен результат «BAD», службы эксплуатации должны предпринять установленные регламентами действия, направленные на устранение расхождений (повторное проведение, отмена, ручная коррекция и т.п.).

3.11 Повтор платежа

ЕСПП передает Агенту в поле dupFlag информацию о том, является запрос повторным или нет. Данное поле не является обязательным. В настоящий момент определен следующий код:

Код	Значение
1	Запрос является повторным
2	Повторный запрос, но исходный запрос инициирован другим источником. Значение возвращается ЕСПП в случае, когда ПА подает команду на отмену платежа, уже отмененного самим Оператором (например, через пользовательский интерфейс ЕСПП или АСР ³).

3.12 Детализация платежа

Если платеж относится не к ЛС в целом, а к отдельной услуге/субсчету в его составе, то Агент передает информацию о субсчете либо в поле svcSubNum (см. п.3.5) либо в виде детализации платежа в поле payDetails (см п.3.12).

Поле payDetails содержит массив записей, содержащих следующие элементы:

- Код услуги/субсчета (используются те же коды, что и для svcSubNum)
- Значение частной суммы (в минимальных единицах валюты, в копейках – для рублей)
- Назначение зачисления (поле необязательное, используются те же коды, что и для поля payPurpose, см. п.3.8)

В случае операции зачисления средств, детализация платежа указывает субсчета и частные суммы, зачисляемые на каждый из них. Поле svcSubNum в этом случае не используется. Таким образом, ЕСПП позволяет зачислить одним платежом средства на несколько субсчетов.

3.13 Счета учета платежей

В случае, если Договор Агента допускает прием платежей по различным условиям, ему необходимо относить платежи к различным статьям учета. Перечень статей и связанных с ними условий определяется на этапе заключения Договора и конфигурирования взаимодействия. Логически, статьи могут быть представлены как «виртуальные Агенты», разделяющие общее техническое подключение к ЕСПП, но имеющие индивидуальные условия работы.

При передаче платежа номер статьи, на которой должен быть отображен платеж, передается в поле agentAccount. Если поле не заполнено, то выбирается «счет по-умолчанию», настроенный в параметрах технического подключения Агента.

³ Если платеж отменен вручную через кабинет операциониста ПА, предоставляемый ЕСПП, то такая отмена считается выполненной самим ПА и поле DUP_FLAG будет содержать значение 1. Значение 2 возвращается только если платеж отменен на стороне Оператора.

4 Формат запросов и ответов

4.1 Использование HTTP

ЕСПП и Агент взаимодействуют по протоколу HTTPS (HTTP/1.1), обмениваясь данными в формате HTTP POST url encoded, либо в JSON формате.

Данные передаются в теле запроса. Стороны обязаны корректно указывать тип передаваемых данных. Если используется HTTP метод POST, Агент и ЕСПП указывают в заголовках HTTP следующие Content-Type и Accept:

```
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Accept: application/x-www-form-urlencoded
Accept-Charset: UTF-8
```

Если используется формат JSON, ЕСПП и Агент указывают в заголовках HTTP следующие Content-Type и Accept:

```
Content-Type: application/json; charset=UTF-8
Accept: application/json
Accept-Charset: UTF-8
```

Допускается использование кодировок Windows-1251 и UTF-8, предпочтительной является UTF-8.

В случае если клиент указывает непредусмотренный Content-Type, сервер ЕСПП вернет HTTP ошибку 400 (Bad Request) или 415 (Unsupported Media Type). Если клиент указывает несоответствующий настройкам взаимодействия Accept, сервер вернет HTTP ошибку 406 (Not Acceptable).

В случае если тело запроса содержит несоответствующие Content-Type данные, то сервер обязан вернуть HTTP ошибку 400 (Bad Request).

HTTP Status Line ответа всегда должен содержать текстовый комментарий, соответствующий ошибке.

Если тело запроса является корректным form-urlencoded или JSON объектом, то сервер вернет ответ HTTP 200 Ok.

Прикладной результат исполнения запроса (код возврата) при этом возвращается в теле ответа в поле reqStatus (см. п. 4.7), который присутствует во всех ответах. Код возврата сопровождается текстовым комментарием reqNote с расшифровкой ошибки, этот комментарий предназначен только для разработчиков и специалистов служб технической поддержки ЕСПП и Агента, и никогда не должен отображаться в интерфейсе конечного пользователя.

Все запросы Агента к ЕСПП передаются на один и тот же URL, конфигурируемый в настройках соединения на стороне Агента на этапе интеграции систем. При этом вид запроса (имя вызываемой функции) передается в параметре запроса reqType (каждый запрос содержит этот обязательный параметр).

4.2 Формирование ответа сервера

При использовании метода HTTP POST значение всех полей подвергаются URL-кодированию, описанному в документе RFC2396.

Название поля field и его значение value объединяются в строку 'field=value', после этого полученные выражения для всех полей объединяются в строку символом '&'. Т.е. «field₁=value₁&field₂=value₂...&field_n=value_n».

Полученная строка выдается клиенту как ответ сервера.

4.3 URL-кодирование

При использовании HTTP-POST формата должны выполняться следующие условия:

- В заголовке HTTP Content-Type должен быть application/x-www-form-urlencoded.
- Все текстовые значения передаются в кодировке UTF-8 или Win-1251 (поля HTTP-заголовка должны иметь соответствующее значение).
- В случае, если значение какого-либо атрибута сообщения содержит символы, не принадлежащие множеству символов '0'..'9', 'A'..'Z', 'a'..'z', '-', '_', '.', '!', '~', '*', '"', '(', ')', ' ' (коды ASCII 0x30..0x39, 0x41..0x5A, 0x61 .. 0x7A, 0x2D, 0x5F, 0x2E, 0x21, 0x7E, 0x2A, 0x27, 0x28, 0x29), символ заменяется на последовательность из символа '%' (код ASCII 0x25) и двух шестнадцатеричных цифр (символов '0'..'9', 'A' .. 'F') с кодом заменяемого символа (более детально данное преобразование описано в документе RFC 2396).
- Имя и значение каждого атрибута сообщения объединяются в строку с символом-разделителем '=' (код ASCII 0x3D).
- Имя сообщения и полученные строки атрибутов объединяются в строку с символом-разделителем '&' (код ASCII 0x26), при этом строки атрибутов следуют в порядке, указанном в описании атрибутов сообщения.
- Разбор сообщения сервера происходит в обратном порядке.

4.4 Передача массивов

Некоторые запросы, а также ответы содержат в себе массивы. Передача массивов в формате HTTP-POST происходит следующим образом: элементы массива отделяются между собой символом '|', строки отделяются переводом строки (парой

символов <CR><LF>, коды 0x0D ASCII 13, 0x0A ASCII 10, или же одиночного символа <LF>, шестнадцатеричный код 0x0A, ASCII 10). Если в тексте строкового значения присутствует символ-разделитель "|", он кодируется в нотации url encoded как сказано в п.4.3. Отформатированный массив подвергается url-кодированию.

При передаче массивов в формате HTTP POST используются следующие поля:

- Поле payDetails для каждого субсчета svcSubNum|payAmount|payPurpose
- Поле payeeRemainDetails для каждого субсчета svcSubNum|payAmount
- Поле searchResults для каждого ЛС svcTypeId|svcNum|svcSubNum

Передача массивов в JSON формате осуществляется стандартным способом.

При передаче массивов в формате JSON используются следующие поля:

- Поле payDetails для каждого субсчета {"svcSubNum": "значение", "payAmount": "значение", "payPurpose": "значение"}
- Поле payeeRemainDetails для каждого субсчета {"svcSubNum": "значение", "payAmount": "значение"}
- Поле searchResults для каждого ЛС {"svcTypeId": "значение", "svcNum": "значение", "svcSubNum": "значение"}

4.5 Передача табличных данных

Ответы на некоторые запросы, например запрос пакетного чтения статуса (getPaymentsStatus, см. п.5.5) содержат в ответе табличную часть. В этом случае при использовании HTTP POST запрос состоит из двух частей: заголовочной (шапки) и табличной. Заголовочная часть содержит общую информацию (тип запроса) и кодируется по правилам HTTP-POST (url form encoded), аналогично HTTP-POST синтаксису основных запросов. Вся заголовочная часть представляет собой одну строку, отделяемую от следующей далее табличной части переводом строки (парой символов <CR><LF>, коды 0x0D ASCII 13, 0x0A ASCII 10, или же одиночного символа <LF>, шестнадцатеричный код 0x0A, ASCII 10). Использование символов перевода строки в значениях полей недопустимо.

Табличная часть кодируется в формате близком к CSV (Comma Separated Values):

1. Записи передаются в текстовом формате, каждая строка соответствует одной записи.
2. В качестве переводов строк, разделяющих записи, допускается использование как одиночного символа <LF> (шестнадцатеричный код 0x0A, ASCII 10), так и пары <CR><LF> (коды 0x0D ASCII 13, 0x0A ASCII 10).
3. Поля в строке отделяются разделителем | (шестнадцатеричный код 0x7C, ASCII 124), порядок и набор полей строго регламентирован для каждого запроса. Если в тексте строкового значения присутствует символ-разделитель "|", он кодируется в нотации url encoded как сказано в п.4.3.
4. Разделители полей | должны примыкать к значениям вплотную без лишних пробелов. Так, строка 123|Строковое значение|123.45 правильна, а 123 |Строковое значение | 123.45 - неправильна. Если для поля допустимо значение null, то оно кодируется как «отсутствие значения», т.е. идущие подряд разделители '|' без символов между ними
5. Разделители '|' располагаются только между полями, после последнего поля записи разделитель не указывается (после значения сразу же располагается перевод строки)
6. Рекомендуется последнюю запись заканчивать переводом строки, хотя допускается после последней записи перевод строки не указывать
7. Все значения полей кодируются в нотации, указанной в п.4.3.
8. В значениях строковых полей допускаются символы с кодом от 32 (пробел) до 255 кроме '%' (знака процента) и '|' (вертикальной черты, которая используется для разделения полей). Символы с кодами до 32, знак процента и вертикальная черта кодируются в нотации url encoded: %hh, где знак процента указывает на начало кода, а hh - шестнадцатеричный код символа. Например, последовательность <CR><LF>, означающую перевод строки, нужно задавать как %0D%0A. Регистр символов шестнадцатеричного кода игнорируется, так %0D и %0d эквивалентны. Знак процента может быть закодирован как кодом (%25) так и последовательностью из двух знаков процента (%%).

В случае использования JSON-формата, табличная часть ответа передается в виде массива:

```
{
  "reqStatus": "значение",
  "payments": [{
    "транзакция_1_поле_1": "значение",
    ...
    "транзакция_1_поле_N": "значение"
  },
  ...
  {
    "транзакция_N_поле_1": "значение",
    ...
    "транзакция_N_поле_N": "значение"
  }
}]
}
```

4.6 Типы данных

Запрос и ответ на него являются объектами, содержащими как обязательные поля (reqType в запросе, reqStatus в ответе), так и поля, специфичные для конкретных видов запросов.

Поля объектов могут иметь типы данных, перечисленные в следующей таблице:

Таблица 2. Обозначения типов данных

Тип	Пояснение
N[n]	Целое число разрядностью не более n
S[n]	Текстовое значение длиной не более n символов
MONEY	Сумма в минимальных единицах валюты (в копейках для рублей)
DATETIME	Дата и время (timestamp) в формате ГГГГ-ММ-ДД Тчч:мм:сс[.мск]±чч:мм ⁴ например: 2005-07-01T13:23:15+6:00 или 2005-07-01T13:23:15.781+6:00 <i>ВНИМАНИЕ: Указание часовой зоны является обязательным, при этом передающая сторона ответственна за правильное значение текущего смещения своей часовой зоны, сервер ЕСПП не делает предположений относительно нее и текущего летнего/зимнего времени на стороне Агента</i>
ARRAY	Массив, формат которого описывается в примечании к соответствующему полю.

Различные поля в запросах могут быть обязательными (not null) или необязательными (nullable). При описании типов данных ниже в п.п.4.7, 5 обязательные поля обозначаются жирным шрифтом.

4.7 Поля ответа на запросы

Ответы со стороны сервера ЕСПП на запросы Агента содержат следующие общие для всех запросов поля:

Поле	Тип	Назначение	Комментарий
reqStatus	N	Статус операции	Статус выполнения операции
payStatus	N	Статус платежа	Статус платежа
reqType	S[64]		Последняя операция (выполненная или обрабатываемая)
reqTime	DATETIME		Время, когда платеж получил текущий статус (начата операция, если статус промежуточный или окончена, если финальный, см. п.3.2)
reqNote	S	Сообщение	Текстовое сообщение (комментарий) об ошибке обработки запроса
errUsrMsg	S	Сообщение плательщику	Сообщение, предназначенное для вывода непосредственно плательщику в случае невозможности выполнения операции

В ответах на запросы возвращаются два значения, показывающие результаты исполнения запроса (reqStatus) и состояния платежа (payStatus) которое тот принял в результате выполнения запроса. Если команда завершилась неуспешно (reqStatus не равен 0), то платеж сохраняет свой предыдущий статус, который указывается в ответе.

Если результат исполнения запроса reqStatus не равен нулю, то в ответ включаются только поля reqStatus и reqNote. Все другие поля, отмеченные в описании соответствующего запроса как обязательные, отсутствуют. Но если такой запрос привел к порождению транзакции на стороне ЕСПП, ЕСПП включает в ответ полный набор полей, предусмотренный к возврату на данный запрос. При устранении причин ошибки такой запрос может быть повторен с тем же srcPayId. Если же в ответе присутствует поле payStatus, то это всегда означает, что транзакция создавалась, и повторное обращение с тем же srcPayId всегда будет трактоваться как повторный запрос: выполнен ЕСПП не будет, а будет возвращен ранее полученный результат с указанием, что это дублирующая операция.

Поле ответа reqNote может содержать текстовое сообщение об ошибке, предназначенное для работы службы сопровождения. Это сообщение не должно отображаться плательщику, т.к. неинформативно для него.

⁴

Формат даты-времени соответствует подмножеству формата ISO 8601, принятому для использования в SOAP (тип xsd:dateTime)

4.8 Статусы выполнения запросов

Допустимые значения поля reqStatus приведены в таблице:

	reqStatus	Описание
0	ERROR_SUCCESS (ERROR_OK)	Запрос выполнен успешно
1	ERROR_PAY_NOT_FOUND	Платеж с таким идентификатором отсутствует
2	ERROR_BAD_AMOUNT	Платеж не может быть выполнен на такое значение payAmount Платеж может быть отклонен с этой ошибкой если сумма платежа находится вне допустимого диапазона и других нарушениях контроля суммы отдельного платежа. Для случаев превышения суммы операций по счету за период предусмотрен отдельный код ошибки: ERROR_PAY_LIMIT_EXCEEDED
-1	ERROR_BUSY	Сервер временно недоступен. По внутренним техническим причинам сервер не обрабатывает запросы
-2	ERROR_ACCESS_DENIED	Доступ Агента к сервису ЕСПП запрещен. Необходимо проанализировать комментарий поля reqNote: возможно запрос делается с недопустимого IP, с некорректным сертификатом, ЕСПП не включен в боевой режим и т.п.
-3	ERROR_BAD_REQ	Неизвестный / не поддерживаемый запрос
-4	ERROR_BAD_FORMAT	Неверный формат запроса. Указаны не все обязательные поля, либо значение полей имеют неправильный формат. Расшифровка должна передаваться в reqNote — имя поля, суть ошибки и т.д.
-5	ERROR_BAD_CURR	Недопустимая валюта; (должна быть RUB или другая из списка допустимых)
-12	ERROR_PAYEE_NOT_FOUND	Неизвестный получатель средств. Получателя средств с таким svcNum/svcSubNum не существует, либо по некоторым причинам информация о нем недоступна
-15	ERROR_REQ_DENIED	Запрос отклонен. Причина отклонения указывается в поле reqNote. Сообщение, выводимое непосредственно плательщику, передается в поле errUsrMsg
-17	ERROR_BAD_SVC_TYPE	Недопустимое пространство имен, см. п.3.5
-21	ERROR_PAY_LIMIT_EXCEEDED	Превышена сумма допустимых операций по счету за период. Ошибка предназначена для таких видов контроля операций по счету, как контроль максимального числа и максимальной суммы операций за период.
-22	ERROR_PAYEE_CLOSED	Данный svcNum/svcSubNum, закрыт или заблокирован
-23	ERROR_ABANDON_DENIED	Отмена невозможна в связи с истечением срока давности. Для отмены необходимо обратиться в соответствующие службы Оператора.

4.9 Статусы платежей

Статус платежа показывает состояние платежа, которое тот получил после исполнения команды.

Код	payStatus	Описание
102	PAY_STATUS_ACCEPTING	Платеж обрабатывается. Начальный статус платежа.
2	PAY_STATUS_ACCEPTED	Платеж выполнен. Финальный статус.
103	PAY_STATUS_ABANDONING	Платеж отменяется.
3	PAY_STATUS_ABANDONED	Платеж отменен. Финальный статус.
4	PAY_STATUS_DENIED	Платеж отклонен. Финальный статус.

5 Описание запросов и ответов

5.1 Запрос на проверку параметров зачисления

Проверка параметров зачисления является опциональной операцией, выполняемой перед операцией зачисления средств (Агент может выполнять операцию зачисления и без предварительной проверки). Также возможно, что после успешной проверки параметров Агент по каким-либо причинам откажется от проведения платежа.

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	checkPaymentParams
svcTypeId	S[20]	Пространство имен	Идентификатор ЛС, см. п.3.5
svcNum	S[20]	Идентификатор ЛС в пространстве svcTypeId	
svcSubNum	S[20]	Идентификатор субсчета	Указывает при необходимости на номер субсчета / услуги к которой относится платеж
payCurrId	S[3]	Код валюты платежа	Код валюты, см. п.3.6
payAmount	MONEY	Сумма платежа в минимальных единицах	Фактическая сумма платежа
payPurpose	N	Назначение платежа	Назначение платежа, см. п.3.8
payComment	S[512]	Основание операции	Основание операции, Текстовый комментарий производимой операции, предназначенный для отображения в интерфейсе пользователя
payDetails	ARRAY	Детализация платежа	Описание детализации платежа по услугам/субсчетам и частным суммам, см. п.3.12
agentAccount	N	Счет учета платежа	Номер счета, на котором должен быть учтен данный платеж, см. п.3.13 Пустое значение или 0 – счет по умолчанию

Поля ответа на запрос

Поле	Тип	Назначение	Комментарий
reqStatus	N	Статус запроса	Статус выполнения операции
reqTime	DATETIME	Время выполнения запроса на проверку параметров платежа	Время фактического выполнения операции проверки
reqNote	S	Сообщение	Текстовое сообщение (комментарий) об ошибке обработки запроса
errUsrMsg	S	Сообщение плательщику	Сообщение, предназначенное для вывода непосредственно плательщику в случае невозможности выполнения операции

Пример запроса в формате HTTP POST:

reqType=checkPaymentParams&svcTypeId=0&svcNum=9123456780&payCurrId=RUB
&payAmount=10000&payPurpose=0&payDetails=3%7C7000%7C0%250D%250A5%7C3000%7C0

Пример ответа в формате HTTP POST:

reqStatus=0&reqTime=2011-10-25T13%3A23%3A15%2B6%3A00

Пример запроса в формате JSON:

```
{
  "reqType": "checkPaymentParams",
  "svcTypeId": "0",
  "svcNum": "9123456780",
  "payCurrId": "RUB",
  "payAmount": 10000,
```

```

"payPurpose": 0,
"payDetails": [{
  "svcSubNum": "3",
  "payAmount": 7000,
  "payPurpose": 0
},{
  "svcSubNum": "5",
  "payAmount": 3000,
  "payPurpose": 0
}]
}

```

Пример ответа в формате JSON:

```

{
  "reqStatus": 0,
  "reqTime": "2011-10-25T13:23:15+6:00"
}

```

5.2 Запрос на зачисление средств

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	createPayment
svcTypeId	S[20]	Пространство имен	Идентификатор ЛС, см. п.3.5
svcNum	S[20]	Идентификатор ЛС в пространстве svcTypeId	
svcSubNum	S[20]	Идентификатор субсчета	Указывает при необходимости на номер субсчета / услуги к которой относится платеж
srcPayId	S[64]	Номер платежной транзакции	Номер платежа в платежной системе Агента
payTime	DATETIME	Время создания платежа	Время фактического приема средств в источнике
payCurrId	S[3]	Код валюты платежа	Код валюты, см. п.3.6
payAmount	MONEY	Сумма принятого платежа в минимальных единицах	Фактическая сумма платежа
payPurpose	N	Назначение платежа	Назначение платежа, см. п.3.8
payComment	S[512]	Основание платежа	Основание платежа, Текстовый комментарий производимой операции, предназначенный для отображения в интерфейсе пользователя
payDetails	ARRAY	Детализация платежа	Описание детализации платежа по услугам/субсчетам и частным суммам, см. п.3.12
agentAccount	N	Счет учета платежа	Номер счета, на котором должен быть учтен данный платеж, см. п.3.13 Пустое значение или 0 – счет по-умолчанию
reqTime	DATETIME	Время операции у Агента	Время начала операции на стороне Агента

Поля ответа на запрос

Поле	Тип	Назначение	Комментарий
srcPayId	S[64]	Номер платежной транзакции	Номер транзакции у Агента. Повторяется SrcPayId, указанный Агентом в запросе
esppPayId	S	Номер платежа в ЕСПП	Уникальный идентификатор транзакции в ЕСПП
reqTime	DATETIME	Время приема платежа	Время выполнения операции на стороне ЕСПП / получения платежом текущего статуса
reqType	S[64]	Тип последней операции	Тип операции, в результате которой платеж

			принял текущий статус
reqStatus	N	Статус запроса	Статус выполнения операции
reqNote	S	Сообщение	Комментарий или текстовое сообщение об ошибке обработки запроса
reqUserMsg	S	Сообщение	Сообщение, выдаваемое плательщику на платежном терминале или др. интерфейсе оплаты по окончании платежной операции
errUsrMsg	S	Сообщение плательщику	Сообщение, предназначенное для вывода непосредственно плательщику в случае невозможности выполнения операции
dupFlag	N	Флаг повторного платежа	Флаг, сигнализирующий Агенту о том, что запрос отправляется повторно, см. п.3.11
payStatus	N	Статус платежа	Статус платежа в ЕСПП
dstDepCode	S	Код подразделения-получателя платежа	По отдельному согласованию ЕСПП может возвращать на сторону ПА код подразделения Оператора, в которое передан платеж. Справочник кодов подразделений сообщается ПА отдельным приложением к данной Спецификации.

Пример запроса в формате HTTP POST:

```
reqType=createPayment&svcTypeId=0&svcNum=9123456780&srcPayId=1237734555
&payTime=2011-10-25T13%3A23%3A15%2B6%3A00&payCurrId=RUB
&payAmount=10000&payPurpose=0&payDetails=3%7C8000%7C0%250D%250A5%7C2000%7C0
```

Пример ответа в формате HTTP POST:

```
reqStatus=0&esppPayId=P-125635613&srcPayId=1237734555
&reqTime=2011-10-25T13%3A23%3A15%2B6%3A00&payStatus=2&reqType=createPayment
```

Пример запроса в формате JSON:

```
{
  "reqType": "createPayment",
  "svcTypeId": "0",
  "svcNum": "9123456780",
  "srcPayId": "1237734555",
  "payTime": "2011-10-25T13:23:15+6:00",
  "payCurrId": "RUB",
  "payAmount": 10000,
  "payPurpose": 0,
  "payDetails": [{
    "svcSubNum": "3",
    "payAmount": 7000,
    "payPurpose": 0
  }, {
    "svcSubNum": "5",
    "payAmount": 3000,
    "payPurpose": 0
  }]
}
```

Пример ответа в формате JSON:

```
{
  "reqStatus": 0,
  "esppPayId": "P-125635613",
  "srcPayId": "1237734555",
  "payStatus": 2,
  "reqType": "createPayment",
  "reqTime": "2011-10-25T13:23:15+6:00"
}
```

5.3 Запрос на отмену зачисления

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	abandonPayment
srcPayId	S[64]	Номер платежа у Агента	Номер транзакции у Агента
agentAccount	N	Счет учета платежа	Номер счета, на котором учтен данный платеж,

			см. п.3.13 Пустое значение или 0 – счет по умолчанию
reqTime	DATETIME	Время операции у Агента	Время начала операции на стороне Агента

Поля ответа на запрос

Поле	Тип	Назначение	Комментарий
srcPayId	S[64]	Номер платежной транзакции	Номер транзакции у Агента. Повторяется srcPayId, указанный Агентом
reqTime	DATETIME	Время выполнения операции	Время фактического выполнения операции
reqType	S[64]	Тип последней операции	Тип операции, в результате которой платеж принял текущий статус
reqStatus	N	Статус операции	Статус выполнения операции
dupFlag	N	Флаг повторного платежа	Флаг, сигнализирующий Агенту о том, что запрос отправляется повторно, см. п.3.11
reqNote	S	Сообщение	Комментарий или текстовое сообщение об ошибке обработки запроса
payStatus	N	Статус платежа	Статус платежа в ЕСПП

Пример запроса в формате HTTP POST:

reqType=abandonPayment&srcPayId=1237734555&payTime=2011-10-25T13%3A23%3A15%2B6%3A00

Пример ответа в формате HTTP POST:

reqStatus=0&srcPayId=1237734555&reqTime=2011-10-25T13%3A23%3A25%2B6%3A00

&payStatus=3&reqType=abandonPayment

Пример запроса в формате JSON:

```
{
  "reqType": "abandonPayment",
  "srcPayId": "1237734555",
  "payTime": "2011-10-25T13:23:15+6:00"
}
```

Пример ответа в формате JSON:

```
{
  "reqStatus": 0,
  "srcPayId": "1237734555",
  "payStatus": 3,
  "reqType": "abandonPayment",
  "reqTime": "2011-10-25T13:23:15+6:00"
}
```

5.4 Чтение статуса

Чтение статуса платежа в основном предназначено для работы с платежами, получившими статус «отложенная обработка» - PAY_STATUS_ACCEPTING или PAY_STATUS_ABANDONING, см. п. 3.9.

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	getPaymentStatus
srcPayId	S[64]	Номер платежа у Агента	Номер транзакции у Агента
agentAccount	N	Счет учета платежа	Номер счета, на котором учтен данный платеж, см. п.3.13 Пустое значение или 0 – счет по умолчанию

Поля ответа на запрос

Поле	Тип	Назначение	Комментарий
reqStatus	N	Текущий статус платежа	Статус выполнения операции
reqNote	S	Сообщение	Комментарий или текстовое сообщение об ошибке обработки запроса
acceptTime	DATETIME	Дата проведения	Дата reqTime, указанная Агентом в команде на

			зачисление
acceptedTime	DATETIME	Дата фактического проведения	Дата фактического проведения (зачисления или списания) платежа на стороне ЕСПП. Не заполняется если платеж еще не проведен (находится в состоянии PAY_STATUS_ACCEPTING)
abandonTime	DATETIME	Дата отмены	Дата reqTime, указанная Агентом в команде на отмену транзакции, если отмена была
abandonedTime	DATETIME	Дата фактической отмены	Дата фактической отмены транзакции на стороне ЕСПП, если отмена была
esppPayId	S	Номер платежа в ЕСПП	Уникальный идентификатор транзакции в ЕСПП
reqType	S[64]	Тип последней операции	Тип операции, в результате которой платеж принял текущий статус
payStatus	N	Статус платежа	Статус платежа в ЕСПП
dstDepCode	S	Код подразделения-получателя платежа	По отдельному согласованию ЕСПП может возвращать на сторону ПА код подразделения Оператора, в которое передан платеж. Справочник кодов подразделений сообщается ПА отдельным приложением к данной Спецификации.
payTime	DATETIME	Время создания платежа	Время платежа в источнике

Пример запроса в формате HTTP POST:

reqType=getPaymentStatus&srcPayId=1237734555

Пример ответа в формате HTTP POST:

reqStatus=0&esppPayId=P-125635613&reqType=createPayment
&acceptTime=2011-10-25T13%3A23%3A20%2B6%3A00&acceptedTime=2011-10-25T13%3A23%3A20%2B6%3A00&payStatus=2&payTime=2011-10-25T13%3A23%3A20%2B6%3A00

Пример запроса в формате JSON:

```
{
  "reqType": "getPaymentStatus",
  "srcPayId": "1237734555"
}
```

Пример ответа в формате JSON:

```
{
  "reqStatus": 0,
  "esppPayId": "P-125635613",
  "acceptTime": "2011-10-25T13:23:15+6:00",
  "acceptedTime": "2011-10-25T13:23:15+6:00",
  "payStatus": 2,
  "reqType": "createPayment",
  "payTime": "2011-10-25T13:23:15+6:00"
}
```

5.5 Пакетное чтение статуса

Агент может запросить в ЕСПП информацию о статусах транзакций за период продолжительностью не более одной недели. Ответ содержит табличные данные с информацией о платежах, в соответствии с выборкой по указанным параметрам. Запрос задает период [startDate, endDate), за который отбираются транзакции, дата зачисления или дата отмены которых попадают в указанный период (это позволяет запросить список транзакций, «изменившихся за период»). При этом, сравнение дат с границами startDate и endDate, проводится последующему неравенству:

$$startDate \leq D_1 < endDate$$

$$startDate \leq D_2 < endDate$$

Где D_1 – дата подачи со стороны Агента команды на проведение (возвращается в поле acceptTime ответа, см. ниже), а D_2 – дата подачи со стороны Агента команды на отмену, если отмена производилась (возвращается в поле abandonTime ответа, см. ниже).

Данная функция используется в том числе для проведения сверок.

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	getPaymentsStatus
statusType	N	Операции, включаемые в	0 – неуспешные,

		отчет	1 – успешные (в том числе отмененные), 2 – в обработке Если не указано значение параметра, возвращаются данные по всем операциям
startDate	DATETIME		Дата начала периода. Если не указана, то берется дата endDate —минус 7 дней
endDate	DATETIME		Дата окончания периода. Если не указана, то берется текущая дата
svcTypeId	S[20]	Пространство имен	Идентификатор ЛС, см. п.3.5
svcNum	S[20]	Идентификатор ЛС в пространстве svcTypeId	
svcSubNum	S[20]	Идентификатор субсчета	Указывает при необходимости на номер субсчета / услуги к которой относится платеж
agentAccount	N	Счет учета платежа	Указывает при необходимости номер счета учета платежей, см. п.3.13 Пустое значение – по всем счетам, 0 – счет по-умолчанию

Поля ответа

Поле	Тип	Назначение	Комментарий
reqStatus	N	Статус	Статус выполнения операции
reqNote	S[512]	Сообщение	Комментарий или текстовое сообщение об ошибке
payments	ARRAY	Массив платежей	Массив платежей, удовлетворяющих заданным условиям

Поля элементов массива payments

Поле	Тип	Назначение	Комментарий
srcPayId	S[64]	Номер платежа у Агента	Номер транзакции у Агента. Повторяется srcPayId, указанный Агентом при создании платежа
esppPayId	S	Номер платежа в ЕСПП	Уникальный идентификатор транзакции в ЕСПП, соответствующий внешней транзакции srcPayId
payType	S[1]	Тип платежа	“Р” – зачисление
reqType	S[64]	Тип последней операции	Тип операции, в результате которой платеж принял текущий статус
payStatus	N	Статус платежа	Статус платежа в ЕСПП
dstDepCode	S	Код подразделения-получателя платежа	По отдельному согласованию ЕСПП может возвращать на сторону ПА код подразделения Оператора, в которое передан платеж. Справочник кодов подразделений сообщается ПА отдельным приложением к данной Спецификации.
payTime	DATETIME	Время создания платежа	Время платежа в источнике
payCurrId	S[3]	Код валюты платежа	Код валюты, см. п.3.6
payAmount	MONEY	Сумма принятого платежа в минимальных единицах	Фактическая сумма платежа
acceptTime	DATETIME	Дата команды проведения	Дата reqTime, указанная Агентом в команде на зачисление
acceptedTime	DATETIME	Дата фактического проведения	Дата фактического проведения (зачисления или списания) платежа на стороне ЕСПП.

			Не заполняется если платеж еще не проведен (находится в состоянии PAY_STATUS_ACCEPTING)
abandonTime	DATETIME	Дата команды отмены	Дата reqTime, указанная Агентом в команде на отмену транзакции, если отмена была
abandonedTime	DATETIME	Дата фактической отмены	Дата фактической отмены транзакции на стороне ЕСПП, если отмена была
payPurpose	N	Назначение платежа	Назначение платежа, см. п.3.8
payComment	S[512]	Основание операции	Основание операции, Текстовый комментарий производимой операции, предназначенный для отображения в интерфейсе пользователя

Пример запроса в формате HTTP POST:

reqType=getPaymentStatus&statusType=1

Пример ответа в формате HTTP POST:

reqStatus=0

```
12366712356|P-13123423|P|createPayment|2|2011-10-25T13%3A23%3A20%2B6%3A00|RUB|20000|2011-10-25T13%3A23%3A21%2B6%3A00|2011-10-25T13%3A23%3A25%2B6%3A00|0|0|1237734500|P-13123400|P|createPayment|2|2011-10-25T13%3A23%3A20%2B6%3A00|RUB|20000|2011-10-25T13%3A23%3A21%2B6%3A00|2011-10-25T13%3A23%3A25%2B6%3A00|0|0|
```

Пример запроса в формате JSON:

```
{
  "reqType": "getPaymentsStatus";
  "statusType": 1
}
```

Пример ответа в формате JSON:

```
{
  "reqStatus": 0,
  "payments": [{
    "srcPayId": "1237734555",
    "esppPayId": "P-13123423",
    "payType": "P",
    "reqType": "createPayment",
    "payStatus": 2,
    "payTime": "2011-10-25T13:20:15+6:00",
    "payCurrId": "RUB",
    "payAmount": 10000,
    "acceptTime": "2011-10-25T13:23:15+6:00",
    "acceptedTime": "2011-10-25T13:23:17+6:00",
    "payPurpose": 0,
  },
  {
    "srcPayId": "1237734500",
    "esppPayId": "P-13123400",
    "payType": "P",
    "reqType": "createPayment",
    "payStatus": 2,
    "payTime": "2011-10-25T13:10:15+6:00",
    "payCurrId": "RUB",
    "payAmount": 20000,
    "acceptTime": "2011-10-25T13:13:15+6:00",
    "acceptedTime": "2011-10-25T13:13:20+6:00",
    "payPurpose": 0,
  }
]
```

5.6 Запрос на получение информации о ЛС/Абоненте

Запрос предназначен для проверки существования ЛС/абонента, а так же для получения информации о нем для показа в интерфейсе платежного терминала. Состав возвращаемой информации задается битовой маской в поле запроса queryFlags. В данной версии доступно только получение остатка на лицевом счете с возможной детализацией по услугам/субсчетам для того, чтобы плательщик мог решить какую сумму необходимо оплатить, получение рекомендуемого платежа и имени, отчества абонента. Пустое значение означает, что производится только проверка существования ЛС без получения информации о нем. Состав возвращаемых данных может расширяться в дальнейшем.

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	queryPayeeInfo
svcTypeId	S[20]	Пространство имен	Идентификатор ЛС, см. п.3.5
svcNum	S[20]	Идентификатор ЛС в пространстве svcTypeId	
svcSubNum	S[20]	Идентификатор субсчета	Указывает при необходимости на номер субсчета / услуги по которой запрашивается информация
queryFlags	N	Состав запрашиваемой информации о Получателе Средств	<ul style="list-style-type: none"> 0 – информация о ЛС/Абоненте не требуется, производится только проверка существования ЛС. Установлен бит 0 (значение 0x0001) – требуется информация об остатке на лицевом счете; значение возвращается в поле payeeRemain ответа на запрос. Установлен бит 1 (значение 0x0002) – требуется детализация остатка по услугам/субсчетам; значение возвращается в массиве в поле payeeRemainDetails ответа на запрос. Установлен бит 2 (значение 0x0004) – требуется информация о рекомендуемом платеже; значение возвращается в поле payeeRecPay ответа на запрос. Установлен бит 3 (значение 0x0008) – требуется информация об инициалах абонента; значение возвращается в поле payeeName ответа на запрос.

Поля ответа на запрос

Поле	Тип	Назначение	Комментарий
reqStatus	N	Статус проверки	Результат выполнения запроса
reqNote	S	Сообщение	Текстовое сообщение (комментарий) об ошибке обработки запроса
errUsrMsg	S	Сообщение плательщику	Сообщение, предназначенное для вывода непосредственно плательщику в случае невозможности выполнения операции
payeeRemain	MONEY	Остаток на лицевом счете получателя в минимальных единицах валюты	Поле присутствует в ответе на запрос если поле queryFlags установлен бит 0 (значение 0x0001). Валюта – всегда рубли В случае задолженности значение отрицательное.
payeeRemainDetails	ARRAY	Детализация остатка на лицевом счете в разрезе услуг/субсчетов	Поле присутствует в ответе на запрос если поле queryFlags установлен бит 1 (значение 0x0002). Описание детализации по услугам/субсчетам и частным суммам передается в том же формате, что и при совершении платежа, за исключением назначения платежа, см. п.3.12 В передаваемом списке должны присутствовать все субсчета, в том числе те, по которым текущий остаток 0 (нет ни задолженности ни аванса).

			Если субсчетов нет, то возвращаемый список должен быть пуст (допускается как отсутствие поля в ответе, так и пустое значение).
payeeRecPay	MONEY	Рекомендуемый платеж в минимальных единицах валюты	Поле присутствует в ответе на запрос, если в поле queryFlags установлен бит 2 (значение 0x0004). Валюта – всегда рубли.
payeeName	S	Инициалы абонента	Поле присутствует в ответе на запрос, если в поле queryFlags установлен бит 3 (значение 0x0008).

Пример запроса в формате HTTP POST:

reqType=queryPayeeInfo&svcTypeId=0&svcNum=9123456780&queryFlags=3

Пример ответа в формате HTTP POST:

reqStatus=0&payeeRemain=104500&payeeRemainDetails=3%7C20000%250D%250A5%7C84500

Пример запроса в формате JSON:

```
{
  "reqType": "queryPayeeInfo",
  "svcTypeId": "0",
  "svcNum": "9123456780",
  "queryFlags": 3
}
```

Пример ответа в формате JSON:

```
{
  "reqStatus": 0,
  "payeeRemain": 104500,
  "payeeRemainDetails": [{
    "svcSubNum": 3,
    "payAmount": 20000
  }, {
    "svcSubNum": 5,
    "payAmount": 84500
  }]
}
```

5.7 Запрос на поиск информации о ЛС абонента по адресу

Запрос предназначен для поиска ЛС абонентов, зарегистрированных по заданному адресу. Номера ЛС могут быть далее использованы Агентами для запроса другой информации по абонентам. Входными параметрами для запроса являются код региона (searchRegion), точное наименование города (searchCity), улицы (searchStreet), дома/корпуса (searchHouse) и квартиры (searchFlat). Результатом выполнения функции является список лицевых счетов, зарегистрированных по конкретному адресу. Результат поиска передается в виде одномерного массива, состоящего из значений лицевых счетов. Запрос выполняется только online.

Поля запроса

Поле	Тип	Назначение	Комментарий
reqType	S[64]	Тип запроса	findPayeeInfoByAddr
searchRegion	S	Регион	Составляющая адреса, по которому ведется поиск лицевых счетов. Перечень кодов регионов указывается в приложении к спецификации протокола ЕСПП-АСР.
searchCity	S	Город	Составляющая адреса, по которому ведется поиск лицевых счетов
searchStreet	S	Улица	Составляющая адреса, по которому ведется поиск лицевых счетов
searchHouse	S	Дом/корпус	Составляющая адреса, по которому ведется поиск лицевых счетов
searchFlat	S	Квартира	Составляющая адреса, по которому ведется поиск лицевых счетов

Поля ответа на запрос

Поле	Тип	Назначение	Комментарий
reqStatus	N	Статус поиска	Результат выполнения запроса
reqNote	S	Сообщение	Текстовое сообщение (комментарий) об ошибке обработки запроса
errUsrMsg	S	Сообщение плательщику	Сообщение, предназначенное для вывода непосредственно плательщику в случае невозможности выполнения операции
searchResults	ARRAY	Результаты поиска	Массив результатов поиска, состоящий из полей svcTypeId, svcNum, svcSubNum. Если поиск вернул нулевой результат, значение поля должно быть пустым.

Пример запроса в формате HTTP POST:

reqType=findPayeeInfoByAddr&searchRegion=RT.10&searchCity=%D0%9E%D0%BC%D1%81%D0%BA&searchStreet=%D0%AE%D1%80%D1%88%D0%B0&searchHouse=20%2F2&searchFlat=5

Пример ответа в формате HTTP POST:

reqStatus=0&searchResults=RT.DV.10.ACCOUNT_NUM%7C123456789%7C3%250D%250A%20RT.DV.10.ACCOUNT_NUM%7C123456789%7C5

Пример запроса в формате JSON:

```
{
  "reqType": "findPayeeInfoByAddr",
  "searchRegion": "RT.10",
  "searchCity": "Омск",
  "searchStreet": "Юрша",
  "searchHouse": "20/2",
  "searchFlat": "5"
}
```

Пример ответа в формате JSON:

```
{
  "reqStatus": 0,
  "searchResults": [{
    "svcTypeId": "RT.DV.10.ACCOUNT_NUM",
    "svcNum": "123456789",
    "svcSubNum": "3"
  }, {
    "svcTypeId": "RT.DV.10.ACCOUNT_NUM",
    "svcNum": "123456789",
    "svcSubNum": "5"
  }]
}
```

От Агента	От Оператора
ЗАО «Винус»	ОАО «Ростелеком»
Подпись: _____	Подпись: _____
М.П.	М.П.
Расшифровка подписи Л.К. Аксенова	Расшифровка подписи В.Г. Беленький
Должность: Генеральный директор	Должность: Заместитель директора Макрорегионального филиала – Коммерческий директор