RUSTACK CLOUD PLATFORM

Интеграция с ERP-системой для автоматизации процесса оплаты услуг юридических лиц

Релиз 1.2.0

2024

Оглавление

1. Термины и определения 3
2. Общая информация....4
3. Подготовка к интеграции....6
4. Описание процессов обмена данными....8
4.1. Получение контрагентов....8
4.1.1. Получение списка партнёров....8
4.1.2. Получение списка клиентов....10
4.1.3. Получение списка клиентов по принадлежности к партнёру....12
4.2. Получение тарифных планов....12
4.3. Получение детализации по потребляемым услугам и ресурсам....13
4.4. Передача платежей клиентов....15
4.5. Передача платежей партнёров....16
5. Описание структур данных....18
5.1. Структура данных «Клиенты»....18
5.2. Структура данных «Партнеры»....19
5.3. Структура данных «Потребляемые услуги»....19
5.4. Структура данных «Тарифные планы»....20
5.5. Структура данных «Платежи клиентов»....21
6. Приложение 1. Пример реализации интеграции....22

1. Термины и определения

Платформа (служба облачных вычислений) — набор облачных услуг («инфраструктуры как сервис», «платформы как сервис» и др.), предоставляемых потребителям через определённый интерфейс. Фактически, это одна или несколько инсталляций платформ виртуализации, в том числе территориально распределённых, под управлением RCP.

Оператор Службы облачных вычислений — организация, осуществляющая техническое и организационное администрирование Службы облачных вычислений, техническую поддержку пользователей.

Поставщик услуг (провайдер) — организация, продвигающая облачные услуги на рынке и заключающая договора с потребителями услуг. Поставщиком услуг может быть как сам Оператор, так и его бизнес партнёры. Для осуществления своей деятельности Поставщик услуг распоряжается частью ресурсов Службы облачных вычислений, выделенных ему Оператором.

Домен — Интернет-домен, с которого потребителям предоставляются услуги, включая функции регистрации, заказа, оплаты и администрирования услуг. По сути, домен является каналом предоставления услуг, которому назначается некоторая часть ресурсов Платформы. Для домена настраиваются собственные правила работы (лимиты, настройки по умолчанию), интеграция со сторонними службами (DNS, SMS, LDAP, ERP), визуальные настройки white-label. В Оркестраторе можно одновременно управлять несколькими доменами. За каждым доменом должен быть закреплён администратор либо со стороны организации-провайдера, либо партнёрской организации-поставщика услуг. В случае партнёрской схемы можно дополнительно настроить биллинг для взаиморасчётов партнёра с провайдером. В публичных облаках в зависимости от выбранной модели оплаты для домена могут быть сконфигурированы индивидуальные тарифные планы для каждого из потребителей услуг или один общий тарифный план-оферта. Для предоставления услуг в RCP должен быть сконфигурирован хотя бы один домен и связанный с ним партнёр.

Потребителем услуг (клиентом) может выступать организация, группа лиц, физическое лицо. В частном облаке это некоторый коллектив или подразделение, центр затрат для учёта потребления, центр финансовой ответственности. В публичном облаке это хозяйствующий субъект (юридическое или физическое лицо), потребляющий услуги облака по договору с поставщиком услуг (провайдером или его партнёром).

Артикул услуги — условное обозначение, состоящее из букв английского алфавита и цифр, присваивается той или иной услуге, предлагаемой клиенту в RCP. Артикулы услуг составляются в зависимости от выбранных классификаторов (модели предоставления услуги, категории услуги, выбора ресурсов или услуг, уровня предоставления, метрики потребления, шкалы измерения потребления, единиц измерения потребления и периода времени потребления). Цены на услуги определяются тарифными планами.

*Пример: резервированные в сегменте KVM виртуального дискового пространства уровня SAS 1 ГБ * 1 месяц: IS-KVM-VHDD-SAS-CA-R-GB-M*.

Баланс — отображает текущую сумму на виртуальном счёте потребителя в RCP. Баланс может быть положительным или отрицательным в зависимости от выбранной клиентом системы оплаты. Баланс уменьшается в соответствии с потреблением услуг и увеличивается в соответствии с пополнением виртуального счёта клиента.

Контракт (Договор) — сущность, моделирующая договор между контрагентами (сроки, цены, лимиты), а также лицевой счёт клиента (баланс). Контракты делятся на коммерческие, демо и внутренние. Внутренний контракт предназначен для учёта потребления в организации.

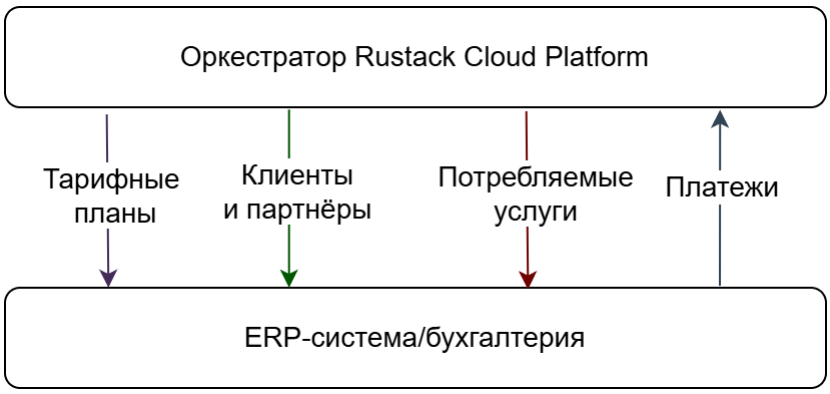
Тарифный план — отношение наименований предоставляемых услуг и цен за единицу услуги. Тарифные планы формируются провайдером услуг. Базовый тарифный план (тарифный план по умолчанию) используется в договоре-оферте, доступен для ознакомления пользователям, используется в калькуляторе для оценки стоимости. При заключении контракта выбирается или создаётся новый тарифный план.

2. Общая информация

Расчёты провайдера с клиентами (юридическими лицами) осуществляются по постоплатной системе. Оплата потребляемых услуг выполняется на основе счетов, выставляемых бухгалтерией провайдера. В свою очередь счета формируются на основе данных о потреблении клиентом облачных ресурсов. После оплаты клиенту направляется первичная документация (акт и счёт-фактура).

Интеграция учётной системы предприятия-провайдера (ERP) с подсистемой биллинга Оркестратора RCP упрощает процесс выставления счетов и формирования первичной документации за счёт автоматизации обмена данными между системами.

На схеме представлены потоки данных и их направление:



Подробное описание структур данных приводится в разделе Описание структур данных.

Соответствие сущностей в ERP и Оркестраторе:

RCP	ERP	Примечание
Клиент	Справочник контрагентов	Контрагент в ERP. В ERP также хранится соответствующий GUID клиента из Оркестратора. При этом ID контрагента в Оркестраторе не хранится.
Партнёр	Справочник контрагентов	Контрагент в ERP. В ERP также хранится соответствующий GUID партнёра из Оркестратора. При этом ID контрагента в Оркестраторе не хранится.
Тарифный план	Справочник тарифных планов	Справочник тарифных планов в ERP заполняется данными по тарифным планам из Оркестратора.
Контракт	Договор	Каждый договор в ERP имеет ID, и также хранится соответствующий ID контракта из Оркестратора. С контрагентом может быть заключено несколько договоров, но в Оркестраторе может быть заведён только один контракт на клиента/партнёра. Контракты загружаются в ERP из Оркестратора вместе со списком контрагентов.
Платёж	Оплаты	Для оплат в ERP фиксируется ID договора и ID контрагента. При передаче платежа из ERP в Оркестратор указываются GUID клиента/партнёра и ID контракта.
Биллинг классы	Номенклатура	Номенклатура заполняется данными по биллинг классам из Оркестратора.

Процессы обмена данными имеют различные условия инициации и периодичность:

№	Процесс	Условия инициации
1	Обновление справочников контрагентов в ERP на основе данных из Оркестратора	Ежедневно в автоматическом режиме регламентным заданием из ERP, либо вручную при необходимости
2	Сопоставление контрагентов в ERP с партнёрами и клиентами в Оркестраторе для автоматического учёта платежей	После загрузки данных о новых клиентах вручную пользователем ERP
3	Загрузка справочника тарифных планов из Оркестратора в ERP	• Ежедневно в автоматическом режиме регламентным заданием из ERP; • Вручную уполномоченным пользователем ERP при необходимости
4	Загрузка данных об оказанных услугах по договорам из Оркестратора в ERP	После окончания отчётного периода (месяца) вручную пользователем ERP
5	Формирование первичных документов: акт, счёт, счёт-фактура, на основании загруженных тарифных планов и данных о потреблённых услугах	Выполняется сотрудником бухгалтерии
6	Получение оплаты и проведение платежа в ERP	Выполняется сотрудником бухгалтерии
7	Передача информации об осуществлённых платежах	Ежедневно в автоматическом режиме регламентным заданием из ERP

3. Подготовка к интеграции

Используйте учётную запись пользователя с ролью бухгалтера в портале самообслуживания RCP. Дальнейшее подключение ERP-системы к Оркестратору выполняется под данной учётной записью через программный интерфейс с помощью токена доступа. Программный интерфейс Оркестратора RCP доступен по ссылкам:

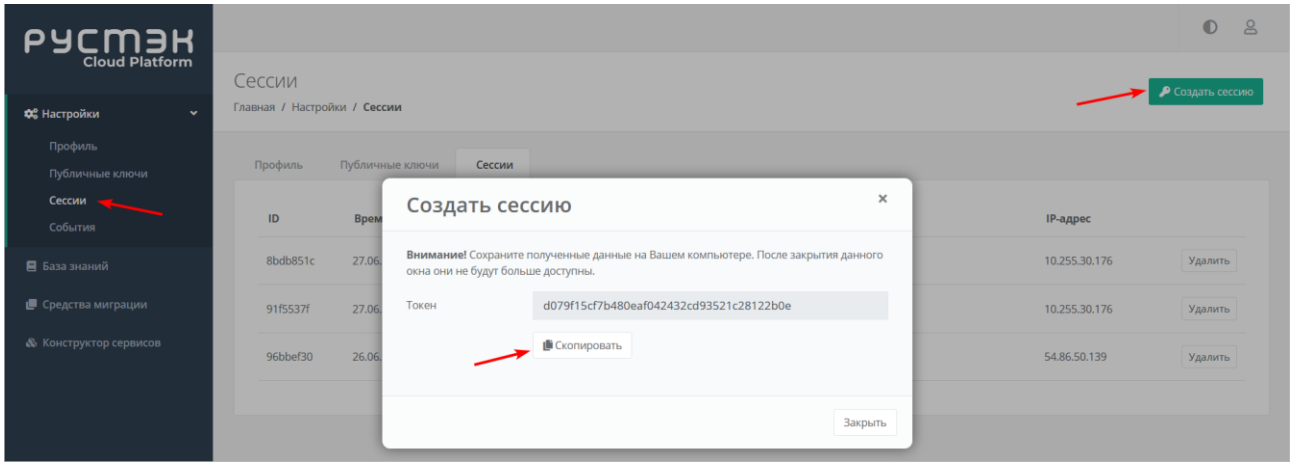
REST API: `https://{{control_panel_address}}/v1`

Swagger: `https://{{control_panel_address}}/swagger/`

Здесь `{{control_panel_address}}` — адрес портала самообслуживания, например, `newcp.cloud.online`. Совпадает с доменом личного кабинета.

Чтобы получить токен доступа в портале самообслуживания:

1. Авторизуйтесь под учётной записью бухгалтера.
2. Перейдите в раздел меню **Настройки** → **Сессии**.
3. Нажмите кнопку **Создать сессию**.
4. В открывшемся окне введите пароль и нажмите кнопку **Создать**.
5. В окне отобразится токен, скопируйте его.



Сохраните полученный токен доступа к Оркестратору в настройках ERP-системы для дальнейшего обмена данными. Токен отправляется в заголовках последующих запросов к API: `-H 'Authorization: Bearer $TOKEN'`.

Токен может быть получен и программным способом при наличии учётной записи (логина и пароля) с помощью метода **POST** по эндпоинту `/v1/auth/token`:

```
curl -X 'POST' 'https://{{control_panel_address}}/v1/auth/token' \
  -H 'Accept: application/json' \
  -H 'Content-Type: application/json' \
  -d   '{"code":   "string","domain":   "default","login":   "user_login","password":
"user_password","ttl": 1314000}'
```

Заполните поля запроса:

- `domain` — домен, в котором зарегистрирован пользователь (`default`),
- `login` — логин пользователя (`user_login`),
- `password` — пароль пользователя (`user_password`).

Ответ со статусом 200 будет содержать JSON-объект с токеном:

```
{
  "id":"16f3b266-d622-4b83-b541-90a50d7f2107",
  "key":"da3730bfe8f2e9fff1450c67a577b46dcb1b9df6",
  "ctime":"2023-04-05T17:09:15.799058",
  "issued_device_info":"string",
  "issued_ip_address":"string",
  "expires":"2023-04-20T22:09:15.795989"
}
```


Значение поля `"key"` будет содержать токен пользователя — `$TOKEN`. В данном примере это `da3730bfe8f2e9fff1450c67a577b46dcb1b9df6`.

Помимо настроек систем, в компании должны быть выстроены соответствующие бизнес-процессы по ведению справочников.

На этапе регистрации нового клиента в Оркестраторе в справочнике ERP-системы должна быть добавлена информация о новом контрагенте, так как в ERP и в Оркестраторе ведётся параллельный учет контрагентов (клиентов). Создание нового контрагента в ERP-системе выполняет ответственный сотрудник в соответствии с принятыми в компании регламентами по ведению справочников.

Аналогично информации о контрагентах и клиентах, данные о договорах фиксируются и в ERP-системе, и в Оркестраторе ответственными сотрудниками.

Сопоставление клиентов и партнёров в Оркестраторе с контрагентами в ERP выполняется вручную в ERP, подробнее в подразделе Получение контрагентов.

4. Описание процессов обмена данными

4.1. Получение контрагентов

В ERP-системе платежи за предоставленные услуги связаны с контрагентами, а в Оркестраторе с партнёрами и клиентами.

Загрузку контрагентов рекомендуется выполнять ежедневно в автоматическом режиме регламентным заданием ERP-системы.

В случаях нарушения работы регламентного задания или необходимости оперативной загрузки данных уполномоченный пользователь ERP-системы может запустить загрузку контрагентов.

После загрузки в ERP-систему информации о новых контрагентах и договорах пользователь ERP должен сопоставить их с соответствующими сущностями в ERP (процесс не автоматизирован).

В ERP-системе с клиентом и партнёром может быть зафиксировано несколько договоров. В Оркестраторе может быть заведён только один контракт (договор). Если требуется работа по разным контрактам, создайте дополнительную учётную карточку клиента (контракта). Таким образом, у организации может быть несколько учётных карточек. По сути карточка клиента играет роль контракта.

Карточке договора в ERP должен быть назначен соответствующий контракт в Оркестраторе.

4.1.1. Получение списка партнёров

Чтобы получить список партнёров, обратитесь методом **GET** к эндпойнту `/v1/partner`:

```
curl 'https://{{control_panel_address}}/v1/partner' -H 'Accept: application/json' -
H 'Authorization: Bearer $TOKEN'
```

Ответ со статусом 200 будет содержать массив объектов-партнёров в секции `items`, а также поля `total` (общее количество элементов) и `limit` (количество элементов на страницу):

```
{
  "items": [
    {
      "id": "d0cb1639-cc7e-44e1-b6ee-beca6cee0e3d",       // идентификатор партнёра
      "domain": {                                  // информация о домене партнёра
        "id": "ef2b8c17-83c1-4f8d-bc99-f1031c52d806",
        "name": "domain_partner1",
        "aliases": [
          {
            "id": "1f8336a9-587e-4d75-8c66-354e8771cb0a",
            "alias": "dev.partner1.online"
          }
        ],
        "initial_balance": 5,
        "initial_billing_plan": {
          "id": "e519e160-95f8-4b95-9e67-cba71bdfd31b",
          "name": "default",
          "contracts": 97
        },
        "initial_internet": 1000,
        "initial_lan": 1000,
        "initial_payment_model": "prepay",
        "initial_disable": false,
        "initial_payment_methods": [
          {
            "id": "bank",
            "name": "Безналичная оплата",
            "min_amount": 0
          }
        ],
        "initial_resource_planning": false,
        "initial_billing_enabled": true,
        "initial_billing_visible": true
      },
      "hypervisors": [
        {
          "id": "a532b7ce-a3db-480f-be3e-e0772f203c4f",
          "name": "РУСТЭК",
          "type": "kvm",
          "position": 111,
          "icon": null,
          "cpu_per_vm": 14,
          "ram_per_vm": 36,
          "zone": {...}
        }
      ],
      "name": "Партнёр №1",                                 // название партнёра
      "contract": {
        "id": "66b1e6a7-7b76-45f5-b86f-5369f8121509",
        "balance": -1872661,
        "counterparty_left": null,
        "counterparty_right": {
          "id": "d0cb1639-cc7e-44e1-b6ee-beca6cee0e3d",
          "name": "Партнёр №1",
          "entity": "partner"
```

```
        },
        "inactive": false,
        "is_new": false,
        "name": "Контракт для партнера Партнёр №1",
        "billing_plan": {
          "id": "2f98eeda-b9f0-4c98-9b0a-ed91310d8ac1",
          "name": "Тарифный план",
          "contracts": 7
        }
      }
    },
    ...
  ],
  "total": 8,  // всего элементов
  "limit": 10  // элементов на страницу
}
```

Из ответа зафиксируйте идентификатор партнёра и его название, информацию о контракте.

В случае если значение поля `total` *больше значения поля* `limit`*, следует рассчитать количество страниц, как результат деления* `total` *на* `limit` *и округлённое до единиц в большую сторону, и запросить последовательно все страницы, добавив к запросу параметр* `page` *со значением равным номеру страницы. Следует обратить внимание, что если не передать параметр* `page`*, то в ответе будет содержаться первые* `{значение limit}` *элементов списка.*

```
curl 'https://{{control_panel_address}}/v1/partner?page=1' -H 'Accept: application/json' -H 'Authorization: Bearer $TOKEN'
```

4.1.2. Получение списка клиентов

Чтобы получить список клиентов в Оркестраторе, обратитесь методом **GET** к эндпойнту `/v1/client`:

```
curl 'https://{{control_panel_address}}/v1/client' -H 'Accept: application/json' -H 'Authorization: Bearer $TOKEN'
```

Ответ со статусом 200 будет содержать массив объектов-клиентов в секции `items`, а также поля `total` и `limit`:

```
{
  "items": [
    {
      "id": "e2cb4854-53c7-456c-84f8-901e0700c876",      // идентификатор клиента
      "ctime": "2023-03-31T06:38:43.051283",             // дата создания клиента
      "projects": [
        {...},
        {...}
      ],
      "contact_emails": [                        // список администраторов клиента
        {
          "id": "02b665e8-42b7-4167-8e16-0b7eeab982fe",
          "username": "Николай Красовский",
          "email": null,
          "phone": ""
        },
        {...}
      ],
      "partner": {
        "id": "2e3be96a-e389-4cf2-98f8-34096a49ecdc",    // идентификатор партнёра
        "name": "ООО Облачное облако"                    // название партнёра
      },
      "internet": 1000,
      "allowed_hypervisors": [
        {...},
        {...}
      ],
      "lan": 1000,
      "payment_model": "prepay",                 // модель оплаты клиента
      "payment_methods": [                       // массив доступных методов оплаты
        {
          "id": "yandex",                        // идентификатор метода оплаты
          "name": "ЮKassa",                      // название метода оплаты
          "min_amount": 0
        }
      ],
      "resource_planning": false,
      "contract": {
        "id": "dda4a3f0-3aa4-4fa6-be7a-d6774d9506fd",
        "balance": -3917.92,                           // баланс клиента
        "counterparty_left": {
          "id": "2e3be96a-e389-4cf2-98f8-34096a49ecdc",
          "name": "ООО \"Облачное облако\"",
          "entity": "partner"
        },
        "counterparty_right": {
          "id": "e2cb4854-53c7-456c-84f8-901e0700c876",
          "name": "ООО \"СмартМониторинг\"",
          "entity": "client"
        },
        "inactive": false,
        "is_new": false,
        "name": "Контракт для клиента ООО \"СмартМониторинг\"",
        "billing_plan": {                              // информация о тарифном плане клиента
          "id": "a8507ae8-5968-4979-b655-86adf46a2fa6",  // идентификатор тарифного плана
          "name": "Выгодный план для клиентов",          // название тарифного плана
          "contracts": 4
        }
      },
      "name": "ООО СмартМониторинг",                         // название клиента
      "billing_enabled": true,
      "billing_visible": true,
```

```
      "job_id": null,
      "locked": false
    },
    ...
  ],
  "total": 73,   // всего элементов
  "limit": 10    // элементов на страницу
}
```

Из ответа зафиксируйте поля:

- идентификатор и название клиента,
- идентификатор и название партнёра,
- модель оплаты клиента:
  - `prepay` — предоплата,
  - `postpay` — постоплата;
- информацию о контракте,
- идентификаторы и названия доступных методов оплаты,
- идентификатор и название тарифного плана*,
- баланс клиента.

*Подробнее о загрузке тарифных планов в разделе Получение тарифных планов.

В случае если значение поля `total` больше значения поля `limit`, следует рассчитать количество страниц, как результат деления `total` на `limit` и округленное до единиц в большую сторону, и запросить последовательно все страницы, добавив к запросу параметр `page` со значением равным номеру страницы. Следует обратить внимание, что если не передать параметр `page` , то в ответе будет содержаться первые `{значение limit}` элементов списка.

```
curl 'https://{{control_panel_address}}/v1/client?page=1' -H 'Accept: application/json' -H 'Authorization: Bearer $TOKEN'
```

4.1.3. Получение списка клиентов по принадлежности к партнёру

Чтобы получить список клиентов по принадлежности к какому-либо партнёру, передайте параметр `partner` с указанием идентификатора нужного партнера:

```
curl 'https://{{control_panel_address}}/v1/client?page=1&partner=31425bd9-a037-4716-93d5-8ac97ab2f7ae' -H 'Accept: application/json' -H 'Authorization: Bearer $TOKEN'
```

В ответе будет содержаться такой же список клиентов, как описан выше, но все клиенты отфильтрованного списка принадлежат одному партнёру.

4.2. Получение тарифных планов

Чтобы определить общую стоимость услуг, предоставляемых клиенту, и создать первичные документы в ERP-системе компании-провайдера, необходимо получить данные о тарифных планах. Тарифные планы могут быть общедоступные и индивидуальные, для партнёров и для клиентов партнёров. Тарифный план включает информацию о номенклатуре услуг. Одна и та же услуга, имеющая уникальный артикул, в различных тарифных планах может иметь различные цены и наименование. Использование партнёром номенклатуры услуг провайдера в своей ERP-системе носит рекомендательный характер и упрощает выставление счётов партнёром своим клиентам.

Для каждой услуги в тарифном плане назначается цена за тариф: c / (m × Δt), где c — стоимость в валюте; m — единица измерения потребляемого ресурса: шт. или ГБ, Δt — период тарификации: минута, час, день, месяц, год. Например: предоставление дискового пространства уровня SSD — 0.7 руб. / (1 ГБ × час), предоставление виртуального процессора ESXi — 24 руб. / (1 шт × день) и т.д.

Для расчёта стоимости услуги за период потребления цена за тариф умножается на величину объёма потребления $V = Q \times \Delta T$, где Q — количество единиц потреблённого ресурса за время использования ΔT. Например, за 12 часов использования SSD диска объёмом 50 ГБ при цене за тариф 0.8 руб. / (1 ГБ × час) с баланса клиента спишется сумма 480 руб.

Периодичность списания оплаты может быть привязана к календарному началу периода. Например, клиент заказал услугу с помесячной тарификацией и привязкой к календарному периоду 5-го октября. С клиента спишется полная сумма за месяц, а следующее списание будет произведено 1-го ноября — в начале календарного месяца. Если привязка к календарному периоду отключена, то следующее списание будет произведено 5-го ноября.

Загрузку тарифных планов рекомендуется выполнять ежедневно в автоматическом режиме регламентным заданием ERP-системы.

В случаях нарушения работы регламентного задания или необходимости оперативной загрузки данных уполномоченный пользователь ERP-системы может запустить вручную загрузку тарифных планов.

Чтобы получить цены на ресурсы для определенного тарифного плана, обратитесь методом **GET** к эндпойнту `/v1/billing_plan/{pk}/price`, где `{pk}` — идентификатор тарифного плана:

```
curl     'https://{{control_panel_address}}/v1/billing_plan/a8507ae8-5968-4979-b655-
86adf46a2fa6/price' -H 'Accept: application/json' -H 'Authorization: Bearer $TOKEN'
```

Ответ со статусом 200 будет содержать массив услуг с ценами:

```
[
  {
    "id": "dd0a38a2-f4f2-4384-8398-48e72e2efadf",     // идентификатор биллинг-класса ресурса
    "cost": 10.0,                                     // цена за тариф
    "billing_plan": "a8507ae8-5968-4979-b655-86adf46a2fa6",
    "billing_class": {                                // информация о биллинг-классе
      "id": "kvm_hdd_ultrafast",   // мнемонический идентификатор биллинг-класса, артикул в
Оркестраторе
      "name": "Предоставление дискового пространства уровня SSD (KVM)", // название биллинг-
класса, номенклатура
      "sku_mask": "IS-KVM-VHDD-SSD-CA-P-GB-D",       // артикул биллинг-класса
      "measure": "ТБ"                                 // единица измерения потребляемой услуги
    },
    "name": "Предоставление дискового пространства уровня SSD (KVM)",
    "period": "hour",              // период тарификации
    "snap_to_period": false        // привязка к календарному началу периода (для дня, месяца
и года; если true, то снятие средств происходит в 00:00 UTC)
  }
  ...
]
```

4.3. Получение детализации по потребляемым услугам и ресурсам

Список потребляемых услуг с детализацией потребления за период и стоимостью услуг можно получить по каждому клиенту отдельно.

Запуск процесса получения информации о потреблённых клиентом услугах (расходах) из Оркестратора выполняется вручную, по завершению отчётного периода (месяца). На основании полученных данных о клиентах и партнёрах, тарифных планах и списка потребляемых услуг

формируется пакет документов: акты, счета и счета-фактуры. Услуги, потреблённые клиентами партнёра, должны учитываться провайдером как потреблённые партнёром по соответствующим контрактам с партнёром.

Соответственно счёт, выставляемый провайдером партнёру, должен включать в себя стоимость услуг, потреблённых клиентами партнёра по всем связанным контрактам в ценах контракта «провайдер-партнёр».

Чтобы получить список потребляемых услуг по клиенту, обратитесь методом **GET** к эндпойнту `/v1/billing_details`:

```
curl     'https://{{control_panel_address}}/v1/billing_details?client=e2cb4854-53c7-
456c-84f8-901e0700c876&start_date=2023-06-05&end_date=2023-06-19&part=day' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer $TOKEN'
```

Параметры строки запроса:

Параметр	Обязательный	Значение
`client`	да	идентификатор клиента
`start_date`	да	начало периода в формате гггг-мм-дд
`end_date`	да	конец периода в формате гггг-мм-дд
`part`	да	квант детализации: день (day), календарная неделя (week), календарный месяц (month), календарный год (year)
`billing_class`		мнемонический идентификатор биллинг-класса
`contract`		идентификатор договора
`partner`		идентификатор партнера

Из-за фиксированного кванта детализации список из ответа может охватывать период превышающий заданный. Примеры:

- при запросе периода 01.02.2024 (четверг) – 07.02.2024 (среда) с квантом детализации *неделя*, ответ будет содержать данные за две календарные недели: 29.01.2024 – 04.02.2024 и 05.02.2024 – 11.02.2024;
- при запросе периода 15.02.2024 – 15.03.2024 с квантом детализации *месяц*, ответ будет содержать данные за два календарных месяца: за февраль и март.

Ответ со статусом 200 будет содержать массив услуг, потребляемых клиентом:

```
[
  {
    "target": {                    // объект, для которого применима услуга
      "id": "985c7266-d759-42bb-9929-f294a8f598bd",
      "type": "vm",
      "name": "Первый сервер"
    },
    "billing_class": {
      "id": "vmware_cpu_3_2",
      "name": "Предоставление виртуального процессора 3.2 GHz (ESXi)",
      "measure": "шт."
    },
    "paid_seconds": 194400,    // объём потребления ресурса за квант детализации, выраженный в
```

```
секундах
    "quantity": 54,       // количество единиц потреблённого ресурса за период тарификации
    "cost": 54,           // сумма списания за указанный квант детализации
    "project": "3df4970f-5e0d-48d4-9c5f-5e2b94481c9e",
    "period": "hour",                                    // период тарификации
    "date": "2023-06-05"
  },
  ...
]
```

Если передать опциональные параметры `project` и/или `billing_class` (идентификатор биллинг-класса), то данные будут отфильтрованы по указанным параметрам:

```
curl     'https://{{control_panel_address}}/v1/billing_details?client=e2cb4854-53c7-
456c-84f8-901e0700c876&start_date=2023-06-05&end_date=2023-06-
19&part=day&billing_class=kvm_hdd_ultrafast' \
-H 'Accept: application/json' \
-H 'Authorization: Bearer $TOKEN'
```

4.4. Передача платежей клиентов

В ERP-системе бухгалтер регистрирует факт оплаты счёта клиентом. Ежедневно регламентным заданием ERP-система передаёт в Оркестратор информацию о поступивших платежах.

Оркестратор регистрирует операцию как платёж (пополнение) и увеличивает баланс клиента на сумму пополнения.

Данный платёж отображается в списке платежей по клиенту в Оркестраторе.

Чтобы передать платёж, обратитесь методом **POST** к эндпоинту `/v1/payment/bank_payment`:

```
curl -X 'POST' 'https://{{control_panel_address}}/v1/payment/bank_payment' \
 -H 'Accept: application/json' \
 -H 'Authorization: Bearer $TOKEN'
 -H 'Content-Type: application/json' \
 -d '{"amount":5000,"client":"client_id","details":"transaction_id"}'
```

Заполните поля тела запроса:

- `amount` — сумма платежа (5000),
- `client` — идентификатор клиента (`client_id`),
- `details` — идентификатор транзакции (`transaction_id`).

Ответ со статусом 200 будет содержать информацию о подтверждении регистрации платежа.

```
{
  "id": "80884414-f3ea-4f77-9185-7d52af5bdd0f",                // идентификатор платежа
  "contract": {
    "id": "dda4a3f0-3aa4-4fa6-be7a-d6774d9506fd",
    "name": "Контракт для клиента ООО \"СмартМониторинг\"",
    "balance": 6217.97
  },
  "client": {
    "id": "e2cb4854-53c7-456c-84f8-901e0700c876",
    "name": "ООО \"СмартМониторинг\""
  },
  "payment_method": {
    "id": "bank",                                              // идентификатор метода оплаты
    "name": "Безналичная оплата"                               // название метода оплаты
  },
  "transaction_id": "transaction_id",
  "status": "succeeded",                                       // статус платежа
  "error_reason": "",
  "amount": 3000.0,                                            // сумма платежа
  "ctime": "2023-06-23T12:22:24.108983",           // дата проведения платежа в Оркестраторе
  "user": {        // информация о пользователе, который выполнил платёж или коррекцию баланса
    "id": "02ba1cb3-4c87-4948-bf3c-f56a7b15487e",       // идентификатор пользователя
    "domain": {                                          // информация о домене пользователя
      "id": "6cd172fd-c96b-48f7-a812-abc1307abac4",
      "name": "cloud_qa",
      "aliases": [
        {
          "id": "834563b3-3f16-4b2e-904f-b0c05b8d26f1",
          "alias": "cloud.online"
        }
      ]
    },
    "login": "buhgalter",                                      // логин пользователя
    "username": "Бухгалтер",                                   // имя пользователя
    "is_ghost": false
  },
  "details": "transaction_id"
}
```

4.5. Передача платежей партнёров

В ERP-системе бухгалтер регистрирует факт оплаты счёта партнёром. Ежедневно регламентным заданием ERP-система передаёт в Оркестратор информацию о поступивших платежах.

Оркестратор регистрирует операцию как платёж (коррекцию баланса) и увеличивает баланс партнёра на сумму пополнения.

Чтобы передать платёж партнёра, обратитесь методом **PATCH** к эндпоинту `/v1/partner/{partner_id}/balance`, где `{partner_id}` — идентификатор партнёра:

```
curl -X 'PATCH' 'https://{{control_panel_address}}/v1/partner/{partner_id}/balance' \
 -H 'Accept: application/json' \
 -H 'Authorization: Bearer $TOKEN'
 -H 'Content-Type: application/json' \
 -d '{"amount":5000}'
```

Заполните поле `amount` (сумма платежа) в теле запроса.

Ответ со статусом 200 будет содержать структуру данных о партнёре.

5. Описание структур данных

5.1. Структура данных «Клиенты»

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
1	`id`	Идентификатор клиента	`string (uuid)`	Да	
2	`name`	Наименование клиента	`string`		
3	`ctime`	Дата создания	`datetime`		Формат даты ISO 8601: YYYY-MM-DDT00:00:00+03:00, где YYYY-MM-DD — год-месяц-число
4	`partner.id`	Идентификатор партнёра	`string (uuid)`		
5	`partner.name`	Наименование партнёра	`string`		
6	`payment_model`	Модель оплаты клиента	`string`		Возможные значения: `prepay`, `postpay`
7	`payment_methods.id`	Идентификатор метода оплаты	`string (uuid)`		
8	`payment_methods.name`	Наименование метода оплаты	`string`		Возможные значения: `yandex`, `bank`, `internal`
9	`payment_methods.min_amount`	Минимальная сумма	`integer`		
10	`contract.balance`	Баланс клиента	`float`		
11	`contract.billing_plan.id`	Идентификатор тарифного плана клиента	`string (uuid)`		
12	`contract.billing_plan.name`	Наименование тарифного плана клиента	`string`		

5.2. Структура данных «Партнеры»

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
1	`id`	Идентификатор партнера	`string (uuid)`	Да	
2	`name`	Наименование партнера	`string`		

5.3. Структура данных «Потребляемые услуги»

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
1	`id`	Идентификатор услуги	`string (uuid)`	Да	
2	`type`	Тип услуги	`string`		
3	`name`	Название услуги	`string`		
4	`billing_class.id`	Идентификатор биллинг-класса	`string`		
5	`billing_class.measure`	Единица измерения метрики потребления	`string`		
6	`billing_class.name`	Название биллинг-класса	`string`		
7	`paid_seconds`	Объём потребления ресурса за квант детализации, выраженный в секундах: $V_d = Q_d \times \Delta T_d$, где Q_d — количество единиц потреблённого ресурса (шт. или ГБ) за квант детализации ΔT_d	`float`		
8	`period`	Период тарификации (периодичность списания)	`string`		

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
9	`quantity`	Количество единиц потреблённого ресурса за период тарификации: $Q_{\Delta t} = V_d / \Delta t$, Δt — период тарификации, выраженный в секундах	`float`		
10	`cost`	Сумма списания за указанный квант детализации (день, календарная неделя, календарный месяц, календарный год)	`float`		
11	`project`	Идентификатор проекта	`string (uuid)`		
12	`date`	Дата списания	`datetime`		Формат даты ISO 8601: YYYY-MM-DDT00:00:00+03:00, где YYYY-MM-DD — год-месяц-число

5.4. Структура данных «Тарифные планы»

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
1	`id`	Идентификатор биллинг-класса ресурса	`string (uuid)`	Да	
2	`cost`	Стоимость	`float`		
3	`billing_plan`	Идентификатор тарифного плана	`string (uuid)`		
4	`billing_class.id`	Идентификатор биллинг-класса	`string`		
5	`billing_class.name`	Название биллинг-класса	`string`		
6	`billing_class.sku_mask`	Артикул	`string`		

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
7	`billing_class.sku_measure`	Единица измерения	`string`		
8	`name`	Наименование услуги	`string`		
9	`period`	Период тарификации (периодичность списания)	`string`		
10	`snap_to_period`	Привязка к календарному началу периода (для дня, месяца и года; если true, то снятие средств происходит в 00:00 UTC)	`bool`		

5.5. Структура данных «Платежи клиентов»

№	Реквизит	Описание	Тип поля	Обязательное	Ограничение допустимых значений
1	`client`	Идентификатор клиента в Оркестраторе, к которому относится платёж	`string (uuid)`	Да	
2	`amount`	Сумма платежа	`float`	Да	
3	`details`	Идентификатор транзакции	`string`	Да	

6. Приложение 1. Пример реализации интеграции

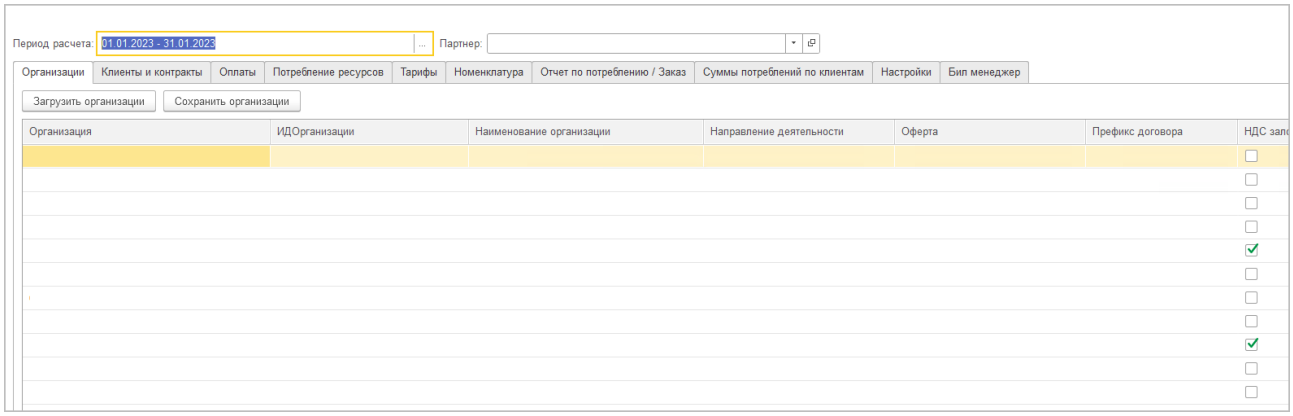
Для интеграции Оркестратора RCP и 1С:ERP реализованы две обработки в 1С:

1. Получение информации по тарифам, клиентам, партнёрам и их потреблении за выбранный период. Обработка позволяет связывать объекты из Оркестратора с контрагентами и договорами в ERP.
2. Передача информации о поступлении денежных средств от ЮЛ в RCP из 1С:ERP. Обработка ежедневно передает новую или измененную информацию по оплате в RCP. Обработка не имеет графического интерфейса.

Использование обработки обмена с RCP в 1С

Ручной запуск получения информации о партнерах (рисунок ниже):

1. Перейти в обработку обмена с RCP.
2. Перейти на вкладку **Организации.**
3. Нажать кнопку **Загрузить организации**.
4. Открыть карточку организации, задать контрагента.
5. Сохранить изменения.

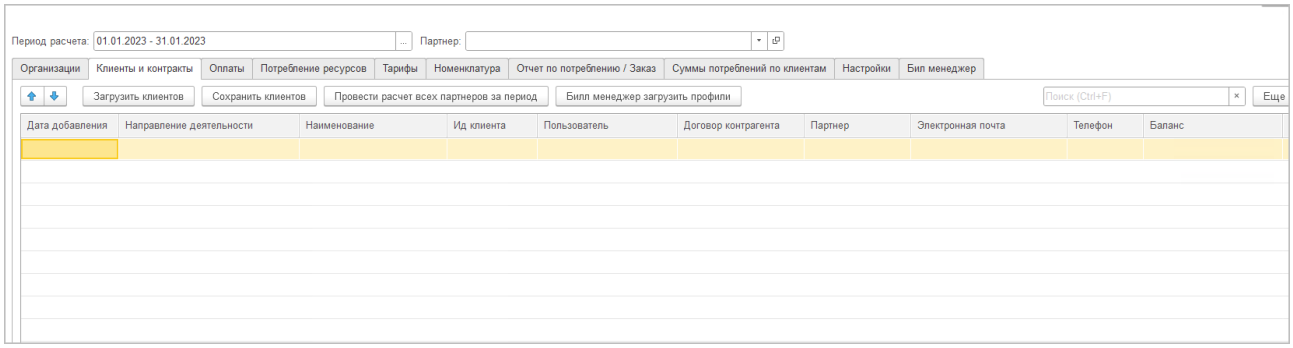


Ручной запуск получения информации о клиентах (рисунок ниже):

1. Перейти в обработку обмена с RCP.
2. Открыть вкладку **Клиенты и контракты**.
3. Если требуется загрузить клиентов определённого партнёра, выбрать партнёра из справочника.
4. Нажать кнопку **Загрузить клиентов**.
5. Открыть карточку клиента, задать контрагента и соответствующий договор.
6. Сохранить изменения.

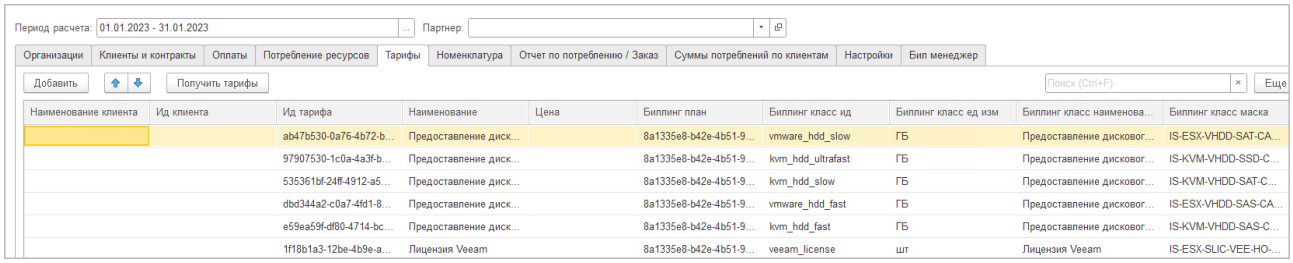
Расчёт стоимости потреблённых услуг (рисунок ниже):

1. Перейти в обработку обмена с RCP.
2. Открыть вкладку **Клиенты и контракты**.
3. Задать период расчёта.
4. Нажать кнопку **Провести расчёт всех партнёров за период**.
5. При необходимости загрузки информации по конкретному клиенту можно выбрать нужный контракт.



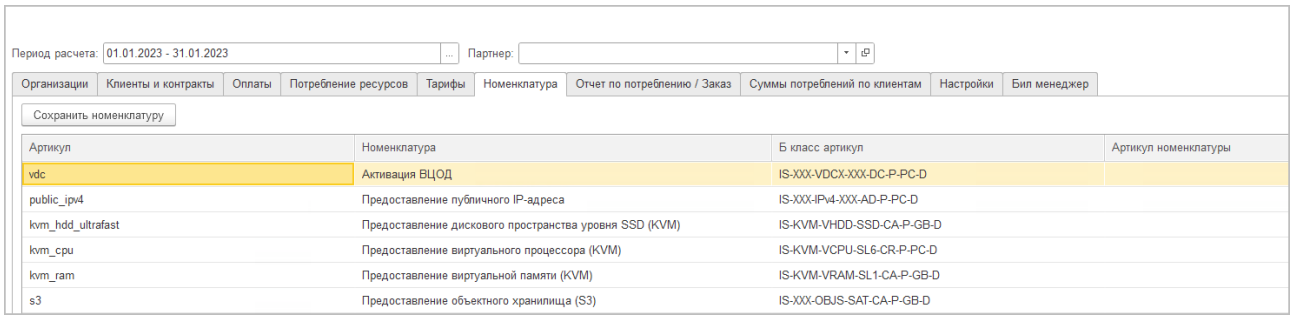
Ручной запуск получения информации о тарифах (рисунок ниже):

1. Перейти в обработку обмена с RCP.
2. Открыть вкладку **Тарифы.**
3. Нажать кнопку **Получить тарифы**.



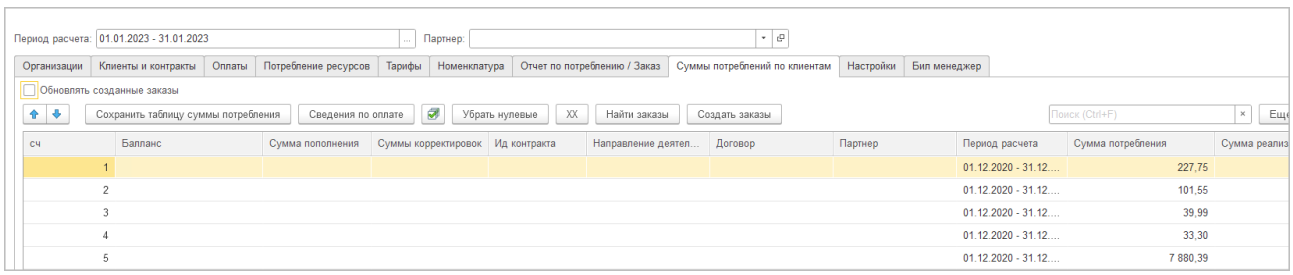
Просмотр информации о номенклатуре (рисунок ниже):

1. Перейти в обработку обмена с RCP.
2. Открыть вкладку **Номенклатура.**



Просмотр итоговой стоимости потреблённых услуг по клиентам/партнерам (рисунок ниже) за период:

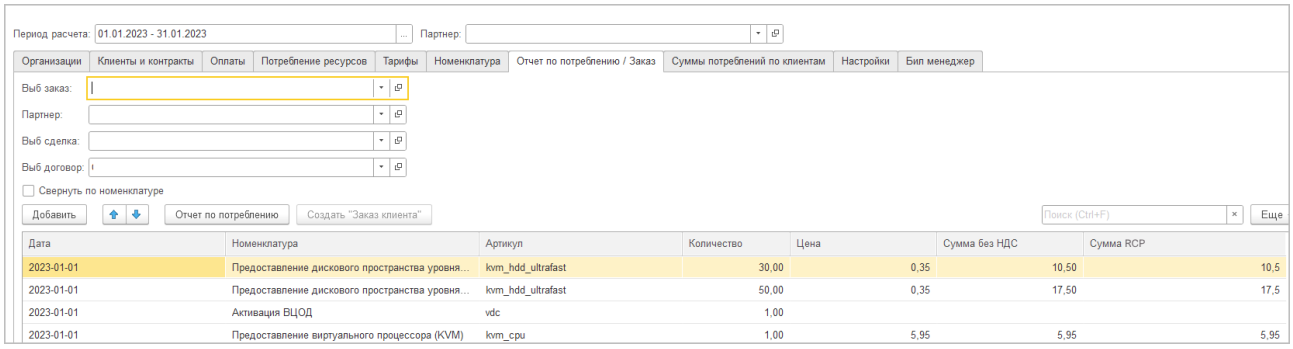
1. Перейти в обработку обмена с RCP.
2. Открыть вкладку **Суммы потреблений по клиентам**.
3. Задать расчётный период.
4. Выбрать в фильтре контрагента (партнера/клиента).

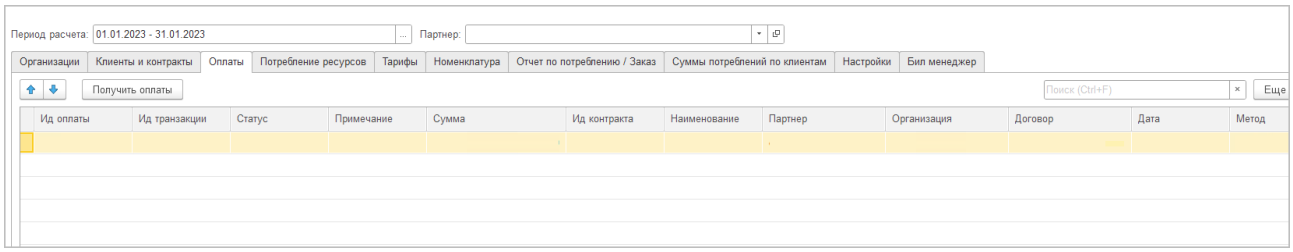


Просмотр детализации потребления услуг по клиентам/партнерам (рисунок ниже):

1. Перейти в обработку обмена с RCP.

2. Открыть вкладку **Отчет по потреблению/Заказ**.
3. Задать расчётный период.
4. Выбрать в фильтре контрагента (партнера/клиента) и договор.



Просмотр информации об оплатах (рисунок ниже):

1. Перейти в обработку обмена с RCP.
2. Открыть вкладку **Оплаты.**
3. Задать период расчёта.
4. Выбрать в фильтре контрагента (партнера/клиента) и договор.