

Экзо Платформа

Руководство администратора

Версия программного обеспечения: 1.0.0

Правообладатель: Индивидуальный предприниматель Архипов Дмитрий Викторович
ИНН 781431389513, ОГРНИП 306784706200235

2026

1. Назначение документа

Документ предназначен для системного администратора, разворачивающего и сопровождающего «Экзо Платформу». Он описывает подготовку инфраструктуры, установку и настройку модулей, первичную настройку доступа и регулярные операции эксплуатации.

Работа пользователя в графическом интерфейсе описана в руководстве пользователя. Состав и функциональные возможности платформы — в описании функциональных характеристик.

2. Архитектура развёртывания

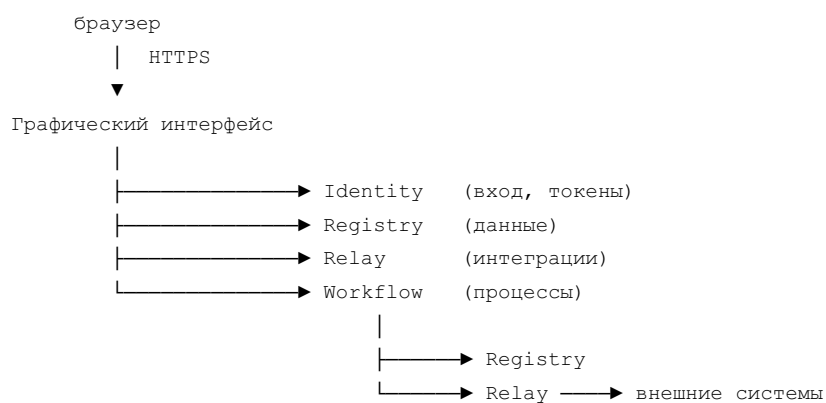
Платформа состоит из модулей, каждый из которых поставляется отдельным образом контейнера и запускается как самостоятельный процесс.

Компонент	Роль в развёртывании	Внешние зависимости
Identity	Выдаёт и проверяет токены доступа для всех остальных модулей и для интерфейса	Собственная база данных
Registry	Управляет данными и подключениями к системам хранения заказчика	Собственная база данных; базы данных прикладных решений
Relay	Обменивается данными с внешними системами	Собственная база данных
Workflow	Исполняет процессы, вызывает Registry и Relay	Собственная база данных
Графический интерфейс	Отдаёт веб-приложение браузеру	Обращается к модулям по их адресам

Каждому модулю требуется **собственная база данных** — модули не разделяют схему и не обращаются к таблицам друг друга. Взаимодействие идёт только через программные интерфейсы поверх HTTP.

Фоновые задания (вызовы коннекторов, таймеры процессов, отправка сообщений) выполняются внутри тех же процессов: очередь каждого модуля лежит в его собственной базе данных, и отдельного сервера очередей платформе не требуется.

Типовая схема потоков:



3. Подготовка инфраструктуры

3.1. Требования к серверу

Минимальная конфигурация — все компоненты, включая систему хранения, на одном узле; ознакомительная установка и небольшие прикладные решения:

Параметр	Значение
Процессор	2 ядра
Оперативная память	4 ГБ
Дисковое пространство	40 ГБ

Рекомендуемая конфигурация — промышленная эксплуатация, система хранения на отдельном узле:

Параметр	Узел приложений	Узел системы хранения
Процессор	4 ядра	от 4 ядер
Оперативная память	8 ГБ	от 8 ГБ
Дисковое пространство	60 ГБ	по объёму прикладных данных

Потребление растёт от числа одновременно работающих пользователей и интенсивности фоновых заданий — вызовов внешних систем и срабатываний таймеров процессов; дисковое пространство — от объёма прикладных данных, файловых вложений и глубины хранения журналов.

Образы модулей занимают порядка 10–15 ГБ дискового пространства до размещения прикладных данных; это учтено в приведённых значениях.

3.2. Контейнерная среда

Требуется среда исполнения контейнеров (Docker 24 и выше либо совместимая) и средство запуска групп контейнеров. Среда входит в состав всех поддерживаемых российских операционных систем; отдельная установка компонентов платформы на узел не требуется — интерпретатор и библиотеки входят в образы.

Проверка готовности узла: команда запуска контейнеров выполняется без ошибок, узел имеет доступ к реестру образов и к системам хранения, у процессов есть право записи в каталоги, смонтированные для файлового хранилища и журналов.

3.3. Базы данных

Внешняя система управления базами данных не является обязательной: по умолчанию модуль хранит данные в файле в примонтированном каталоге и создаёт структуру хранения при первом запуске, поэтому установка работает без отдельно развёрнутой СУБД.

При переходе на PostgreSQL создаются четыре базы данных — по одной на модуль — и учётные записи к ним с правом создания и изменения таблиц: миграции схемы модуль выполняет самостоятельно.

Модуль	Поддерживаемые СУБД	Назначение
Identity	PostgreSQL, SQLite	Пользователи, роли, приложения, токены, сеансы
Registry	PostgreSQL, SQLite	Источники, схемы, каталоги, задания
Relay	PostgreSQL, SQLite	Подключения, запросы, журнал обмена
Workflow	PostgreSQL, SQLite	Определения и экземпляры процессов, задачи, подписки на события

Встроенная файловая система хранения (SQLite) — режим по умолчанию. Она пригодна для ознакомительной установки, стендов и небольших прикладных решений с ограниченным числом одновременно работающих пользователей.

PostgreSQL 14 и выше, включая российские сборки, рекомендуется для промышленной

эксплуатации при высокой нагрузке, а также там, где предъявляются требования к резервированию и кластеризации: репликация, аварийное переключение, работа нескольких экземпляров модуля с одними и теми же данными. Выбор системы хранения задаётся переменной `DATABASE_URL` (п. 4.2).

Базы данных **прикладных решений** — те, которыми управляет Registry, — создаются отдельно и могут находиться под управлением любой из поддерживаемых систем хранения. Учётной записи, под которой Registry к ним подключается, требуются права на чтение и изменение данных, а при создании каталогов средствами платформы — и на изменение структуры.

4. Установка

4.1. Получение дистрибутива

Дистрибутивом являются образы контейнеров модулей. Передаются они одним из двух способов, по выбору заказчика.

Загрузка из реестра образов правообладателя. Заказчику выдаются учётные данные, узел авторизуется в реестре и загружает образы нужных версий. Способ удобен там, где у узла есть доступ в сеть правообладателя, и позволяет получать последующие обновления тем же путём.

Передача файлом. Образы выгружаются в архив и передаются заказчику согласованным способом; на узле архив загружается в локальное хранилище образов. Способ применяется для закрытых контуров, не имеющих доступа во внешнюю сеть.

В обоих случаях состав поставки определяется приобретённым набором модулей: образ каждого модуля и образ графического интерфейса. Образ внешней системы хранения (СУБД) в состав поставки не входит и для запуска не требуется: модули работают на встроенной системе хранения. Если заказчик выбирает PostgreSQL, её дистрибутив или образ он берёт из собственных источников.

4.2. Переменные окружения

Модули настраиваются переменными окружения. Значения проверяются при запуске: если обязательная переменная не задана или значение не проходит проверку, модуль не стартует и сообщает, какая именно переменная неверна. Это сделано намеренно — ошибка конфигурации обнаруживается при развёртывании, а не в работе.

Общие для всех модулей

Переменная	Назначение	По умолчанию
<code>RAILS_ENV</code>	Режим работы; для промышленной установки — <code>production</code>	<code>development</code>
<code>DATABASE_URL</code>	Строка подключения к базе данных модуля; задаётся при переходе на PostgreSQL	файл встроенной системы хранения в каталоге данных модуля
<code>PORT</code>	Порт, на котором модуль принимает запросы	<code>3000</code>
<code>RAILS_MAX_THREADS</code>	Число одновременно обрабатываемых запросов	<code>3</code>
<code>RAILS_LOG_LEVEL</code>	Уровень журналирования	<code>info</code>
<code>PIDFILE</code>	Путь к файлу идентификатора процесса	не задан
<code>SECRET_KEY_BASE</code>	Ключ подписи служебных данных приложения	задаётся обязательно

Проверка доступа (все модули, кроме Identity)

Переменная	Назначение	По умолчанию
ENABLE_AUTH	Включает проверку токенов доступа. Для промышленной установки — true	false
OAuth_SERVER_URL	Адрес модуля Identity	http://localhost:3007
OAuth_INTROSPECT_TOKEN	Проверять токен обращением к Identity вместо локальной проверки подписи	false
OAuth_INTROSPECTION_CLIENT_ID	Идентификатор приложения модуля в Identity	не задан
OAuth_INTROSPECTION_CLIENT_SECRET	Секрет этого приложения	не задан

Identity

Переменная	Назначение	По умолчанию
SERVICE_URL_HOST	Внешний адрес модуля; попадает в метаданные обнаружения и в поле издателя токена	http://localhost:3007
JWT_SECRET	Ключ подписи токенов	задаётся обязательно
ACCESS_TOKEN_JWT	Выдавать токены доступа в формате JWT	true
ACCESS_TOKEN_EXPIRES_IN	Срок жизни токена доступа в формате ISO 8601	PT2H
USE_REFRESH_TOKEN	Выдавать токены обновления при запросе области offline_access	true
SKIP_APPLICATION_AUTHORIZATION	Пропускать экран согласия пользователя при выдаче прав приложению	true
LOG_ISSUED_TOKENS	Журналировать выдаваемые токены: 0 — нет, 1 — состав утверждений, 2 — полностью	0

Значение LOG_ISSUED_TOKENS, отличное от нуля, включается только на время разбора неисправности: в журнал попадают сведения о выданных правах.

Registry

Переменная	Назначение	По умолчанию
SOLID_QUEUE_IN_PUMA	Выполнять фоновые задания внутри процесса модуля	true
JOB_CONCURRENCY	Число одновременно выполняемых заданий	1
LOCALS	Значения, подставляемые в описания источников данных (например, реквизиты подключения, которые не хранятся в базе)	не заданы
SPATIALITE_PATH	Путь к библиотеке пространственных функций для SQLite	mod_spatialite
TASKLIST_URL, TASKLIST_USERNAME, TASKLIST_PASSWORD	Адрес и учётные данные службы пользовательских задач	локальные значения

Relay

Переменная	Назначение	По умолчанию
SOLID_QUEUE_IN_PUMA	Выполнять фоновые задания внутри процесса модуля	true
JOB_CONCURRENCY	Число процессов, разбирающих очередь заданий	1
WEB_CONCURRENCY	Число рабочих процессов	не задано

Переменная	Назначение	По умолчанию
SERVICE_URL_HOST	Внешний адрес модуля; подставляется в порождаемые описания интерфейсов	127.0.0.1:3001
SERVICE_STRICT_SCHEMAS	Строгая проверка данных по схемам при работе в роли поставщика	false
SIGNATURE_TYPE	Механизм электронной подписи: openssl или cryptopro	openssl
SIGNATURE_CERTIFICATE, SIGNATURE_PASSWORD	Сертификат подписи и пароль к нему	не заданы
SMEV3_NODE_ID	Идентификатор узла в СМЭВ	не задан
SMEV3_POLLING_RATE	Период опроса очередей СМЭВ 3, секунды. Не задан — опрос не выполняется: установка, не работающая со СМЭВ, оставляет переменную пустой. Значения от минуты приводятся к целым минутам	не задан
SMEV3_ACK_MISSING	Подтверждать получение сообщений, не соответствующих ни одному подключению	false
SMEV3_DEV_URL, SMEV3_TEST_URL, SMEV3_PROD_URL, SMEV3_PORTAL_URL	Адреса сред СМЭВ	адреса сред по умолчанию
NOTIFY_URL	Адрес приёма уведомлений от внешних систем	не задан
VETIS_POLL_INTERVAL	Период опроса при работе с ВЕТИС, секунды	10
CRPT_AUTH_URLS	Адреса аутентификации системы маркировки	адреса по умолчанию

Пароли к сертификатам и секреты приложений передаются модулю переменными окружения и не хранятся в образе.

Workflow

Переменная	Назначение	По умолчанию
SOLID_QUEUE_IN_PUMA	Выполнять фоновые задания внутри процесса модуля	true
JOB_CONCURRENCY	Число одновременно выполняемых заданий: вызовы коннекторов и срабатывания таймеров	1
CAMUNDA_API_VERSION	Версия совместимого программного интерфейса, сообщаемая редактору моделей	8.7.0

4.3. Первичная инициализация

1. **Миграции.** При первом запуске каждый модуль приводит схему своей базы данных в актуальное состояние. Отдельной команды не требуется; в закрытых контурах миграции можно выполнить заранее отдельным запуском контейнера.
2. **Администратор.** В модуле Identity создаётся первая учётная запись и назначается роль администратора.
3. **Приложения модулей.** В Identity регистрируются приложения для каждого модуля (поток учётных данных клиента) и для графического интерфейса (поток авторизационного кода с РКСЕ, с указанием адресов перенаправления). Полученные идентификаторы и секреты прописываются модулям в OAUTH_INTROSPECTION_CLIENT_ID и OAUTH_INTROSPECTION_CLIENT_SECRET.
4. **Включение проверки доступа.** После регистрации приложений модулям выставляется ENABLE_AUTH=true и они перезапускаются.
5. **Проверка.** Выполняется проверка работоспособности по перечню из описания

функциональных характеристик.

5. Настройка модулей

5.1. Identity — приложения, роли, выдача токенов

- **Приложения.** Для каждого потребителя токенов заводится приложение: тип потока, области доступа, адреса перенаправления, секрет. Секрет отображается один раз при создании.
- **Роли.** Роль — именованный набор прав. Роли назначаются пользователям; действующий набор ролей попадает в токен и учитывается всеми модулями.
- **Состав утверждений токена.** Для приложения задаются правила на языке FEEL, определяющие, какие сведения о пользователе и его правах включаются в токен. Это точка, через которую прикладное решение получает нужные ему атрибуты, не меняя модуль.
- **Сеансы.** Список активных входов пользователя; администратор может завершить любой из них принудительно.
- **Срок жизни токенов** задаётся `ACCESS_TOKEN_EXPIRES_IN`; токены обновления выдаются только приложениям, запросившим область `offline_access`.

5.2. Registry — подключение источников данных

- **Источник** описывает подключение к системе хранения: тип, адрес, учётные данные, параметры драйвера. Реквизиты, которые не должны храниться в базе, подставляются через переменную `LOCALS`.
- **Схема** описывает структуру данных на языке JSON Schema и может использоваться несколькими каталогами.
- **Каталог** связывает схему с источником. При создании каталога платформа создаёт таблицу или коллекцию; при подключении существующей таблицы структура читается из системы хранения и остаётся неизменной.
- **Связи** между каталогами объявляются отдельно и работают в том числе между каталогами в разных системах хранения.
- **Проверка подключения** выполняется из интерфейса; ошибки подключения и прав доступа сообщаются с указанием причины, полученной от системы хранения.

5.3. Relay — подключения к внешним системам, сертификаты

- **Подключение** создаётся по описанию интерфейса внешней системы: загружается документ (WSDL, XSD, описание OpenAPI, метаданные OData) либо указывается адрес, по которому он доступен. Модуль строит схемы данных и перечень операций.
- **Учётные данные и сертификаты** внешней системы задаются в параметрах подключения; пароли к контейнерам ключей передаются переменными окружения.
- **Электронная подпись.** Для взаимодействия со СМЭВ выбирается механизм подписи (`SIGNATURE_TYPE`); при использовании ГОСТ на узле должен быть установлен криптопровайдер, а сертификат подписи указан в `SIGNATURE_CERTIFICATE`.
- **Проверка подключения** выполняется на встроенном эмуляторе, без обращения к боевой системе поставщика.
- **Наблюдение за очередями** — состояние заданий видно в базе данных модуля: очередь, отложенные и не выполнившиеся задания лежат в таблицах `solid_queue_*`.

5.4. Workflow — публикация процессов и форм

- **Формы** публикуются первыми: процесс, ссылающийся на неопубликованную форму, к публикации не принимается, и в отказе перечисляются все недостающие формы.
- **Публикация процесса** создаёт новую версию описания. Ранее запущенные экземпляры продолжают исполняться на своей версии; новые запускаются на последней.
- **Публикация из редактора.** Модуль принимает развёртывание непосредственно из

стандартного редактора моделей: в настройках подключения редактора указывается адрес модуля с префиксом совместимого интерфейса.

- **Коннекторы** служебных задач настраиваются в самой модели процесса; адреса и учётные данные внешних систем задаются параметрами задачи.
- **Инциденты.** Если задача исчерпала повторы, создаётся инцидент. После устранения причины администратор отмечает инцидент решённым, и шаг выполняется повторно без перезапуска процесса.

6. Эксплуатация

6.1. Резервное копирование и восстановление

Копированию подлежат:

- базы данных всех модулей — штатными средствами используемой СУБД, а на встроенной системе хранения — копированием файла базы из примонтированного каталога;
- базы данных прикладных решений, которыми управляет Registry;
- файловое хранилище вложений;
- значения переменных окружения и ключи подписи — вне резервной копии баз данных, в защищённом хранилище.

Восстановление: развернуть образы той же версии, восстановить базы данных и файловое хранилище, вернуть переменные окружения, запустить модули в порядке из п. 4.3.

Копированию подлежат две сущности, и обе доступны штатными средствами: **базы данных**, которые использует платформа, и **каталоги, примонтированные к контейнерам** (тома), где лежат файловые вложения и данные, сохраняемые между перезапусками.

Периодичность копирования, глубина хранения копий, ответственный и порядок проверки восстановления определяются требованиями заказчика к сохранности данных и закрепляются в его регламенте: платформа не накладывает на них ограничений и не требует остановки модулей на время копирования, если средство копирования СУБД поддерживает согласованный снимок.

6.2. Обновление версий

Обновление выполняется администратором заказчика по его решению; платформа не обновляется самостоятельно и не управляется извне.

Порядок: получить образы новой версии, остановить модули, снять резервную копию баз данных, запустить модули новой версии — миграции схемы выполняются автоматически при старте, — проверить работоспособность по перечню п. 4.2 описания функциональных характеристик. Откат: вернуть образы прежней версии и восстановить базы данных из копии, снятой перед обновлением.

Модули обновляются независимо друг от друга. Рекомендуемый порядок при обновлении всех: Identity, затем остальные модули, затем графический интерфейс.

6.3. Журналирование и наблюдение за состоянием

- **Журнал модуля** пишется в стандартный вывод контейнера и собирается средствами инфраструктуры заказчика. Подробность задаётся `RAILS_LOG_LEVEL`.
- **Журнал обмена** модуля Relay хранит запросы и ответы внешних систем — основной источник при разборе интеграционных ошибок.
- **Фоновые задания:** состояние заданий каждого модуля видно в его собственной базе данных — очередь, число попыток, задания, исчерпавшие повторы. Отдельного сервера очередей и его панели управления в поставке нет.
- **Состояние процессов** наблюдается через интерфейс модуля Workflow: активные экземпляры, переменные, ожидающие задачи, инциденты.

- **Готовность модуля** определяется ответом на запрос к его программному интерфейсу; этот же запрос используется как проба работоспособности в средствах оркестрации.

6.4. Типовые неисправности и их устранение

Признак	Вероятная причина	Действия
Модуль не стартует, в журнале — сообщение о переменной окружения	Не задана обязательная переменная или значение не прошло проверку	Исправить значение, указанное в сообщении, и перезапустить
Все запросы к модулю получают отказ в доступе	Недоступен Identity, неверный адрес <code>OAUTH_SERVER_URL</code> или неверные реквизиты приложения	Проверить доступность Identity и совпадение идентификатора и секрета приложения
Токен принимается одним модулем и отклоняется другим	Модули указывают на разные экземпляры Identity либо срок жизни токена истёк	Привести <code>OAUTH_SERVER_URL</code> к одному значению, проверить время на узлах
Запись в каталог не сохраняется, ошибка проверки	Данные не соответствуют схеме каталога	Сообщение об ошибке содержит перечень несоответствий с указанием места в документе
Каталог не создаётся	У учётной записи источника нет прав на изменение структуры	Выдать права либо подключить существующую таблицу без создания
Процесс остановился, у экземпляра инцидент	Служебная задача исчерпала повторы	Разобрать инцидент, устранить причину, отметить решённым — шаг выполнится повторно
Внешняя система отвечает ошибкой	Изменился её интерфейс, недействительны реквизиты или сертификат	Журнал обмена Relay содержит запрос и ответ целиком; проверить срок действия сертификата
Задания не выполняются	Отключено выполнение заданий внутри процесса модуля	Проверить <code>SOLID_QUEUE_IN_PUMA</code> ; задания, исчерпавшие повторы, лежат в <code>solid_queue_failed_executions</code>