

Настройка контейнерных компонентов

Сервис машинного обучения

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

Сервис машинного обучения	4
Подготовка к установке сервиса машинного обучения	4
Установить сервис машинного обучения	5
Обновить компоненты сервиса машинного обучения	8

Сервис машинного обучения

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

Сервис машинного обучения предназначен для прогнозирования значений на основании больших объемов исторических данных и текущих фактов. Подробнее: [Предиктивный анализ данных](#).

Важно. Для настройки сервиса машинного обучения необходимы базовые знания администрирования Docker, Linux или Windows.

Подготовка к установке сервиса машинного обучения

На заметку. Для корректной работы сервиса в приложении on-site требуется [лицензия](#) “creatio predictive service on-site”. По вопросам ее приобретения необходимо обратиться к ответственному менеджеру.

Для настройки сервиса необходим сервер (физический или виртуальный компьютер) с установленной ОС Linux или Windows.

Важно. Для промышленной среды рекомендуется использовать сервер с ОС Linux. Сервер на базе Windows можно использовать только для среды разработки. Для получения Docker-контейнеров, предназначенных для Windows, обратитесь в службу поддержки.

Установка компонентов сервиса выполняется с помощью ПО Docker. Для этого скачайте архив с конфигурационными файлами и скриптами установки. [Скачать архив](#). В зависимости от потребностей вашей компании можно использовать Docker Community Edition (CE) или Enterprise Edition (EE). Подробную информацию вы найдете в [документации Docker](#).

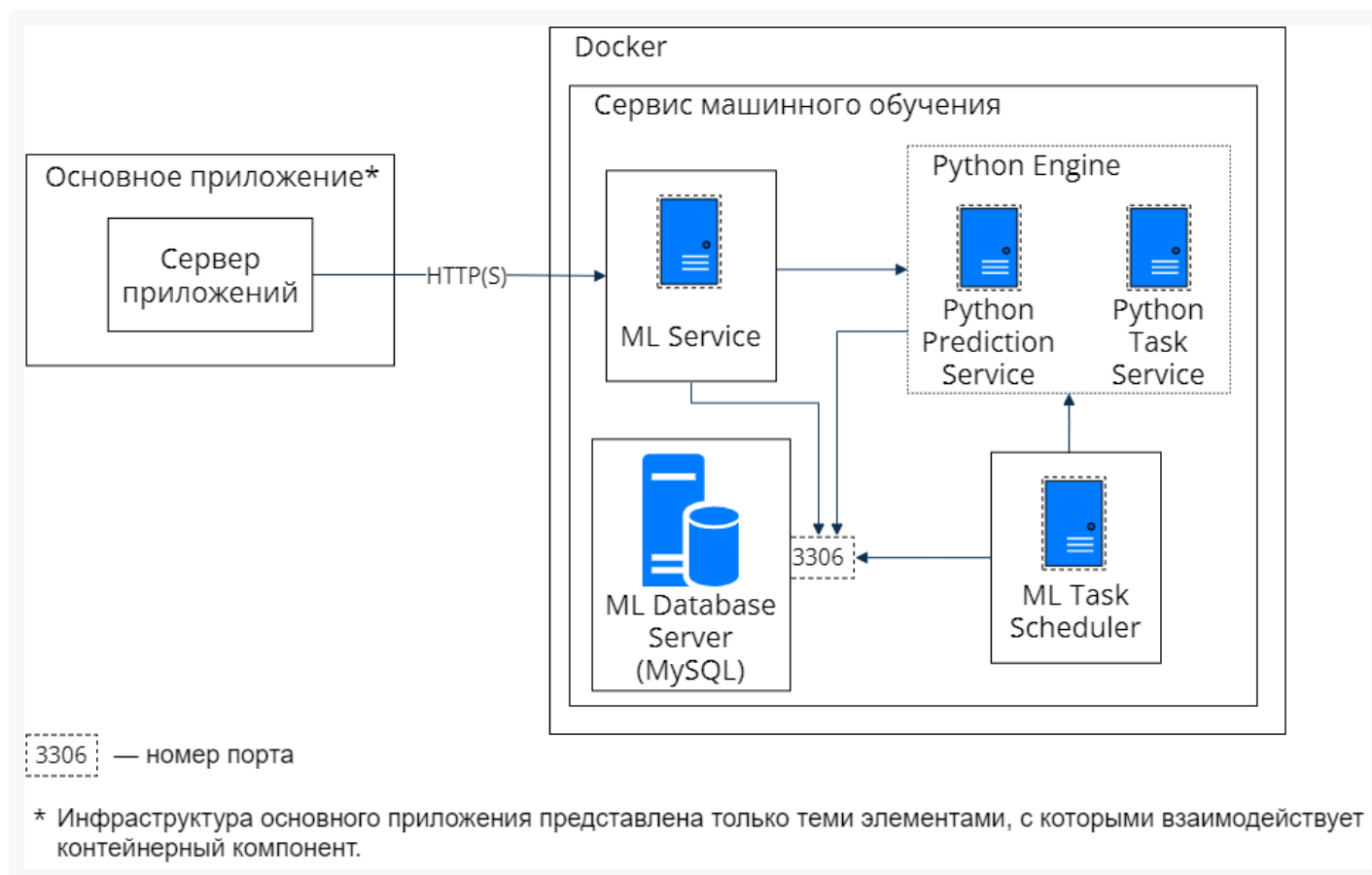
Для расчета требований к серверам воспользуйтесь [калькулятором системных требований](#).

Компоненты сервиса машинного обучения

Сервис машинного обучения использует следующие компоненты (Рис. 1):

- **ML Service** — web-сервис машинного обучения. Единственный компонент, доступный извне.
- **Python Engine** — движок машинного обучения, представляющий собой сервисную оболочку библиотек машинного обучения с открытым исходным кодом.
- **ML Task Scheduler** — планировщик задач.
- **MySQL** — база данных MySQL. Доступна по стандартному порту 3306.

Рис. 1 — Компоненты сервиса машинного обучения



Для удобства установки сервиса on-site все компоненты упакованы в Docker-образы.

Установить сервис машинного обучения

Последовательность действий для установки сервиса машинного обучения:

1. Установить Docker. [Подробнее >>>](#)
2. Установить Docker-Compose. [Подробнее >>>](#)
3. Установить и настроить компоненты сервиса. [Подробнее >>>](#)
4. Выполнить проверку установки. [Подробнее >>>](#)

Установить Docker

Для установки Docker на операционную систему Linux воспользуйтесь инструкцией в [документации Docker](#).

Для проверки установленной версии Docker выполните команду `docker --version` в терминале (консоли) Linux-машины.

Установить Docker-Compose

Для установки Docker на операционную систему Linux воспользуйтесь инструкцией в [документации](#)

[Docker](#).

Для проверки установленной версии Docker Compose выполните команду `docker-compose --version` в терминале (консоли) Linux-машины.

Настроить компоненты сервиса машинного обучения

Все контейнеры компонентов сервиса машинного обучения развертываются с помощью утилиты Docker Compose. Для этого скачайте конфигурационные файлы и скрипты, необходимые для развертывания и настройки компонентов сервиса. [Скачать архив](#)

На заметку. Конфигурационные файлы содержат все необходимые настройки по умолчанию для сервера на базе Linux.

Структура архива с конфигурационными файлами и скриптами:

`/etc/`

`...\ml-service\appsettings.json` — конфигурация web-сервиса.

`...\ml-service\log4net.config` — настройка уровня логирования web-сервиса.

`...\task-scheduler\appsettings.json` — конфигурация утилиты ML task scheduler.

`...\task-scheduler\log4net.config` — настройка уровня логирования ML Task Scheduler.

`docker-compose.yml` — конфигурация утилиты Docker Compose.

`.env` — файл с переменными окружения для запуска компонентов. Например, содержит пароль для MySQL.

Важно. Если необходимо изменить пароль к базе данных MySQL, то его нужно обновить не только в файле `.env`, но и в других конфигурационных файлах, содержащих секции настройки доступа к базе данных.

Установить компоненты сервиса машинного обучения

1. Скачайте и разархивируйте содержимое архива с конфигурационными файлами и скриптами в произвольный каталог, например, `/opt/ml`.
2. Используя терминал Linux, перейдите в каталог `docker-compose` распакованного архива, например, `/opt/ml/docker-compose`.
3. Выполните команду `sudo docker-compose pull` в терминале. Дождитесь завершения загрузки необходимых образов с компонентами сервиса с [Docker Hub](#).

Важно. Если на сервере запрещен доступ в интернет, скачайте на машине с открытым доступом все необходимые образы вручную (см.конфигурационный файл `docker-compose.yml`). Затем воспользуйтесь командами [sudo docker export](#) и [sudo docker import](#) для переноса образов в виде файлов на целевую машину.

4. Выполните команду **sudo docker-compose run dbmigration** для инициализации структуры базы данных. Дождитесь завершения выполнения команды.
5. Выполните команду **sudo docker-compose up -d** для запуска сервисов. Также в текущем каталоге будет создана папка logs.

Проверить установку компонентов сервиса машинного обучения

1. Для проверки установки веб-сервиса ML service выполните в терминале Linux команду:

```
curl -X GET localhost:5005/readiness
```

Сервис должен вернуть ответ:

Healthy

2. Для проверки запуска планировщика ML Task Scheduler выполните в терминале Linux команду:

```
curl -X GET localhost:5004/readiness
```

Сервис должен вернуть ответ:

Healthy

3. Для проверки запуска R Engine выполните в терминале Linux команду:

```
curl -X GET localhost:8081/readiness
```

Сервис должен вернуть ответ:

R Service is ready

4. Для проверки создания таблиц выполните в терминале команду:

```
docker exec -it DB Container Id mysql -u root --password=Supervisor ml -e "show tables;"
```

Где [*DB Container Id*] — идентификатор контейнера с компонентом базы данных. Узнать идентификатор контейнера можно с помощью команды **sudo docker ps**.

Пример. Проверка создания таблиц:

```
docker exec -it 7c9de07b53e8 mysql -u root --password=Supervisor ml -e "show tables;
```

В результате должны отобразиться названия основных таблиц сервиса: “modelinstance”, “traindata”, “trainsession” и т. д.

Заполнить настройки Creatio

Для работы с сервисом прогнозирования в Creatio необходимо заполнить:

1. Системную настройку “API ключ облачных сервисов Creatio” (код “CloudServicesAPIKey”) — отвечает за аутентификацию экземпляра Creatio в облачных сервисах.
2. Системную настройку “Периодичность запуска задач машинного обучения” (код “MLModelTrainingPeriodMinutes”) — определяет периодичность запуска синхронизации моделей.
3. Адрес развернутого сервиса прогнозирования в поле [*ServiceUrl*] для всех записей в справочнике [*Задачи машинного обучения*].

Обновить компоненты сервиса машинного обучения

Важно. Перед обновлением сервисов рекомендуется сохранить резервную копию базы данных MySQL. Подробную информацию вы найдете в [документации MySQL](#).

1. Используя терминал Linux, перейдите в каталог docker-compose с настроечными файлами, например, /opt/ml/docker-compose.
2. Выполните команду **sudo docker-compose stop** для остановки контейнеров компонентов сервиса.
3. Выполните команду **sudo docker-compose pull** в терминале. Дождитесь завершения загрузки необходимых образов с компонентами сервиса с [Docker Hub](#).
4. Выполните команду **sudo docker-compose run dbmigration** для инициализации структуры базы данных. Дождитесь завершения выполнения команды.
5. Выполните команду **sudo docker-compose up -d** для запуска сервисов.

Важно. Если у вас в приложении уже есть настроенные и обученные модели, после обновления сервиса рекомендуем их переобучить.