

Конфигурирование административного модуля

- [Файл настроек wcs-manager.properties](#)
- [Файл настроек log4j.properties](#)
- [Настройка базы данных WCS Manager](#)
 - [Создание базы](#)
 - [Наполнение базы](#)
 - [Миграция на MySQL](#)
 - [Установка и настройка MySQL](#)
 - [Настройка WCS](#)
 - [Импорт данных из H2 в MySQL](#)

Файл настроек wcs-manager.properties

Основные настройки административного модуля расположены в файле /usr/local/FlashphonerWebCallServer/conf/wcs-manager.properties.

Настройка	Значение по умолчанию	Описание
spring.datasource.url	jdbc:h2:C:/tmp/wcs_manager;FILE_LOCK=NO	JDBC URL базы данных
spring.datasource.driverClassName	org.h2.Driver	Класс драйвера базы данных
spring.datasource.database	org.hibernate.dialect.H2Dialect	Используемый диалект SQL
spring.datasource.username	admin	Имя пользователя
spring.datasource.password	admin	Пароль
spring.datasource.ddlMode	update	Создавать ли необходимые таблицы в базе данных при первом запуске create - создавать validate - не создавать, проверить целостность update - не создавать, обновить схему
spring.jpa.hibernate.ddl-auto	update	Автоматически создавать и заполнять базу при запуске: create - создавать validate - не создавать, проверить целостность update - не создавать, обновить схему create-drop - создавать, очищать таблицы при завершении сеанса
spring.jpa.hibernate.naming_strategy	org.hibernate.cfg.ImprovedNamingStrategy	Стратегия именования полей в таблицах БД
spring.jpa.database	H2	Используемый диалект БД
spring.jpa.show-sql	true	Показывать SQL запросы к базе.
server.port	9091	HTTP порт, на котором WCS Manager будет отвечать на входящие HTTP запросы.
manager.enable_https	true	Включить HTTPS

manager.https_port	8888	HTTPS порт
manager.keystore	wss.jks	Хранилище SSL сертификатов
manager.keystore_password	password	Пароль к хранилищу SSL сертификатов
spring.profiles.active	dev	Активный профиль. dev - Если включен этот профиль, WCS Manager будет слушать HTTP порт и не будет требовать аутентификации. production - Если включен этот профиль, то WCS Manager будет слушать HTTPS порт и будет требовать Basic-аутентификации от входящих HTTPS соединений. Профиль может быть переопределен при запуске сервера, как и другие настройки. См. раздел Параметры запуска .
node_api.port	1098	Локальный порт WCS Core к которому WCS Admin будет подключаться по RMI-интерфейсу для обмена данными.
node_api.service_name	NodeApi	Имя, под которым WCS Core зарегистрирован в RMI-реестре.
node_api.host	localhost	Хост, к которому будет осуществляться обращение по RMI.
rest_template.allow_self_signed	true	При обращении к Web-серверу по HTTPS/REST принимать от Web-сервера самоподписанные сертификаты.
shell.disabled_commands	autoconfig*,beans*,cron*,dashboard*,egrep*,endpoint*,env*,filter*,java*,jmx*,jul*,jvm*,jpa*,jdbc*,jndi*,less*,mail*,metrics*,shell*,sleep*,sort*,system*,thread*,repl*	Отфильтровать ненужные команды
spring.main.show_banner	false	Показывать баннер
node.enable_stdout	false	Писать лог WCS Server в файл logs/server_stdout.log. Может использоваться для отладки.
sun.rmi.dgc.client.gcInterval	36000000000	Задать интервал запуска сборщика мусора для RMI-клиента, мс. Данная настройка позволяет отключить принудительный запуск сборщика мусора, в целях повышения производительности сервера
sun.rmi.dgc.server.gcInterval	36000000000	Задать интервал запуска сборщика мусора для RMI-сервера, мс. Данная настройка позволяет отключить принудительный запуск сборщика мусора, в целях повышения производительности сервера
stream_stats.persist_data	false	Записывать статистику потоков в БД
server.session-timeout	10	Таймаут сессии, в минутах
loggc	/usr/local/FlashphonerWebCallServer/logs/gc-manager-	Расположение и префикс имени лога сборщика мусора. Имя формируется как /usr/local/FlashphonerWebCallServer/logs/gc-manager-YYYY-MM-DD_HH-MM.log
manager.rest_redirect_http_port	8081	Порт, на который перенаправляются REST API запросы, поступившие на порт server.port
manager.rest_redirect_https_port	8444	Порт, на который перенаправляются REST API запросы, поступившие на порт manager.https_port

Файл настроек log4j.properties

Часть настроек административного модуля находится внутри файла wcs_manager-1.0.jar:

Файл `log4j.properties` содержит настройки логирования WCS Manager. По умолчанию логи WCS Manager пишутся в файл `flashphoner_manager.log`. См. раздел [Логирование](#) для расширенных настроек и управления логами.

Для редактирования этих настроек можно воспользоваться `midnight commander` под Linux. Просто 'зайдите' в `wcs_manager-1.0.jar`, отредактируйте в нем файл и сохраните изменения.

Чтобы изменения вступили в силу, потребуется [перезагрузка сервера](#).

Настройка базы данных WCS Manager

Создание базы

WCS Manager по умолчанию использует встроенную (embedded) базу данных H2, которая создается и заполняется при первом запуске WCS-сервера.

При этом файлы базы данных создаются в домашней директории WCS-сервера в каталоге 'database'.

- Если файл базы данных не существует, при запуске будет предпринята попытка создать базу заново.
- Если файл базы уже есть, WCS Manager просто подключится к этой базе.

Наполнение базы

База будет наполняться значениями по умолчанию (пользователи, группы, приложения, и т.д.) только в том случае, когда база не содержит записи с супер группой.

Во всех остальных случаях содержимое базы не будет затрагиваться.

Миграция на MySQL

Установка и настройка MySQL

Установите MySQL (на примере CentOS 7) командами

```
rpm -Uvh http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm
yum install mysql-server
```

Запустите MySQL и установите пароль администратора

```
/sbin/service mysqld start
/usr/bin/mysqladmin -u root password 'rootpassword'
```

Войдите с правами администратора

```
mysql -uroot -prootpassword
```

Создайте базу данных и пользователя, дайте пользователю права на доступ к базе

```
create database wcs_db;
create user 'webcallserver'@'localhost' identified by 'password';
grant all privileges on wcs_db.* to 'webcallserver'@'localhost';
```

Настройка WCS

Добавьте следующие строки в файл [wcs-manager.properties](#)

```
-Dspring.datasource.url=jdbc:mysql://localhost/wcs_db
-Dspring.datasource.driverClassName=com.mysql.jdbc.Driver
-Dspring.datasource.database=org.hibernate.dialect.MySQL5Dialect
-Dspring.datasource.username=webcallserver
-Dspring.datasource.password=password
```

Здесь

- 'wcs_db' - имя базы данных
- 'webcallserver' and 'password' - имя и пароль пользователя

Перезапустите WCS сервер

```
service webcallserver restart
```

При первом запуске WCS, в wcs-manager.properties нужно выставить spring.datasource.ddlMode=create. В этом случае будут автоматически созданы необходимые таблицы. При повторном запуске нужно переключить настройку в spring.datasource.ddlMode=validate. В результате произойдет подключение к существующей базе данных уже без создания таблиц.

Импорт данных из H2 в MySQL

При миграции с H2 на MySQL данные, экспортированные из базы данных H2, могут быть импортированы в базу данных MySQL:

а) используя SQL скрипт

```
mysql -uroot -prootpassword wcs_db < wcs_h2_db.sql;
```

б) загрузив данные в таблицы из CSV файлов:

остановите WCS сервер

```
service webcallserver stop
```

войдите в MySQL

```
mysql -uroot -prootpassword
```

выберите базу данных

```
use wcs_db;
```

загрузите данные из CSV файлов в таблицы, например

```
LOAD DATA INFILE "/var/lib/mysql-files/STREAM_INFO_3.csv"
REPLACE INTO TABLE stream_info_3
FIELDS TERMINATED BY ','
OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 LINES;
```

Отметим, что SQL скрипт и CSV файлы, сгенерированные H2, не полностью совместимы с MySQL и требуют изменений для импорта. Например, для таблицы STREAM_INFO_3 потребуется изменить переменные типа Boolean.