

Инструкция по патчингу

Подготовка

Перед патчингом изменений необходимо:

1. Скачать нужный релиз БД (<https://github.com/essence-community/core-backend>)
2. Скачиваем patcher (<https://github.com/essence-community/core-backend/tree/master/patcher>)
3. Прописываем в PATH местоположение патчера

- Подготовка
- Патчинг платформы
 - Сервисы из t_query
 - Системные настройки
 - Ошибки
 - Страницы/каталоги/модули
 - Классы и их атрибуты
 - Локализация
- Патчинг внутренней авторизации

Патчинг платформы

1. Зайти в папку core-backend/dbms
2. в файле liquibase.properties указать путь к БД, с которым будем снимать дамп
3. запустить соответствующую команду в командной строке (из папки core-backend/dbms)

Что патчим	Команда	Комментарий	Формат имени файла
Запросы из t_query	<code>patcher -c query -qp " " -p liquibase.properties</code>	где "имя провайдера" - ИД провайдера, чьи запросы будем выгружать (ck_id из t_provider)	Querys_"Имя провайдера".sql
Системные настройки	<code>patcher -c syssetting -p liquibase.properties</code>	Выгружает все системные настройки	MSysSettings.sql
Ошибки	<code>patcher -c messages -p liquibase.properties</code>	Все значения из t_message, ck_id которых > 1000	MMessages.sql
Страницы/каталоги/модули	<code>patcher -c page -o "id //" -p liquibase.properties</code>	где "id модуля/каталога/страницы" - ck_id из t_page, которую будем выгружать.	Page_"Ид страницы".sql
Классы и их атрибуты	<code>patcher -c class -p liquibase.properties</code>	Выгружает все классы, значения их атрибутов, связи родительских и дочерних классов	MetaClass.sql
Локализация	<code>patcher -c lang -p liquibase.properties</code>	Выгружает все языки и словарь локализации. Выгружает файлы для каждого языка	Localization_"ИД языка".sql
Локализация конкретного языка	<code>patcher -c lang -o "код языка" -p liquibase.properties</code>	Выгружает словарь локализации указанного языка	Localization_"ИД языка".sql

Сервисы из t_query

Патчинг всех сервисов указанного провайдера из таблицы t_query

Команда
<code>patcher -c query -qp ck_provider -p liquibase.properties</code>

▼ где

ck_provider - ИД провайдера, чьи запросы будем выгружать (ck_id из t_provider)

После отработки патчера в папке core-backend/dbms/meta будет сформирован файл Querys_"Имя провайдера".sql

Если были добавлены новые сервисы, то их необходимо:

- включить их в файл core-backend/dbms/t_query/query.xml (Пример строки: `<include file="./t_query/MTClass.sql" />`)
- создать sql для каждого нового сервиса и выложить их в папку core-backend/dbms/t_query/

Системные настройки

Выгружает все системные настройки.

Команда

```
patcher -c syssetting -p liquibase.properties
```

После отработки патчера в папке core-backend/dbms/meta будет сформирован файл MSysSettings.sql

Ошибки

Выгружает все значения из t_message, ck_id которых > 1000.

Команда

```
patcher -c messages -p liquibase.properties
```

После отработки патчера в папке core-backend/dbms/meta будет сформирован файл MMessages.sql

Страницы/каталоги/модули

Выгружает модули, каталоги, страницы и все имеющиеся на них объекты, из связи и настройки.

Команда

```
patcher -c page -o "id_page" -p liquibase.properties
```

где

```
id_page - // (ck_id t_page)
```

При патчинге в папке core-backend/dbms/meta будет сформирован файл Page_"Ид страницы".sql

При патчинге каталога либо модуля будут также выгружены файлы для каждой вложенной страницы.

При данной команде также выгружаются слова/фразы из локализации, используемые на указанных страницах.

Если были созданы новые модули/каталоги/страницы, необходимо добавить их в файл core-backend/dbms/meta/meta.xml

Пример строки: <include file="./meta/Page_483653E7F5BC481D86375F5A3C803AC2.sql" />

При добавлении новых записей в файл необходимо соблюдать иерархию: сначала перечисляются все модули, затем каталоги и страницы.

Классы и их атрибуты

Выгружает все классы, атрибуты и значения атрибутов классов.

Команда

```
patcher -c class -p liquibase.properties
```

При патчинге в папке core-backend/dbms/meta будет сформирован файл MetaClass.sql

Локализация

Выгружает все языки и словарь локализации:

Команда
<pre>patcher -c lang -p liquibase.properties</pre>

Выгружает указанный язык и словарь для него:

Команда
<pre>patcher -c lang -o "id_lang" -p liquibase.properties</pre>

где

`id_lang - (ck_id t_d_lang)`

При патчинге в папке core-backend/dbms/meta будет сформирован файл Localization_"ИД языка".sql

Патчинг внутренней авторизации

Файлы будут выгружены в папку /core-backend/dbms_auth/meta/

1. Зайти в папку core-backend/dbms_auth
2. в файле liquibase.auth.properties указать путь к БД, с которым будем снимать дамп
3. запустить соответствующую команду в командной строке (из папки core-backend/dbms_auth)

Команда
<pre>patcher -c auth -p liquibase.auth.properties</pre>

Будут выгружены файлы:

- Accoun.sql - пользователи;
- Action.sql - действия;
- DInfo.sql - дополнительная информация пользователя;
- Roles.sql - роли.