

- [Главная](#)
- [Вопросы и ответы](#)

Экспорт mif/mid в PostgreSQL/PostGIS

Описание инструмента и примеры применения

[Обсудить в форуме](#) Комментариев — 0

Для конвертации данных mif/mid (обменный формат векторных данных Mainfo) в PostgreSQL/PostGIS удобно использовать следующий скрипт ([загрузить](#)). Этот скрипт был написан по причине текущей неспособности ogr2ogr полностью сконвертировать некоторые данные ([подробнее](#)).

Скрипт пригоден как для простой конвертации данных, так и для одновременного применения некоторых функций обработки. На данный момент доступна следующая функциональность:

- исключение указанных типов геометрий
- модификация структуры таблиц
 - транслитерация имен полей (необходим модуль Convert::Cyrillic)
 - переименование полей
 - пропуск полей по имени и/или их порядковому номеру
 - помещение дополнительных строк в определение таблиц
 - помещение дополнительных строк в вывод до любых сгенерированных данных, после выражений определения таблиц, после всех данных
 - генерация ddl/sql по собственному шаблону (необходим модуль Template)
- растяжение/сжатие (умножение/деление)
- отражение по x/y
- поворот на заданный градус
- удаление записей с невалидной геометрией

Для работы скрипта необходим модуль AppConfig. Для загрузки данных из mid необходим модуль Text::CSV_XS.

Исходные данные могут читаться как из файла, так и из STDIN. Вывод по-умолчанию в STDOUT, может быть направлен в указанный файл.

В одном файле mif могут находиться любые типы данных. Скрипт обрабатывает следующие типы геометрий: Point, Line, PLine, PLine Multiple, Region, Text (с привязкой в виде полигона). Прочие пропускаются. Поддерживаются только 2D данные.

PLine Multiple и Region, там где это возможно, конвертируются в полигоны. Незамкнутые объекты Region конвертируются в линии.

Имя столбца с геоданными по умолчанию geom. SRID по умолчанию -1.

Замечание. При конвертации данных создаются таблицы по количеству типов геометрий, они именуются так: к имени таблицы, заданному параметром --table, прибавляется окончание из имени геометрии. Например, если был указан ключ --table test, будут созданы таблицы test_poly, test_line, test_point. Если по окончании конвертации в таблицу не попало данных, она может быть удалена при помощи ключа --dropempty

Скрипт далек от совершенства, пожелания и сообщения об ошибках приветствуются.

Примеры использования

Справка по ключам

```
perl mif2pg.pl --help
```

Считать mif данные из data.mif, задать имя таблицы data, переименовать столбец id в old_id, добавить столбец

id типа serial, перена вывод в data.sql

```
perl mif2pg.pl --mif data.mif --table data --rename id=old_id --extra "id serial" > data.sql
```

Считать mif данные из STDIN, загрузить сразу в postgres:

```
cat data.mif | perl mif2pg.pl --mif - --table data --rename id=old_id --extra "id serial" \
| psql -U user db
```

Считать mif данные из data.mif, считать mid данные из data.mid, записать вывод в файл data.sql

```
perl mif2pg.pl --mif data.mif --mid data.mid --table data --rename id=old_id \
--extra "id serial" --out data.sql
```

Пропустить столбец id, не генерировать выражения ddl

```
perl mif2pg.pl --mif data.mif --mid data.mid --table data --skip id --nodd1
```

Задать SRID равным 4326, задать имя солбца с геоданными равным the_geom

```
perl mif2pg.pl --mif data.mif --table data --column the_geom --srid 4326 --out data.sql
```

Отразить по X, повернуть на 43.1232 градусов по часовой:

```
perl mif2pg.pl --mif data.mif --mid data.mid --table data --rotate 43.1232 --mirrorx \
--rename id=old_id --extra "id serial" --out data.sql
```

[Обсудить в форуме](#) Комментариев — 0

Ссылки по теме

- [Использование языка PERL для автоматизации работы с пространственными данными в среде Windows](#)

Последнее обновление: March 02 2011

Дата создания: 08.12.2008

Автор(ы): [Максим Дубинин](#)