

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Емельянов Сергей Геннадьевич

Должность: ректор

Дата подписания: 18.02.2023 14:50:14

Уникальный программный ключ:

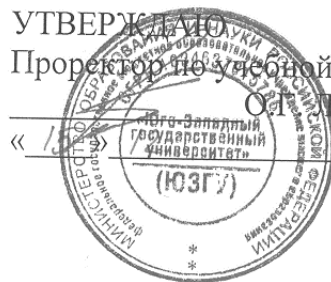
9ba7d3e34c012eba476ffd2d064cf2781953be730df2374d1c67c0ce536f0fc6

МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ
Проректор по учебной работе
О.А. Доктионова
« / » 2017 г.



ПОЛУЧЕНИЕ ДАННЫХ ИЗ POSTGIS

Методические указания по выполнению лабораторной работы
по дисциплине «Геоинформационные системы»
для студентов направления подготовки 09.04.04 «Программная инженерия»

Курск 2017

УДК 004.65

Составители: В.Г. Белов, Т.М. Белова

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

Получение данных из POSTGIS: методические указания по выполнению лабораторной работы по дисциплине «Геоинформационные системы» для студентов направления подготовки 09.04.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост.: В.Г. Белов, Т.М. Белова, – Курск, 2017. – 13 с.: ил. 13.

Изложена последовательность получения данных из PostGIS/PostgreSQL с использованием SQL или загрузчика/дампера шейп-файлов.

Материал предназначен для студентов направления подготовки 09.04.04 «Программная инженерия», а также будет полезен студентам всех направлений подготовки, изучающим технологии разработки пространственных баз данных.

Текст печатается в авторской редакции.

Подписано в печать 15.12.17 . Формат 60x84 1/16.

Усл. печ. л. 0,8. Уч.-изд. л. 0,7. Тираж 100 экз. Заказ 4439. Бесплатно.

Юго-Западный государственный университет.

305040, Курск, ул.50 лет Октября, 94.

Содержание

1	Получение данных из GIS	4
1.1	Использование SQL.....	4
1.2	Использование дампера	7
	Вопросы.....	12
	Индивидуальные задания	12

1 Получение данных из GIS

Данные могут быть извлечены из базы с помощью SQL или загрузчика/дампера шейп-файлов. В разделе об SQL будут описаны некоторые запросы к пространственным таблицам и операторы для сравнения данных.

1.1 Использование SQL

Самый простой способ получить данные из базы - использовать SQL-запрос "SELECT" с сохранением результирующих столбцов (см. рисунок 1).

The screenshot shows a database query interface. At the top, there is a text area containing the following SQL query:

```
1 SELECT * FROM public.roads
2 ORDER BY road_id ASC
```

Below the query, there are tabs for "Data Output", "Explain", "Messages", and "History". The "Data Output" tab is selected, displaying a table with the following data:

<input type="checkbox"/>	road_id [PK] integer	road_name character varying (20)	roads_geom geometry
<input type="checkbox"/>	1	Jeff Rd	010200000002000000000000...
<input type="checkbox"/>	2	Geordie Rd	010200000002000000000000...
<input type="checkbox"/>	3	Paul St	010200000002000000000000...
<input type="checkbox"/>	4	Graeme Ave	010200000002000000000000...
<input type="checkbox"/>	5	Phil Tce	010200000002000000000000...
<input type="checkbox"/>	6	Dave Cres	010200000002000000000000...

Рисунок 1 – Запрос "SELECT"

Для более удобного отображения значений столбца `roads_geom` выполните запрос, представленный на рисунке 2.

```

1 SELECT road_id, road_name, ST_AsText(roads_geom) AS geom
2 FROM public.roads;

```

Data Output			
Data Output Explain Messages History			
	road_id integer	road_name character varying (20)	geom text
<input type="checkbox"/>	1	Jeff Rd	LINestring(191232 243118,191108 243242)
<input type="checkbox"/>	2	Geordie Rd	LINestring(189141 244158,189265 244817)
<input type="checkbox"/>	3	Paul St	LINestring(192783 228138,192612 229814)
<input type="checkbox"/>	4	Graeme Ave	LINestring(189412 252431,189631 259122)
<input type="checkbox"/>	5	Phil Tce	LINestring(190131 224148,190871 228134)
<input type="checkbox"/>	6	Dave Cres	LINestring(198231 263418,198213 268322)

Рисунок 2 – Запрос с ST_AsText

Иногда необходимо сократить число возвращаемых полей. В случае ограничений на основе атрибутов, просто используйте тот же синтаксис SQL, как в случае обычной, непространственной таблицы. В случае пространственных ограничений можно использовать следующие операторы:

Таблица 1 - Операторы

&&	Этот оператор используется для проверки, пересекаются ли границы одной геометрии с границами другой.
~=	Этот оператор проверяет, являются ли две геометрии геометрически идентичными. Например, «POLYGON((0 0,1 1,1 0,0 0))» совпадает с «POLYGON((0 0,1 1,1 0,0 0))».
=	Более простой оператор. Он проверяет только совпадение границ геометрий.

Эти операторы могут быть использованы в запросах. Заметим, что если геометрии и охваты одновременно указываются в строке команды SQL, следует явно указывать необходимость конвертации строки в геометрию с помощью функции «ST_GeomFromText()». Пример приведён на рисунке 3.

```

1 SELECT road_id, road_name
2 FROM roads
3 WHERE roads_geom ~= ST_GeomFromText('LINESTRING(191232 243118,191108 243242)',-1);

```

Рисунок 3 – Запрос с оператором «~=»

Этот запрос возвратит единственную запись таблицы "ROADS_GEOM", геометрия которой равна указанному значению.

Data Output			
Explain			
Messages			
History			
	road_id	road_name	
	integer	character varying (20)	
<input type="checkbox"/>	1	Jeff Rd	

Рисунок 4 – Результат запроса 1

Когда используется оператор "&&", вы можете для сравнения указывать как BOX3D, так и GEOMETRY. Разумеется, если вы указываете GEOMETRY, для сравнения будет использоваться охват объекта.

```

1 SELECT road_id, road_name
2 FROM roads
3 WHERE roads_geom && ST_GeomFromText('POLYGON((189412 252431,189631 259122,189412 259122,189631 259122,189412 252431))',-1);

```

Data Output			
Explain			
Messages			
History			
	road_id	road_name	
	integer	character varying (20)	
<input type="checkbox"/>	4	Graeme Ave	

Рисунок 5 – Запрос с оператором «&&»

Этот запрос будет использовать для сравнения охват полигона.

Большинство простых пространственных запросов, вероятно, являются "фрейм-ориентированными" ("frame-based") запросами, используемыми в клиентском ПО таком, как обзорщики данных и приложения для веб-картографии, захватывающие "фрейм" данных для показа. Используйте для фрейма объект «BOX3D», как в запросе ниже на рисунке 6.

```

1 SELECT ST_AsText(roads_geom) AS geom
2 FROM roads
3 WHERE
4   roads_geom && ST_SetSRID('BOX3D(191232 243117,191232 243119)::box3d,-1);

```

Data Output Explain Messages History

<input type="checkbox"/>	geom text
<input type="checkbox"/>	LINestring(191232 243118,191108 243242)

Рисунок 6 – Запрос с объектом «BOX3D»

Обратите внимание, что при использовании SRID для определения проекции BOX3D значение -1 используется для указания неопределенного SRID.

1.2 Использование дампера

Дампер таблиц shp2pgsql соединяется с базой данных и конвертирует таблицу (возможно, заданную запросом) в шейп-файл. Дампер имеет графический интерфейс, для его запуска необходимо обратиться к файлу shp2pgsql-gui.exe. После запуска главное окно программы будет выглядеть соответственно рисунку 7.

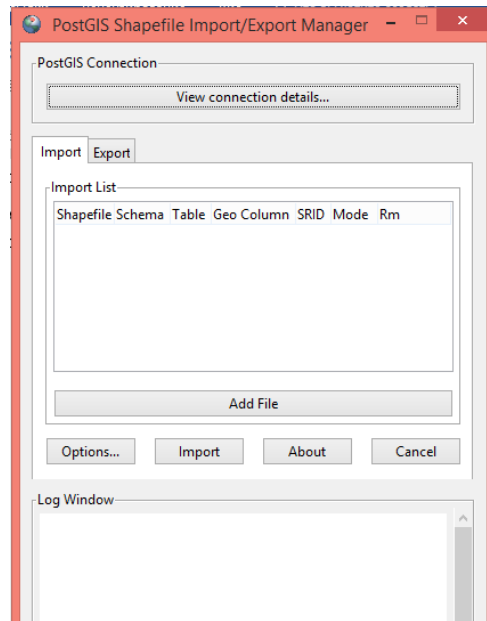


Рисунок 7 – Главное окно shp2pgsql-gui

Затем нажмите кнопку «View connection details...» для перехода в меню данных соединения с базой данных PostGIS/PostgreSQL. Заполните все поля появившейся формы (см. рисунок 8).

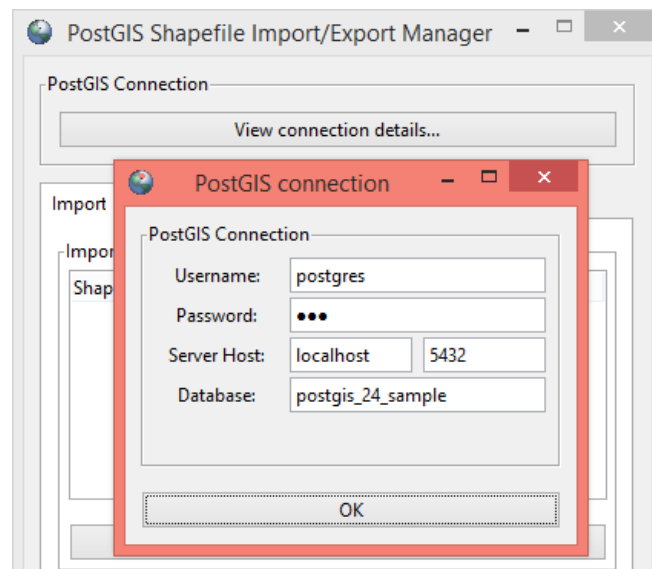


Рисунок 8 – Соединение с базой

Нажмите «OK». В результате в нижнем окне главной формы появится надпись соответственно рисунку 9.

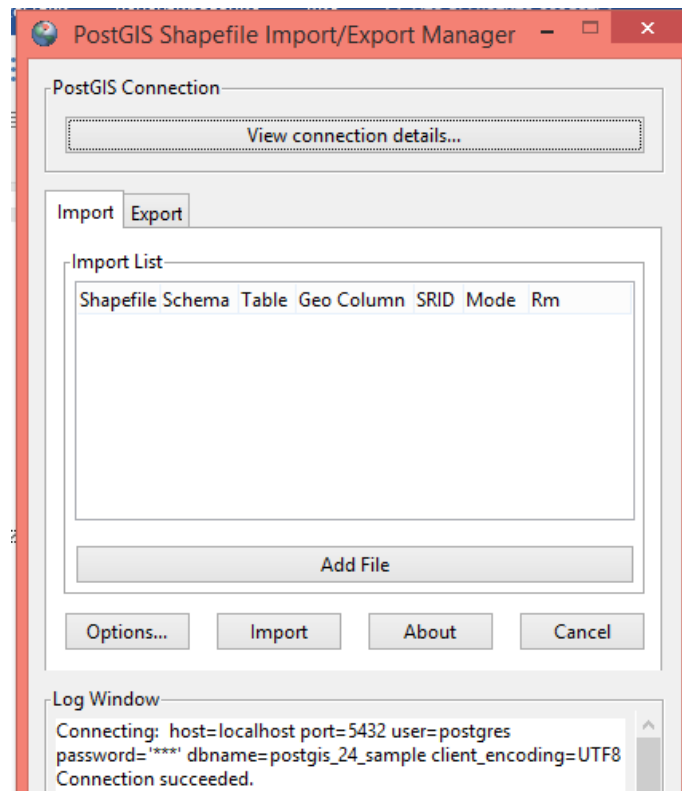


Рисунок 9 – Успешное соединение

Для записи данных таблицы в шейп-файл (файл формата *.shp) перейдите во вкладку «Export».

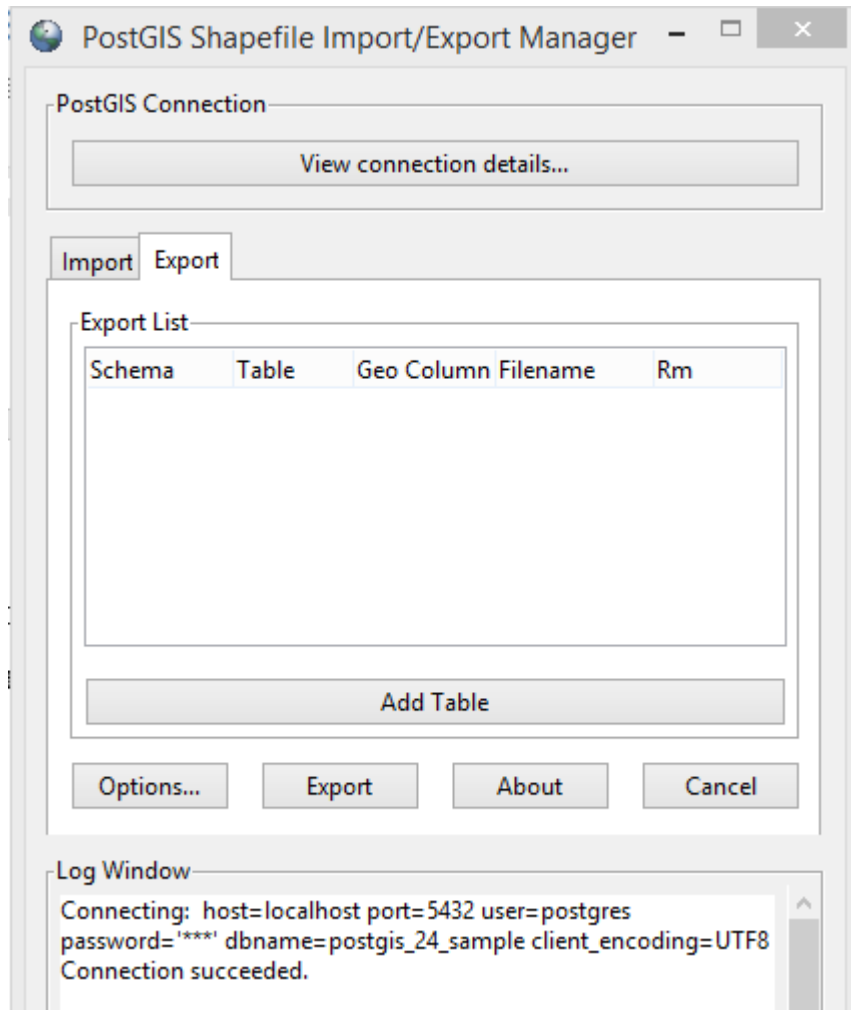


Рисунок 10 – Вкладка «Export»

Далее нажмите «Add Table», в появившемся меню выберите нужную таблицу и нажмите «ОК» (см. рисунок 11).

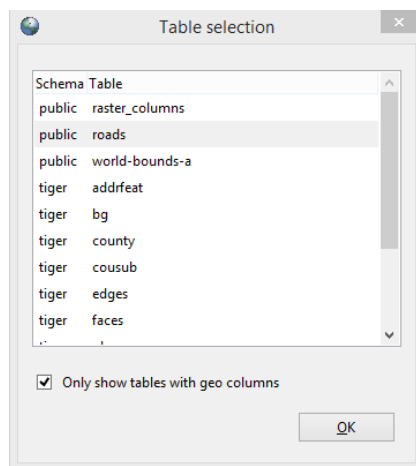


Рисунок 11 – Выбор таблицы

Выбранная таблица отобразится в меню «Export list». Далее нажмите кнопку «Export» и в проводнике укажите место сохранения файла *.shp. После удачной выгрузки данных главное окно программы будет выглядеть соответственно рисунку 12.

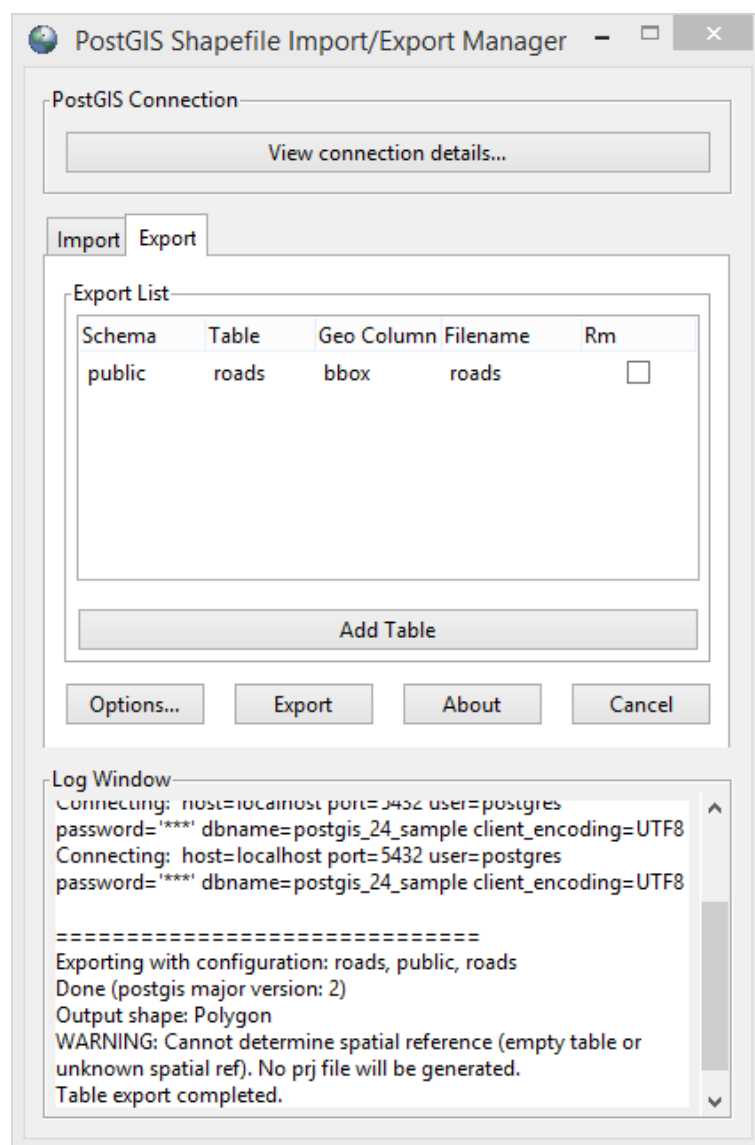


Рисунок 12 – Таблица выгружена

В указанной папке появятся файлы таблицы (см. рисунок 1.13).





 roads	11/8/2017 5:41 PM	Электронная таблица OpenOffice.org XML 1.0	3 KB
 roads.cpg	11/8/2017 5:41 PM	CPG File	1 KB
 roads.shp	11/8/2017 5:41 PM	SHP File	4 KB
 roads.shx	11/8/2017 5:41 PM	SHX File	1 KB

Рисунок 13 – Данные получены

Вопросы

1. Какими способами данные могут быть получены из базы?
2. Какая конструкция языка SQL позволяет получать данные таблицы?
3. Какая функция позволяет отображать данные формата geometry в текстовом виде?
4. Существует ли другой способ работы с базой данных PostgreSQL /PostGIS помимо pgAdmin?
5. Какие операторы используются в запросах для соотнесения границ объектов?
6. Какое приложение надо запустить для работы с дампером?
7. Какие данные надо сообщить дамперу для установки соединения с базой данных?
8. В какой вкладке программы осуществляется работа по выгрузке информации из базы данных?
9. Файл какого формата будет построен в результате работы дампера?
10. На основании каких данных базы будет строиться конечный файл данных?

Индивидуальные задания

1. С помощью простого SQL-запроса получите данные, хранящиеся в таблице test_table в среде pgAdmin.
2. С помощью SQL-запроса получите данные, хранящиеся в таблице test_table, изменив отображение столбца геометрии на текст. Выполните данный запрос в оболочке psql.
3. Выполните выборку данных из таблицы test_table, используя один из операторов пространственных ограничений.
4. Получите данные из таблицы test_table, в запросе используйте объект «BOX3D».

5. Сохраните данные таблицы `test_table` на компьютере в файл формата `*.shp` с помощью дампера.