DHAROHAR

Installation manual

DHAROHAR PROJECT INSTALLATION

Software that are required:

S.No.	Software	Links
1.	Python 2.7.3 or higher	Python releases
2.	Microsoft Visual C++ Compiler 9.0 for Python 2.7	MS VC++ Compiler for Python 2.7
3.	Qt Creator v4.0.1(based on Qt 5.6.1)	Qt Creator
4.	OpenCV 2.4.3 or higher	Download OpenCV
5.	CMake 3.10.win64-x64.msi	CMake-3.10.0-rc4-win64-x64
6.	PostgreSQL 9.3	Download PostgreSQL
7.	GDAL	http://gisinternals.com/
8.	Python wheel files	https://www.lfd.uci.edu/~gohlke/pythonlibs/

Installing Python and bindings:

Install Python 2.7 (minimum version Python 2.7.3 and 32 bit) **into the default path (otherwise, you will have to change a lot of paths in the project) and include that path** (i.e., *"C:\Python27\"*) to PATH environment variable.

Download and install Visual C++ Compiler for Python 2.7 from MS VC++ Compiler.

Install the following Python bindings using pip: GDAL, matplotlib, numpy, pygame, PyQt4, scikit-image, scikit-learn, scipy, secure-smtplib, affine, imutils, pandas, psycopg2,cx_freeze. To install the bindings, there are two ways:

- First and recommended way is to type *"C:\Python27\Scripts\pip install <binding-name>"* in the command prompt. This will install the most compatible version for your Python.
- Otherwise, if the first one does not work, go to this page Python Extension Packages for Windows - Christoph Gohlke and download the corresponding wheel file for the binding you want to install. Make sure you download the proper version matching the Python version with your Python installed. Then open command prompt in the folder where you have downloaded the wheel file and type *"C:\Python27\Scripts\pip install <binding-name-with-extension>"*. This will also install them.

Installing Qt Creator:

Install Qt Creator with the **default settings**.

Add *"C:\Qt\Qt5.6.1\Tools\mingw492_32\bin\"* and *"C:\Qt\Qt5.6.1\5.6\mingw49_32\bin\"* to PATH environment variable.

Then create a new variable named **QT5_ROOT_PATH** and set the value to *"C:\Qt\Qt5.6.1\5.6\mingw49_32\bin\"*.

Installing PostgreSQL and PostGIS:

Install PostgreSQL 9.3 into the **default path (please try to install it in the default path)**. While installing PostgreSQL, try to keep both **username and password as "postgres" and the port 5432** for convenience. After installing PostgreSQL, install only PostGIS2.2 bundle for your system from Stack Builder.

Add folders *"C:\Program Files\PostgreSQL\9.3\bin"* and *"C:\Program Files\PostgreSQL\9.3\lib"* to PATH environment variable. Path variables will change according to your version of postgres sql.

Database Manual:

To run Dharohar application with the database, make sure the database server accepts remote connections. To access the database server remotely, the network address of the server must be mentioned in the **pg_hba.conf** file.

One more thing that you need to be sure of is that appropriate PostGIS Bundle for PostgreSQL database should be installed on the machine to make Dharohar work properly.

Create new database:

After creating a new database, run the SQL script "setupDatabase.sql" on the database. To run the script from command line, follow the syntax below.

```
psql –U "username" -d "database_name> -h "host_address" -p "port" < setupDatabase.sql
```

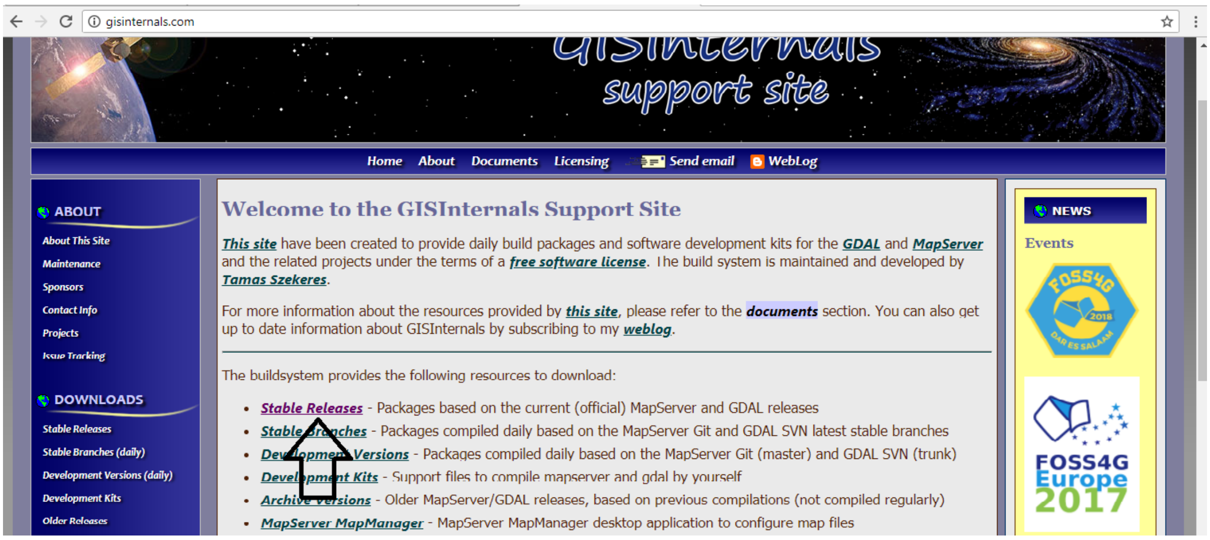
setupDatabase.sql file is attached with this document.

Provide the password of the user on prompt.

GDAL INSTALLATION:

Install the required gdal according to your processesor requirements.

Gdal can be downloading from the website http://gisinternals.com/



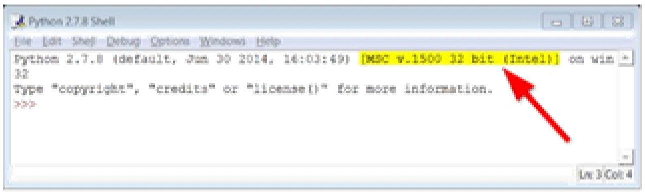
Welcome to the GISInternals Support Site

This site have been created to provide daily build packages and software development kits for the *GDAL* and *MapServer* and the related projects under the terms of a *free software license*. The build system is maintained and developed by *Tamas Szekeres*.

For more information about the resources provided by *this site*, please refer to the **documents** section. You can also get up to date information about GISInternals by subscribing to my *weblog*.

The buildsystem provides the following resources to download:

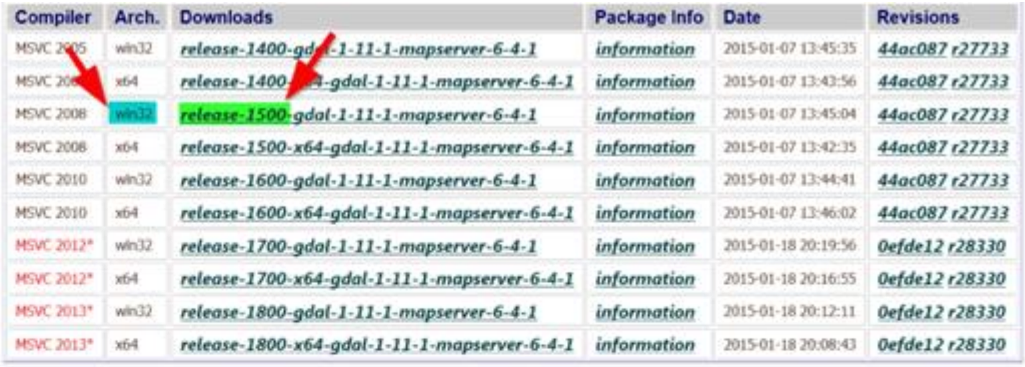
- *Stable Releases* - Packages based on the current (official) MapServer and GDAL releases
- *Stable Branches* - Packages compiled daily based on the MapServer Git and GDAL SVN latest stable branches
- *Development Versions* - Packages compiled daily based on the MapServer Git (master) and GDAL SVN (trunk)
- *Development Kits* - Support files to compile mapserver and gdal by yourself
- *Archive Versions* - Older MapServer/GDAL releases, based on previous compilations (not compiled regularly)
- *MapServer MapManager* - MapServer MapManager desktop application to configure map files



```
Python 2.7.8 (default, Jun 30 2014, 16:03:49) [MSC v.1500 32 bit (Intel)] on win
32
Type "copyright", "credits" or "license()" for more information.
>>>
```

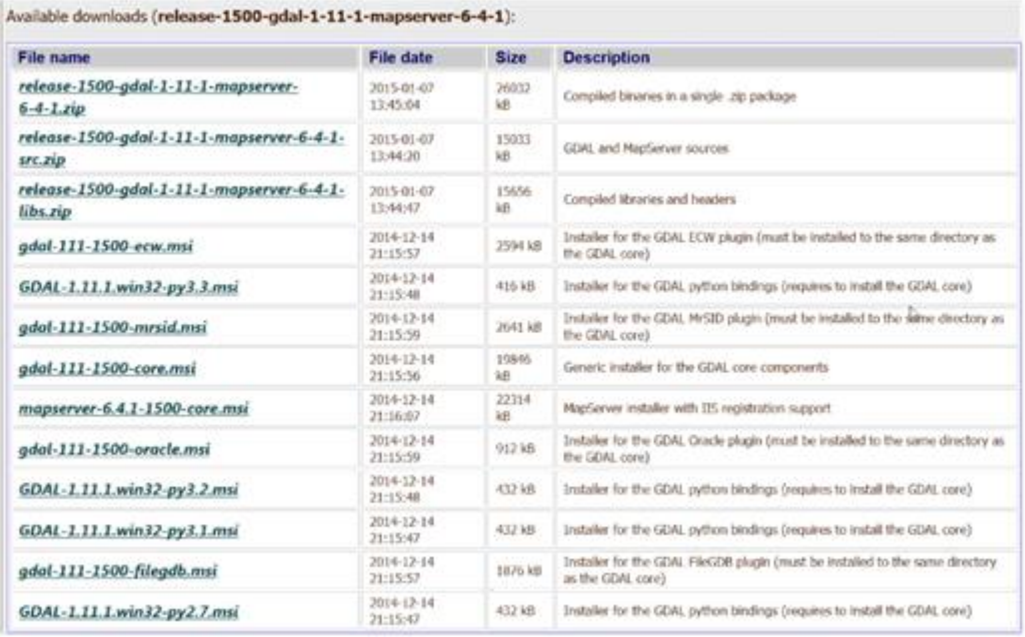
Note: MSC v.1500 may differ if you are using a different Python installation, if it does then please make a note of that number. Note, if you installed the 64-bit version of Python, for the rest of the tutorial please remove the (x86) from the paths.

we are using the MSC v.1500 on a 32-bit system, the picture below illustrates how to match the version with your own python version. The blue highlight is where you should look for either 64-bit or 32-bit systems, and the green shows the release-1500 number which should match the number from IDLE in step 4 above.



Compiler	Arch.	Downloads	Package Info	Date	Revisions
MSVC 2005	win32	release-1400-gdal-1-11-1-mapserver-6-4-1	information	2015-01-07 13:45:35	44ac087 r27733
MSVC 2005	x64	release-1400-x64-gdal-1-11-1-mapserver-6-4-1	information	2015-01-07 13:43:56	44ac087 r27733
MSVC 2008	win32	release-1500-gdal-1-11-1-mapserver-6-4-1	information	2015-01-07 13:45:04	44ac087 r27733
MSVC 2008	x64	release-1500-x64-gdal-1-11-1-mapserver-6-4-1	information	2015-01-07 13:42:35	44ac087 r27733
MSVC 2010	win32	release-1600-gdal-1-11-1-mapserver-6-4-1	information	2015-01-07 13:44:41	44ac087 r27733
MSVC 2010	x64	release-1600-x64-gdal-1-11-1-mapserver-6-4-1	information	2015-01-07 13:46:02	44ac087 r27733
MSVC 2012*	win32	release-1700-gdal-1-11-1-mapserver-6-4-1	information	2015-01-18 20:19:56	0efde12 r28330
MSVC 2012*	x64	release-1700-x64-gdal-1-11-1-mapserver-6-4-1	information	2015-01-18 20:16:55	0efde12 r28330
MSVC 2013*	win32	release-1800-gdal-1-11-1-mapserver-6-4-1	information	2015-01-18 20:12:11	0efde12 r28330
MSVC 2013*	x64	release-1800-x64-gdal-1-11-1-mapserver-6-4-1	information	2015-01-18 20:08:43	0efde12 r28330

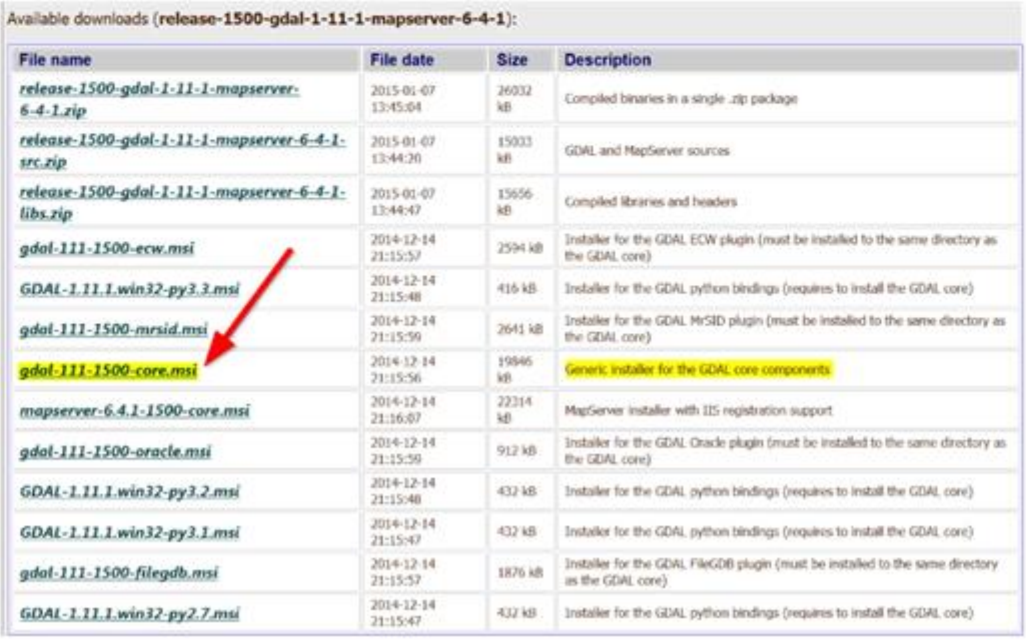
Clicking the link will take you to the list of binaries (installers) to download



Available downloads (**release-1500-gdal-1-11-1-mapserver-6-4-1**):

File name	File date	Size	Description
release-1500-gdal-1-11-1-mapserver-6-4-1.zip	2015-01-07 13:45:04	26032 kB	Compiled binaries in a single .zip package
release-1500-gdal-1-11-1-mapserver-6-4-1-src.zip	2015-01-07 13:44:20	15033 kB	GDAL and MapServer sources
release-1500-gdal-1-11-1-mapserver-6-4-1-libs.zip	2015-01-07 13:44:47	15656 kB	Compiled libraries and headers
gdal-111-1500-ecw.msi	2014-12-14 21:15:57	2594 kB	Installer for the GDAL ECW plugin (must be installed to the same directory as the GDAL core)
GDAL-1.11.1.win32-py3.3.msi	2014-12-14 21:15:48	416 kB	Installer for the GDAL python bindings (requires to install the GDAL core)
gdal-111-1500-mrsid.msi	2014-12-14 21:15:59	2641 kB	Installer for the GDAL MrSID plugin (must be installed to the same directory as the GDAL core)
gdal-111-1500-core.msi	2014-12-14 21:15:56	19846 kB	Generic installer for the GDAL core components
mapserver-6.4.1-1500-core.msi	2014-12-14 21:16:07	22314 kB	MapServer installer with IIS registration support
gdal-111-1500-oracle.msi	2014-12-14 21:15:59	912 kB	Installer for the GDAL Oracle plugin (must be installed to the same directory as the GDAL core)
GDAL-1.11.1.win32-py3.2.msi	2014-12-14 21:15:48	432 kB	Installer for the GDAL python bindings (requires to install the GDAL core)
GDAL-1.11.1.win32-py3.1.msi	2014-12-14 21:15:47	432 kB	Installer for the GDAL python bindings (requires to install the GDAL core)
gdal-111-1500-filegdb.msi	2014-12-14 21:15:57	1876 kB	Installer for the GDAL FileGDB plugin (must be installed to the same directory as the GDAL core)
GDAL-1.11.1.win32-py2.7.msi	2014-12-14 21:15:47	432 kB	Installer for the GDAL python bindings (requires to install the GDAL core)

Locate the “core” installer, which has most of the components for GDAL.



Available downloads (**release-1500-gdal-1-11-1-mapserver-6-4-1**):

File name	File date	Size	Description
release-1500-gdal-1-11-1-mapserver-6-4-1.zip	2015-01-07 13:45:04	26032 kB	Compiled binaries in a single .zip package
release-1500-gdal-1-11-1-mapserver-6-4-1-src.zip	2015-01-07 13:44:20	15033 kB	GDAL and MapServer sources
release-1500-gdal-1-11-1-mapserver-6-4-1-libs.zip	2015-01-07 13:44:47	15656 kB	Compiled libraries and headers
gdal-111-1500-ecw.msi	2014-12-14 21:15:57	2594 kB	Installer for the GDAL ECW plugin (must be installed to the same directory as the GDAL core)
GDAL-1.11.1.win32-py3.3.msi	2014-12-14 21:15:48	416 kB	Installer for the GDAL python bindings (requires to install the GDAL core)
gdal-111-1500-mrsid.msi	2014-12-14 21:15:59	2641 kB	Installer for the GDAL MrSID plugin (must be installed to the same directory as the GDAL core)
gdal-111-1500-core.msi	2014-12-14 21:15:56	19846 kB	Generic installer for the GDAL core components
mapserver-6.4.1-1500-core.msi	2014-12-14 21:16:07	22314 kB	MapServer installer with IIS registration support
gdal-111-1500-oracle.msi	2014-12-14 21:15:59	912 kB	Installer for the GDAL Oracle plugin (must be installed to the same directory as the GDAL core)
GDAL-1.11.1.win32-py3.2.msi	2014-12-14 21:15:48	432 kB	Installer for the GDAL python bindings (requires to install the GDAL core)
GDAL-1.11.1.win32-py3.1.msi	2014-12-14 21:15:47	432 kB	Installer for the GDAL python bindings (requires to install the GDAL core)
gdal-111-1500-filegdb.msi	2014-12-14 21:15:57	1876 kB	Installer for the GDAL FileGDB plugin (must be installed to the same directory as the GDAL core)
GDAL-1.11.1.win32-py2.7.msi	2014-12-14 21:15:47	432 kB	Installer for the GDAL python bindings (requires to install the GDAL core)

After downloading your version, install GDAL with standard settings.

Add C:\Program Files\GDAL\gdal-data to PATH environment variable.

Then create a new variable named **GDAL_DATA** and set the value to C:\Program Files\GDAL\gdal-data.

Installing OpenCV and configuring with CMake:

There are two ways to do this, the first one is recommended. Please try to proceed with the first procedure.

Recommended procedure:

Run OpenCV and extract it to **C drive**.

To configure OpenCV for Python, copy file *cv2.pyd* from *"C:\opencv\build\python\2.7\<python architecture installed>\"* to *"C:\Python27\Lib\site-packages\"* and to **C drive**.

Copy contents of *opencv-build* folder to OpenCV's build directory (i.e., *"C:\opencv\build\"*) folder. These are already build files hence, no need to build again.

Add *bin* directory under OpenCV's build directory (i.e., *"C:\opencv\build\bin\"*) to PATH environment variable.

OR,

Alternate procedure:

Run OpenCV and extract it to **C drive**.

To configure OpenCV for Python, copy file *cv2.pyd* from *"C:\opencv\build\python\2.7\<python architecture installed>\"* to *"C:\Python27\Lib\site-packages\"* and to **C drive**.

Install CMake with **default settings** and add it to the PATH environment variable.

Run CMake and select OpenCV's *sources* directory (i.e., *"C:\opencv\sources"*) in the ***source*** section and select OpenCV's *build* directory (i.e., *"C:\opencv\build"*) in the ***build*** section.

Configure and then Generate.

Add *bin* directory under OpenCV's *build* directory (i.e., *"C:\opencv\build\bin\"*) to PATH environment variable.

For Language Translation:

First we need to create a .ts file for all the .ui file and with that .ts file we can create .qm script file which contain the translation script using QtLinguistics.

For qt:

Open cmd where .pro file is present and then run the following command:

```
lupdate –pro <pro file name>.pro -ts <language name>.ts
```

using QtLinguistics open .ts file and then translation for each word in the respective language.

After translating all the word release the file it will create .qm file script.

For python:

Create a pro file which contain all python .ui files

Open cmd where .pro file is present and then run the following command:

```
pylupdate4 –pro <pro file name>.pro -ts <language name>.ts
```

pylupdate4 is used as we are using pyqt4.

The Dharohar software can be executed from the main.py file inside the untitled folder then the process will continue.

All the python files are inside the source folder.

Without Gdal saving to database will not happen. Proper installation of postgreSQL, Postgis and Gdal is required for the database process.