Article

By Aleksey Borodulin and Aleksandr Kleshnin

Since the mid 2000s open GIS has gradually been attracting the attention of businesses and government agencies around the world. Nevertheless, the introduction of open GIS in business processes is hampered by the fact that at the moment there is no ready-made method of implementation for open GIS solutions. In this article we want to share the experience of our implementation and the use of open GIS software at Intetics.

Open Source GIS Software

User Experiences at Intetics

Intetics employees

The idea of using open software at Intetics was first aired in May 2011. At that time the main working tools for Intetics' specialists was proprietary software provided by the client. The volume of manual vectorization projects was increasing along with the growing number of employees. We planned to hire approximately 50 new employees; though the time spent using GIS software did not exceed 5-16% for a single employee. As an alternative to purchasing extra proprietary licenses, a decision to use open software was made.

Briefly about the project

The client provided us with shp-files containing road graph and building geometry. In the course of the project, we had to clarify the client's road graph and buildings contours spatial positions data. The first step was to clarify existing geometry, and then add the new one. After completing this, we followed the same procedure with the building polygons.

Choosing and implementing software

After careful evaluation we chose gvSIG and Quantum GIS, mainly because these products are widely available as desktop GIS software. Then we completed several test projects using both gvSIG and QGIS and compared the results and the expert reviews. Generally, gvSIG performed better, although some tools were less user-friendly. Among QGIS strengths are nice and logical GUI, multiframe support, high pace of development, an active users' and developers' community, lots of plug-ins and rather good documentation. Weaknesses include slow data panning, slow raster rendering and overall low performance and instability.

Though QGIS performance was not so good, it was finally chosen as GUI as ease of mastering played a very important role for us. Due to the specific features of the project, software performance did not influence the overall work flow much. At the same time, software simplicity and GUI consistency is very important during the first exposure to open GIS software, as GUI ease minimizes users' discomfort.

The difficulties with Quantum GIS began with the lack of network software installation tools. Initially users had to install and configure the software all by themselves. It was time consuming and initiated a torrent of questions from users. Beginning with the 1.7.3 release we started using the portable QGIS version as its configuration was set and all the necessary plug-ins loaded by default. The portable version is distributed by copying.

The 1.7.0, 1.7.3, 1.7.4, 1.8.0 QGIS releases were used in the project. Whilst preparing for the project, the new releases were tested and a changes analysis made. The decision to use it or not was made based on the data collected; evaluation of all the advantages and disadvantages of new releases compared to the current version. Assuming a positive decision, we performed pilot works using the new release of the software.

Issues that appeared during QGIS utilization and new version releases:

- shp-files merge didn't work in 1.7.4 (fixed in 1.8.0);
- snapping to segment didn't work in 1.7.4 (fixed in 1.8.0);
- Cyrillic symbols weren't displayed in attribute table in 1.8.0. We imported layers which contained Cyrillic attributes in DB using QGIS version 1.7.4.

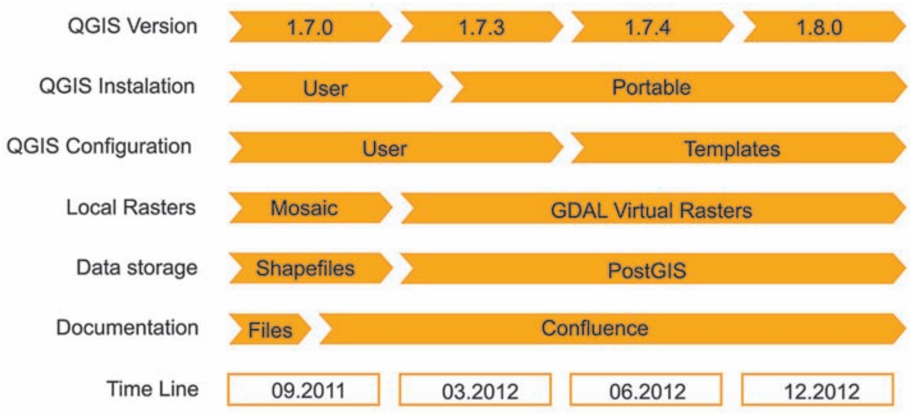
According to the requirements specification, the road graph should not have geometry gaps and overlaps. During the project our specialists mainly worked on separate distanced territories and the amount of geometry situated on the borders of the neighboring working areas was minimal. Vectoring data was stored as shp-files on the network drive. When the editing was completed, data from separate shp-files were combined into one, common geometry on borders was checked and manual editing was performed if needed.

As the project developed, we began vectorization of large urbanized territories and faced the problem of users' interaction in working areas which had common geometry. The solution was to create a common information space using DB.

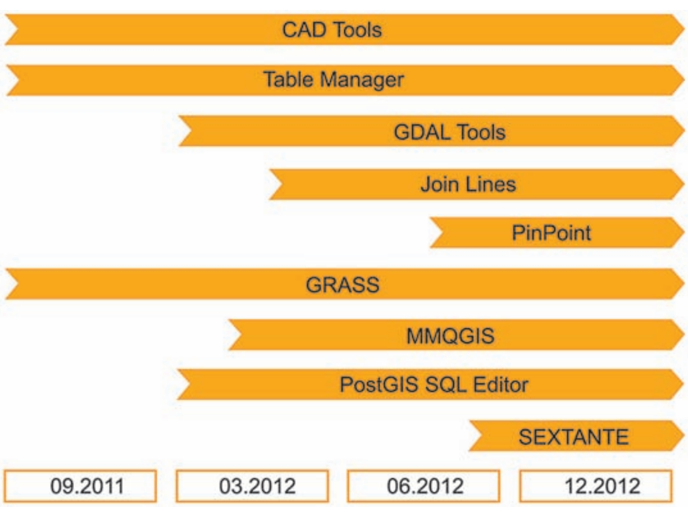
Initially, we considered DB implementation using SQLite/SpatiaLite or PostgreSQL/PostGIS. Some tests were done and we discovered SpatiaLite performance is significantly reduced when used by many users simultaneously; this was the reason for declining it. PostGIS performance met our requirements completely.

During the preparation process, database engineer imports pre-processed source data to the DB and editing rights were adjusted using pgAdmin application and Database Manager Plug-in. PostGIS allowed us to coordinate the work on borderline areas and perform quality control rapidly. It allowed us to evaluate current progress in particular working areas and the overall project progress (SQL queries on the number of polygonal objects and the sum of road graph edges lengths), derive performance index, perform statistics calculation and stream monitoring (project manager is able to see any work area any time). The use of SQL queries allowed us to perform DB layers merge and copying and create intersection layer between two layers. Creating DB layer copies using SQL queries and DB layer data unload to local drive using OGR library is also possible.

In the first stages of DB usage we performed vectorization for several large and geographically distanced working areas. Two road layers were created in the DB – for the new and amended geometry. Problems in the QGIS work appeared when a large number of objects were created in one layer (more than 60 thousand for a linear layer), for example, long data loading, delays in panning, layer attributes table opening, object selection, operation performance using field calculator, etc. We decided, therefore, to create several layer pairs for every working area instead of using two layers for all the areas. This meant that the number of records for every table in the DB remained supportable even at the end of the process and did not cause loss of efficiency.



Implementation process



QGIS Plugins

Tool suite

Our people

For Quantum GIS there is a fairly complete manual in Russian (translation done by gis-lab.info community). However, for staff training on QGIS, an internal guide was prepared. This described basic functionality and the processes of typical project tasks. A users' and developer's community support and official documentation were used for the internal guide. We want to express thanks to all who supported us and special thanks to the GIS-Lab community. Written instructions were tested by experts and were amended in the process as a result of experience gained.

Due to the peculiarities of the technological process, vectorization (activity performed with the help of QGIS) is done at Intetics every quarter. The time in between was used for analysis and process changes introduction.

Implementation and maintenance of the new process was lead by the leading experts (LE). LE duties included: analysis of the project experience, working instructions development and improvement, helping specialists in mastering software and the whole process and solving any arising problems.

All the working documentation was uploaded to the team collaboration software, Confluence. LE edit and add documentation along with work instructions to the Confluence. Employees use Confluence to ask for document clarification, to add information (improvement suggestions, tips on dealing with issues) and share experience. Changes are reviewed and moderated by LE. Confluence enables the sharing experience online and minimizes the number of meetings.

At first we faced employees' resistance to the innovation, which manifested itself as inaction. Some employees were not motivated enough to master new, sometimes unstable, software. Often employees compared QGIS and commercial GIS functionality (in favour of the latter). The process actively developed, new tools and instructions were introduced, but a number of employees continued working "as before." These difficulties were not caused by specific tasks or software, however; all engineering companies face the same problems in the process of new working tools implementation. In order to solve such problems, each quarter and prior to the project start, a list of innovations is created and e-mailed to the employees. Additionally, meetings are held allowing leading experts to explain the peculiarities of the project. A separate section in the project info DB is created to share experience with open source software, both positive and negative (such tips often prevent common mistakes being made).

Our internal QGIS training is developed for employees who have at least some experience with GIS software and with real-life projects. Normally, with some experience with GIS software, employees master QGIS to an adequate level in 1-2 hours. This was achieved by assigning a mentor to every junior developer. Subsequently, the assimilation process for new employees was optimized.

Technical and organizational specifics of the project

The preparation process included initial WMS-data quality check. If, for some reason WMS could not be used (slow data loading, bad quality of the image), then pre-loaded satellite images in the form of tiles were used. Virtual rasters are created out of tiles, using module GDAL.

To reduce time spent by the user on provisioning, qgs-project templates were used. Templates were created at the preparation stage. The template contains all the necessary layers from the PostgreSQL/PostGIS DB (shp-files were supplied together with templates before the DB was created), WMS layer added, styles/group views, coordinate systems, snapping options and options for displaying and editing attribute data configuration set. Templates are distributed by copying via the network.

In the first stages of the project the topology errors correction was performed with the QGIS GRASS module (snapping of the line vertex in the nodes inside the specified tolerance, lines breaking upon intersections, removing pseudo-nodes). We also tried the topology check using gvSIG, but found it both inconvenient and time consuming. At the end of the day we decided to do all the required checks with GRASS, as it gave us output acceptable to our customer. The following was done: search for lines not snapped (dangled nodes), removing pseudo-nodes, lines self-intersection correction and removing self-overlapped geometry.

JIRA is a tracking system our company uses for task tracking and time management. Territory polygons marked by the client are divided into small-sized working areas. For each area a record in JIRA is created by PM. Every record contains the following information: the name of the region, estimated time required for vectorization, the deadline and name of the employee assigned. JIRA allows ongoing monitoring daily (status, readiness percentage and the time spent) and statistic gathering.

Conclusion

We successfully implemented the execution of large-scale works using open source software combined with the proprietary software. We now have a well-established documented production process.

We plan to develop this process by implementing the following steps:

- Develop an efficient bug reporting process;
- Implement topology processing using PostGIS;
- Create local service of raster data on the Geoserver basis;
- Finalize the transition process to newer software releases.

Intetic's example shows the real possibility of successful open source GIS software use for large scale projects. Thanks to very detailed process description and precisely defined team roles, the project was completed on time and with the quality level required by customer.

Aleksey Borodulin and Aleksandr Kleshnin are Intetics GIS Analysts with specialization in open source and multiuser database solutions. For more information please visit www.intetics.com/geo or send an e-mail to geo@intetics.com.