

# GeKon – Applying Novel Approaches to GIS Development

Tomáš Richta

Department of Computer Science, FEE, CTU - Czech Technical University in Prague,  
Karlovo nám. 13, 708 33, 121 35 Prague 2, Czech Republic  
richtt1@fel.cvut.cz

**Abstract.** This paper describes a few ideas concerned with geographical information systems (GIS) development. Those ideas come from a GIS development project named GeKon, which is now held on the Department of Computer Science at CTU. The first of them deals with a huge semantic gap between the complex structure of real world and its representation in GIS. Some papers describe this problem as a consequence of wide usage of procedural programming languages and relational database management systems in contemporary GIS development. The object-oriented approach is usually recommended as a better way of constructing such systems. We demonstrate here our findings in object modelling, programming and data management achieved in GeKon project. The other idea shows the fact, that not always the term object-oriented is understood in the same way. People usually have in mind some specific programming language and its structures, instead of the real problem and its solution. In this paper we want to clarify those misleadings and try to describe requirements, that should be fulfilled to achieve expected benefits of this approach. We also introduce next supposed steps in the GeKon project.

**Keywords:** GIS, semantic gap, object-orientation, data modelling, Smalltalk, OODBMS

## 1 Introduction

At the beginning, we have to slightly describe geographical information systems (GIS) evolution process to reader be more familiar with questions connected with this area of research. Present GIS were principally constructed as tools for computer based maintenance of geo-referenced data formerly kept in cartographical tools, like maps, atlases, cadastral plans, etc. Evolution of such systems probably started with the idea of cartography digitalization drawn ahead by the vision of map producing acceleration, and also faster access to cartographical data, including sorting, searching, and other functions that could lead to better utilization of geo-referenced data.

## 1.1 Present GIS data management

Present problems of GIS systems have probably their origin in the way of GIS data production. The process of cartography digitalization consists of scanning cartographical tool, and then its vectorization by laying vector paths over the lines in the map. The data produced by this process tend to come under three main geometry types:

- spotted, when some interesting spot had been captured, like a church, a tree, a well, etc.
- linear, when some linear data had been captured, like roads, borders, railways, networks, etc.
- planar, where some area of interest had been captured, like cadastre area, city district, etc.

This separation lead to three main data types used in present GIS: points, lines and polygons. Almost all such systems use as its storage device those three elements. One of the first companies in GIS field named ESRI had invented special file format named shapefile (SHP) for storing these data. SHP becomes something like a standard in GIS data management. Probably for better preservation of cartographical information, data were collected separately in thematic parts. It means, that all churches in observed area were vectorized in one point-based shapefile, all roads in one line-based shapefile, etc. This approach lead to very discrete representation of real world. When the map is to be retrospectively reconstructed in GIS, all thematic files has to be layered over themselves to obtain proper cartographical representation. Layers are not connected among themselves, so ie. the information about roads leading to a well is not present.

Further needs of additional information capturing and storage lead to extension of shape format with so called attribute data component like names, numbers, etc. At the same time as this problem arose, the relational approach to data management achieved great concern. So the GIS files were extended with database file (DBF) including attribute data, connected to their geometry representation in shapefiles. Both those files were stored separately. Relational tables consist of domains, and records, so the real world information has to be separated and simplified into these parts to be stored in DBF. This approach also lead to more discrete GIS data quality and omitting of more complicated information.

Both stated approaches have not changed since the beginning of GIS. The only novel approach is to store SHP and DBF files into relational database management system and reach them through the SQL. But this way only leads to bigger conservation of the problem.

## 1.2 Semantic gap

Wikipedia describes semantic gap as a mismatch caused by some conflicts emerging in layered systems, when a high level of abstraction need to be translated

to lower, more concrete artifacts [1]. When we try to imagine our modern informational world as a very complex layered system, we see the real world structure, semantics, and topology on one side and its computer based representation in information systems on the other side. Between them, there are multiple sensors, but not only mechanic ones, but also human senses as a main set of tools used in present digitalization process. Rapant defines the term of geoinformatics as a scientific discipline dealing with quality, behavior, and reciprocal interactions of spatial objects, phenomena, and processes by force of their digital models combined with information technologies [2]. When a GIS may be able to serve as a tool for better understanding of the real world, its way of storing real data must be as natural as possible. But present GIS with their fragmented data model, layered representation, and relational simplicity are not able to provide it. For example, when we want to describe a village, we need to prepare a number of layers covering all fields of our interest, ie. buildings, roads, fields, gardens, forests, sewages, electrical wiring, gas pipelines, churches, wells, pubs, shops, bus stations, etc. For each theme, we need a relational table blueprint that describes whether there are points, lines, or polygons, and what attributes we want to store along with geometry. This leads to strenuous process of uniformization of gathered information and quanta of discrete data production. All houses must be described by the same group of attributes, and the information whether house is a pub is stored somewhere else, than the number of its stories. Also when we want to combine together villages belonging to some administrative area, we need to flatten out all the villages thematic layers to be coherent and thus comparable together. This process isn't very natural and its results do not exactly cover described original. And that is why the semantic gap emerges.

## 2 Object-oriented approach

Merunka recommends an object-oriented approach (OOA) as a good solution of semantic gap in information systems development [3]. He states that OOA solves this problem by introducing the idea of higher level of abstraction in software development with emphasis on modularity, reusability, and standardization. He also summarizes main criteria of object-oriented system:

1. Data and its functionality are encapsulated in one logical entity, which is called object.
2. Objects communicate by sending messages to each other.
3. Objects are able to inherit their properties from other objects.
4. Objects are collected in classes together with other similar objects.
5. Different objects are able to react to the same message, which is called polymorphism.
6. Objects could also have other relationships like composing, dependency, or delegation.
7. Methods are programs consisting of operations over object data. Methods are components of objects.

## 8. Object identity is independent on object data.

Comparing OOA ideas with GIS problems gives a new horizon to their solution. Our natural way of perception of material aspect of the real world is that everything is an object, and now with OOA everything is possible to be an object in our computer memory. What we only need is to start using OOA in instant. No more thinking about geometrical representation of objects or the way how to store them into tables.

Of course, this is not the novel idea. Mitrovic and Djordjevic-Kajan described in 1996 OOA as natural paradigm for highly complex domains, especially because it maintains a direct correspondence between real world and application objects [4]. Kofler in 1998 in his PhD. thesis about large 3D GIS databases recommended OOA as an obvious choice for any new GIS [5]. Chance et al., key developers of GE Smallworld GIS, describe OOA as highly effective when applied to the requirements of GIS [6]. The question may arise, why nothing has changed yet, even if those articles had already been written within past ten years.

### 2.1 Object-oriented modelling

Looking for some indications of OOA emerging in GIS, we could find a few papers describing object modelling in GIS, especially when the 3rd dimension has to be introduced to those systems. For example, Nebiker wrote in 2003 about his fully object-oriented model for 3D geo-objects [7]. Also Kolbe and Groeger work on their unified standard for 3D city models that is strictly based on OOA [8].

### 2.2 GIS development projects

There are also some projects, that have already implemented parts of GIS using OOA. One of them is already cited in Kofler's PhD. thesis [5] where he described a few tests of GIS database implemented over two examples of object-oriented database management systems - ObjectStore and O2. Kofler used as a platform SGI Indigo workstations with MIPS R4400 CPU at 250 MHz and 128 Mbyte main memory, which is not very strong but it is adequate to the year of publishing. He states that OODBMS has bigger performance demands than traditional file systems, but still recommends them as a best solution for GIS database.

Balovnev et al. implemented software for 3D/4D geo-scientific applications development named GeoToolKit [9]. They used C++ and ObjectStore as core technology tools. GeoToolKit is a class library for the storage and retrieval of spatial objects within an object-oriented database. Developers state that their approach lead to separation of focus on the geoapplication semantics and the need of spatial objects assembly from multiple relational tables. They also mention the reduction of the code written, improvement of its understandability, and they describe some interesting applications implemented with the use of GeoToolKit library.

One of the other projects is GeoViewer implemented by Lurie et al. in 1997 [10]. Its an object-oriented GIS framework with optimized spatial geometry representation providing transparent linkage to data objects. GeoViewer is written in Smalltalk with small amount coded in C. Minimal documented configuration for running GeoViewer is Sun Sparc 10 with 64MB of RAM or Pentium 166 with 64MB of RAM. This project is remarkable because it incorporates our main ideas about design of GIS, such modelling natural relationships between the geodata, or independent representation of objects and its geometric representation. Taking the year of publication into account, the measure of innovation ideas is stunning. Also the question arises, why this is not the main approach in nowadays GIS.

### 2.3 GeKon project

Because we see all of those projects very important as a way to improve present GIS applications, we have also opened research project concerning OOA used in GIS development. The name of this project is GeKon an it was started as a semestral work in one of our Software engineering courses. Members of the development team were three undergraduate students - Ivo Kondapaneni, Petr Novosad, Jiri Verunek, and author himself. Ivo Kondapaneni had to leave our project later for his other academic duties, so project extent had to be partially narrowed.

GeKon system is now able to load data from shapefiles and display it on the computer screen. It is also possible to zoom displayed data and move over it. As a test case the city GIS problem was chosen and as a development platform we used Squeak Smalltalk dialect. In Fig. 1. we introduce our object model.

This model shows the GeKon structure - space subdivision, geometry and city object model. In the space subdivision part we formed our idea that it was necessary to maintain some sophisticated indexed structure that allowed fast searching. We use two trees here. First one is responsible for logical subdivision of the city (city districts, basic settlement units, and counting districts) and it is implemented with collections. Second one is responsible for searching the physical space within the counting district and it is implemented as a R-tree. Loaded objects are primarily classified according to the logical space subdivision and then inserted into R-tree of relevant counting district.

Geometry part of our model is responsible only for the geometric representation of objects. Now it contains only necessary representations - point, line and polygon, expanded with 3D solid representations. Each shape has its own bounding box which is stored in space subdivision structures.

The logical object structure of city is the only demonstrative example of part of the city configuration. Its main purpose is to show how we need to model the reality - naturally. We expect that in the future this part of the GIS will be constructed in some visual metamodeler directly by the GIS user at the moment of data import. This approach assumes strict clear disjunction of the data and its representation.

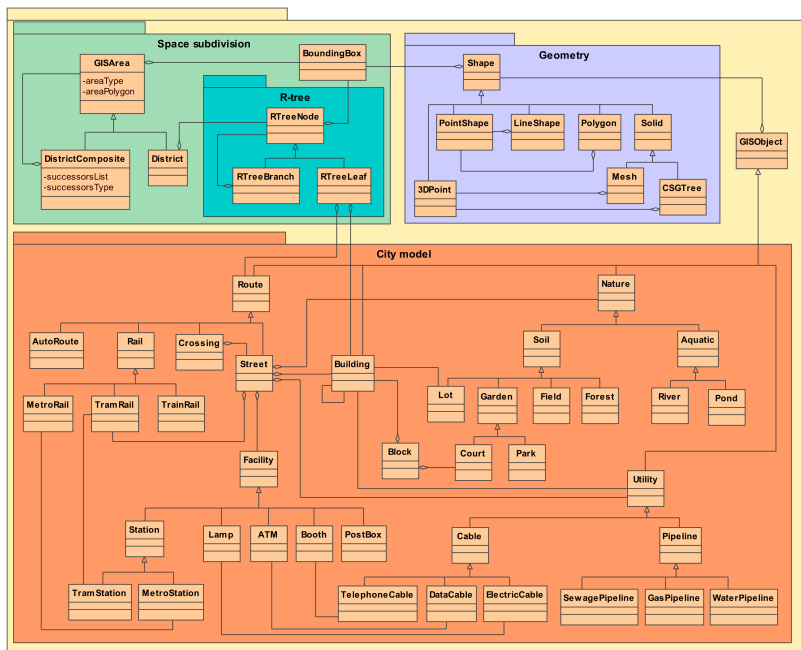


Fig. 1. GeKon object model

Visualization was partly implemented using OpenGL, but after loss of I. Konapaneni we had to choose easier way. So we used Squeak morphic visualization system, which is sufficient, but lacks possibilities to improve graphics performance in the future like redrawing using textures etc. So we plan to redevelop OpenGL rendering engine for GeKon in future.

Data management has not been covered yet, we are storing all data in Smalltalk image. In future, we plan to incorporate OmniBase as a database management system, which ensures native storage and retrieval of objects. We also plan to cooperate with another project held on our department named Cell-Store, which deals with heterogeneous data storage and now serves as native XML database.

During the development we used sample data from the ICIP (Institute of City Informatics - Prague), covering the Josefov city district. The average loading time was 1749ms per 1MB and the refreshing time about 320ms, which is not bad. But we see these results orientational only, because the sample is too small to give evidence of the GeKon capabilities. Further we want to use bigger data collections for testing, but they were not available at the moment.

### 3 Conclusion and future work

In this paper we described semantic gap problem between the real world and its computer based representation. This problem was described from the GIS point of view. GIS are still very tightly coupled with the old fashioned representation of the real world by using files and relational tables. We discussed here OOA as one of the approaches that are able to overcome semantic gap. Some interesting projects concerning object modelling and object-oriented implementations of GIS systems were slightly introduced. In the end of our paper, we also presented our own project GeKon, in which we developed the prototype of object-oriented GIS. Now we want to summarize discovered requirements for the GIS development and also planned work in the GeKon project.

#### 3.1 GIS development requirements

Based on our experience in GeKon system development and also on the knowledge from previously cited papers we strongly recommend following rules to be applied in OOA GIS development:

- Separate geometric representation from object itself to be independent on its shape.
- Use R-trees or other indexing structures for fast searching in space and efficient data retrieval.
- Pay attention to very fast rendering algorithms to prepare the most comfortable environment for users.
- Use metamodeler controlled by the user to obtain the logical structure of the place of interest.
- Use strictly pure non-hybrid object-oriented language to avoid programmers cheating (for details see [3]).

#### 3.2 Further steps in GeKon project

In future, we want to continue in development of GeKon system on our department, partly in the form of dissertation, diploma and other thesis, partly as student semestral projects. Fig. 2. describes the roadmap of planned work.

### References

1. Wikipedia, the free encyclopedia, *encyclopedia*, [www.wikipedia.org](http://www.wikipedia.org).
2. Rapant P.: *Zaklady geoinformatiky I. (Geoinformatics fundamentals I.), course lecture*, Ostrava, The Czech Republic, 2005.
3. Merunka V.: *Objektove orientovany pristup k projektovani informacnich systemu (Object oriented approach to information systems development), habilitation thesis*, Department of Information Engineering, CUA in Prague, The Czech Republic, 2005.

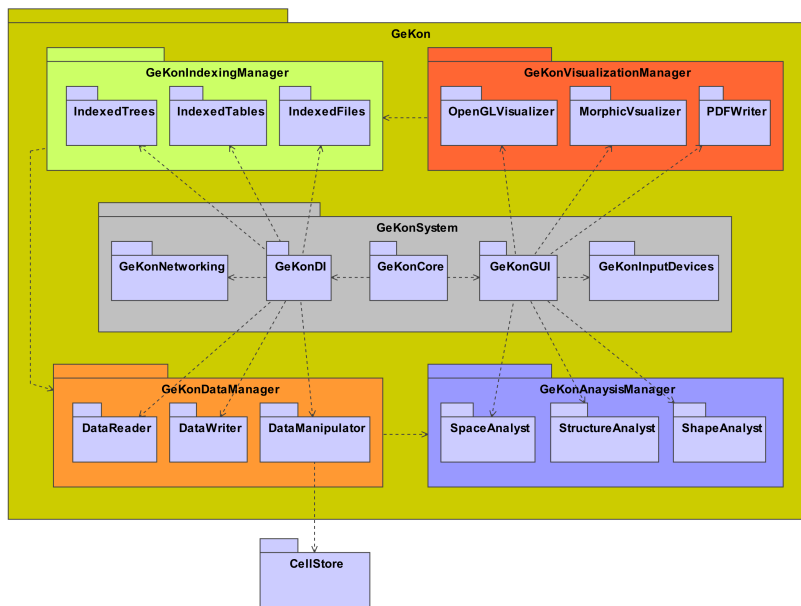


Fig. 2. GeKon structure model

4. Mitrovic A., Djordjevic-Kajan S.: OO paradigm meets GIS: a new era in spatial data management, *invited paper*, presented at YUGIS'96, Belgrade, Yugoslavia, 1996, <http://www.cosc.canterbury.ac.nz/tanja.mitrovic/>
5. Kofler M.: R-trees for Visualizing and Organizing large 3D GIS Databases, *PhD. Thesis*, TU Graz, Austria, 1998.
6. Chance A., Newell R.G., Theriault D.G.: Smalworld GIS: An Object-Oriented GIS - Issues and Solutions, *Smallworld GIS white paper*, 2000, <http://www.logis.ro/downloads/>
7. Nebiker S.: Support for visualization and animation in scalable 3D GIS environment - motivation, concepts and implementation, *scientific paper*, International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXIV-5/W10, 2003.
8. Kolbe T.H., Groeger G.: Towards unified 3D city models, *article in: Proceeding of the ISPRS Comm. IV Joint Workshop on "Challenges in Geospatial Analysis, Integration and Visualization II"*, Stuttgart, 2003.
9. Balovnev O., Breunig M., Cremers A.B., Shumilov S.: GeoToolKit: Opening the access to object-oriented geo-data, *scientific paper*, Interoperating Geographic Information Systems, Boston: Kluwer Academic Publishers, 1999.
10. Lurie G.R., Korp P.A., Christiansen J.H.: A Smalltalk-based Extension to Traditional Geographic Information Systems, *students paper*, <http://www.dis.anl.gov/geoviewer/>, 1997