Applying Analysis Patterns in the GIS Domain

Jugurta Lisboa F.[1,2], Cirano Iochpe[1] & Kate Beard[3]

[1]Instituto de Informática
Universidade Federal do Rio Grande do Sul. Porto Alegre, Brazil
Phone: +55 51 316-6842
Fax: +55 51 319-1576
Email: {jugurta, iochpe}@inf.ufrgs.br

[2]Departamento de Informática
Universidade Federal de Viçosa. Viçosa, Brazil
Email: jugurta@dpi.ufv.br

[3]NCGIA - Department of Spatial Information Science and Engineering
University of Maine. Orono, USA
Email: beard@spatial.maine.edu

Presented at the 10th Colloquium of the Spatial Information Research Centre, University of Otago, New Zealand, 16-19 November, 1998

Abstract

Patterns have been used in software engineering to enable the reuse of successful solutions for recurrent problems in various steps of the software development process. In the development of geographic information systems (GIS), usually only geographic data are reused. This paper shows the adequacy of deriving as well as applying analysis patterns in the conceptual data modeling process of geographic applications. By means of analysis patterns, already tested and validated solutions for recurrent problems can be reapplied to different GIS projects. The requirement of both a suitable pattern definition formalism and a structure for a GIS pattern catalogue are also discussed. Finally, an example of an analysis pattern for the GIS domain is presented.

Keywords and phrases: analysis patterns, conceptual data modeling, GIS applications.

1. Introduction

Geographic Information Systems are used to support database applications that both store and analyze georeferenced data (e.g., environmental control, utilities network). Conceptual data modeling in the domain of GIS applications typically identifies a large number of geographic entities as well as relationships between them. According to Gordillo et al (1997), expert GIS designers do not solve every problem from scratch. On the contrary, they usually reuse previously worked out solutions to solve similar problems. Since the acquired experience remains with the designer him/herself and is usually not made public, other designers are not able to share or reuse that knowledge.

On the other hand, the exchange of geospatial data through the Internet among different users or organizations increases constantly. Standardized metadata describing sets of geospatial data are stored in clearinghouses (Nebert, 1996). In such a way, a GIS designer can find, assess, and obtain available data sets, avoiding having to produce them once again. However, the design of the GIS database must still be done from scratch, since nowadays no metadata for model description is available. Currently, the most well known system of clearinghouses is supported by the Federal Geographic Data Committee (FGDC, 1997).

Patterns express a new concept which software engineers have been using to both document and share their experiences in object-oriented system analysis and design. There are different kinds of patterns that are used at different levels of abstraction as well as in different phases of the software development process. Though the use of patterns to enhance GIS database design has not yet been investigated in depth. Balaguer et al (1997) describe one of the rare examples, where the use of design patterns is considered in the development of conventional applications with geospatial features. In the present paper, we discuss the suitability of both deriving and applying analysis patterns to respectively describe and reuse experiences of GIS conceptual data modeling.

The idea of reuse has already been explored in the GIS applications domain, but not in the data modeling process. Marr et al (1997) proposed a structure of spatial process modeling that can be reused and was made available through the Web. We think that the reuse of spatial process modeling is complementary to our proposal, supposing that a spatial process could be defined as a functional pattern and could be integrated with the analysis patterns and the spatial metadata in a clearinghouse system.

We believe that the integration of both a GIS patterns catalogue (including all already existing pattern categories) and a set of spatial metadata into a clearinghouse will become important issues of research in the next years along with the increase of geographic data sharing through the Internet. Results of this investigation shall contribute to the development of higher quality GIS applications.

The remainder of this article is organized as follows. Section 2 briefly introduces today's existing pattern categories. Section 3 presents, in more details, the concept of analysis patterns. In Section 4, we emphasize the advantages of using analysis patterns in the process of modeling GIS applications. Section 5 describes expected features of a patterns catalogue for the GIS domain. Conclusions and suggestions of future work are presented in Section 6.

2. Pattern Categories

The pattern approach is still very recent in computer science. Thus, there does not exist a consensus on a definition for the word "pattern" in this context. The use of patterns is directly related with the idea of reuse in software engineering and has origins in the work of Christopher Alexander, an architect that, in *(Alexander et al., 1977)*, defined:

> *"Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice. (Page x)"*

The ideas of Alexander had given origin to the development of the research on design patterns and, later, several other pattern types. Fowler (1997) defines a pattern as:

> *"An idea that has been useful in one practical context and will probably be useful in others. (Page xv)"*

According to Gamma et al (1994):

> *"A pattern presents the essence of the solution for a recurring problem in a specific context. (Page 2-3)"*

The pattern approach became more popular after the book *Design patterns: elements of reusable object-oriented software* (Gamma et al., 1994) that presents a catalogue of solutions for recurrent problems in object-oriented design. Gamma's definition comprehends the fundamental ideas of a pattern. The expression *"the solution for a problem"* means that each pattern identifies one problem and presents one solution for it. The term *"essence of the solution"* means that only the essential elements are described, leaving specific aspects to be detailed by the designer, since specific aspects of solutions are usually not reusable. *"recurring problem"* means that patterns should be created for problems that already occurred some times and will occur again. *"in a specific context"* means that the solution is valid for a particular context.

According to Buschmann et al (1995), patterns can be divided in three categories: *architectural patterns*; *design patterns*; and *idioms*, the latter are also known as *implementation patterns* (Pree, 1995). The first two categories include patterns related to the system's design process. Though, they differ from one another in the level of detail where each one is applied. Design patterns are more abstract (and of smaller granularity) than architectural patterns, while idioms are used in the level of programming languages.

A fourth category of patterns, introduced by Fowler (1996), is the one of *analysis* or *conceptual patterns*, according Riehle and Züllichoven (1996). This type of patterns is used to describe solutions related to problems that arise during both the requirement analysis and the conceptual data modeling phases. Analysis patterns reflect rather conceptual structures of the application domain and are not concerned with computational solutions.

Most of the analysis patterns that were proposed until now have been designed mainly to solve problems that are recurrent in the conventional applications domain (Hay, 1995; Coad, 1997; Fowler, 1997). However, the idea of analysis patterns can be applied to improve quality and productivity in the development of nonconventional applications (e.g. GIS applications) as well.

All pattern categories cited above can be used in a combined way. For example, analysis patterns can be used to register how designers from an application domain usually model the most important entities and its relationships in that domain. Design patterns can be used, for example, to document how data about the relief of a region are represented through a Digital Terrain Model (DTM). Finally, idioms can define how a DTM should be implemented in each commercial GIS product.

3. Analysis Patterns

Since the pattern approach was introduced in the discipline of software engineering by the community of object-oriented system designers, the category of design patterns became more popular and well known. However, several researchers have been investigating patterns to help system analysts to reuse prior, already tested modeling strategies for recurring problems (Hay, 1995; Gerth et al., 1996; Riehle and Züllighoven, 1996; Coad, 1997; Fowler, 1997).

According to Coad (1992), traditional reuse techniques in the context of object-oriented analysis rely upon the inheritance mechanism. Thus, to reuse predefined classes, analysts need to know where to find the existing classes and then derive new subclasses from them. These existing classes are used as simple building blocks in a new application design.

An analysis pattern, from now on just pattern, describes a set of classes, possibly of different class hierarchies, and the associations among them[1]. Patterns can be seen, therefore, as a way of describing more complex design schemes, which are recurrent in the modeling process of many (similar) applications. Thus, they can speed up considerably the design of new applications and increase the quality of that design (Hay, 1996).

A pattern presents a solution in a more suggestive than prescriptive way, providing a model and a discussion of why the solution proposed has its actual form, and its strengths and weaknesses. According to Fowler (1996), the really important thing about a pattern is not the model but the rationale behind it.

To illustrate how an analysis pattern can look, an example borrowed from (Hay, 1995) is showed. Hay presents a set of patterns that he called *data model patterns*. These patterns are recurrent in many conventional applications. Among them there is a pattern called *Geographic Locations*, which is shown below and represents a solution to the address-modeling problem. This pattern can be applied to modeling GIS applications as well.

Notice that the example was rewritten according to the object-oriented approach. The pattern description follows the form proposed by (Coplien, 1995), and the diagrams are in UML notation (Rational, 1997).

Data model graphic notations are not powerful enough to support patterns descriptions. The graphic notation only captures the final design result, representing it as a network structure of classes and associations between them. A pattern description, though, also includes the documentation of decisions taken, choices made, identified alternatives, and weaknesses of the model. In order to help designers to take a better pattern understand, examples should be included on the pattern description (Gerth, 1996).

1 The following nomenclature is used in this article. A *geographic class* is an abstraction of a *geographic entity*, which occurs in the reality (e.g. river, road), and an *association* is an abstraction of a *relationship* that occurs between entities of the reality (e.g. a river cross a road).

Pattern name: Geographic Locations

Problem: The Geographic Locations pattern in such a way describes an alternative solution to the problem of address-modeling, being useful both for persons and organizations.

Context: It is common in many real life situations to find one street address related to more than one person (or organization) and vice-versa. The simplest way to model addresses is to include a set of address-related fields (e.g. street, number, ZIP code) as attributes of a class. In this case, a series of known problems can occur as, for example, incomplete information or database inconsistency.

Forces:

- Often it is of interest to collect addresses by neighborhood, city, state, or by other geographic location where the address is located. Thus, each site must be in one geographic location, which means that each geographic location may be the location of one or more sites.
- For mailing labels, an address is a single piece of information. If the purpose of the address is only for mailing labels, the class attributes choice could be considered.

Solution: Figure 1 shows the class diagram of the pattern. The class Party is a generalization of the classes Person and Organization. The class Party has an association called placement (of cardinality m:n) with the class Site. This association can include attributes as period and type of the placement. The class Site should contain the attributes that characterize the address of one Party instance. The GeographicLocation class is defined to create a hierarchic structure of locations (e.g.: Country, State, City, Quarter). In this way, each instance of Site must be associated with only one instance of GeographicLocation.

4. Usage Requirements of Analysis Patterns in Gis Data Conceptual Modeling

The domain of geographic information systems can benefit from the pattern approach, since several sets of geographic entities are recurring in many GIS applications. As an example, we can mention geographic entities that are usually represented in maps which form the basic layers of many GIS systems (e.g. hydrographic as well as topographical maps) concerning a common geographic area.

Usually, by starting the conceptual design of a new GIS application, the experienced analyst already knows a minimum set of classes, pertaining to the application domain, and the main associations between such classes (Gordillo et al., 1997). Therefore, the data modeling process evolves from this initial set of classes. In fact, this development approach is being used for a long time already, but it has never been supported by any mechanism which could assist the analyst in this initial stage of the project.

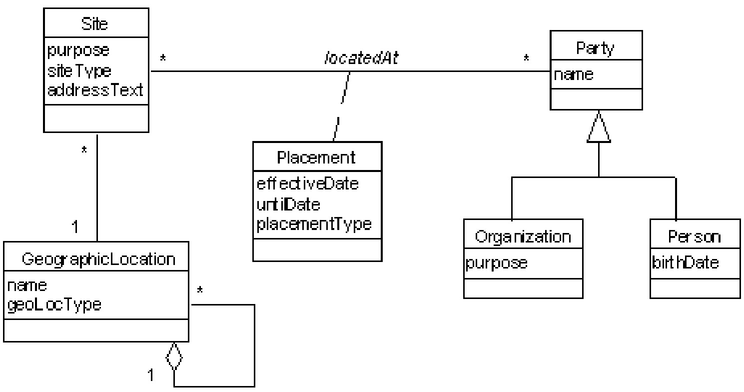

Figure 1 – Geographic Locations Analysis Pattern [HAY95]
According to Hay (1995), an analysis pattern should be customized for specific situations. The above described pattern provides an example of a case in which there is no final model. As can be seen in this example, a pattern definition may rely on other pattern definitions. The generalization of the Party class has actually been defined as a pattern itself in both (Hay, 1995) and (Fowler, 1997). On the other hand, the GeographicLocation class can be reused in the design of several other GIS applications.

The pattern approach provides an important contribution to this problem. Relying on it, provided that patterns can be defined and applied according to some formalism, it is possible to develop software mechanisms that are capable to both integrate pattern definitions into either existing or new geographic information models and also translate these definitions in definition languages accepted by commercial GIS products.

The use of analysis patterns implies some changes in the way conceptual data modeling is done today. Besides the usual requirements analysis (or engineering), the designer should search one or more pattern libraries trying to find already existing solutions for the modeling requirements he/she identified through the requirements analysis.

In fact, the study of existing patterns for some applications domain can give the designer helpful insight about the geographical reality being modeled. In this way, the requirements analysis process can be carried out with much more confidence as well as understanding.

According to Gerth et al (1996), besides allowing the reuse of knowledge and experience, patterns can improve the quality of software documentation as well as enhance the communication between designers.

As for the pattern approach to both be effective and evolve, every designer who makes use of pattern catalogues should try to optimize existing patterns relying on his/her own modeling experiences as well as to derive new patterns and insert them into a catalogue as soon as they have proven to be effective in the designer's own modeling process.

It should be clear by now that for the pattern approach to be helpful for designers pattern catalogues should be well organized and easy to both find and access. These aspects will be discussed in more detail in Section 5.

The use of patterns in GIS modeling depends on four important requirements:

- GIS designers should be interested in sharing their experiences;
- A standardized pattern description form should be adopted by the GIS designers community;
- Either a new or an extended data model should be adopted that is both suitable for GIS data modeling (Iochpe and Lisboa, 1998) and allows for the definition and integration of patterns as they can be described using the standardized pattern description form;
- Software mechanisms should be developed that support designers in selecting as well as locating, accessing, and reusing specific patterns, possibly on the basis of some query language.

The culture of data sharing through the Internet is widespread among GIS users all over the world. This is leading to a huge proliferation of sites that offer some kind of (interesting) data. The concepts of both metadata (Weber et. al., 1998) that can describe available geographic data sets as well as their actual location in the network, and that of a clearinghouse system that keeps metadata bases and offers a query language interface on them are greatly contributing to organize this ever growing, worldwide federated geographic database system.

Similarly, we believe that GIS application designers should be supported by some patterns catalogue system distributed over a clearinghouse system where patterns are described and made available together with the metadata related to them. We also believe that GIS application designers will have interest in sharing their successful experiences in GIS data modeling as it already occurs in the community of object-oriented designers.

4.1 An Example of GIS Analysis Pattern

To illustrate the potential use of analysis pattern in the GIS domain, this sub-section describes a pattern called *Environmental Quality Parameters*. We first identified this pattern during the data modeling process of a coastal management information system (SIGERCO-RS) that has been developed in partnership with FEPAM/RS - Foundation for Environmental

Protection of the State of Rio Grande do Sul (Lisboa and Iochpe, 1996).

Some time later, the same pattern was reused during the data modeling process of SIGPROGB, the GIS of Pró-Guaíba, a program financed by the World Bank and the Government of the State of Rio Grande do Sul (Lisboa et al., 1997). Both applications belong to the environmental domain and, thus, present various similarities concerning their respective data models.

Pattern name: Environmental Quality Parameters

Problem: how to model data representing environmental quality parameters?

Context: the quality of some environmental elements (e.g. water, air) is assessed trough periodical measurements of related quality parameters in different sampling points.

Forces:

- The number of parameters can vary for each sampling point during the monitoring time interval.
- The measurement periodicity can vary for each sampling point.
- Each parameter is always measured in the same unit.

Solution: Figure 2 shows a class diagram for the pattern. This pattern is based on the *Measurement Pattern* described in (Fowler, 1996). In each sampling point is taken, periodically, several measurements, corresponding to the analyzed parameters of quality control. We drew an association between the class Unit and the class ControlParameter because each parameter always is measured in the same unit of measurement. If this were not necessary, the association would be between the classes Unit and Measurement, which would cause a problem of replication.

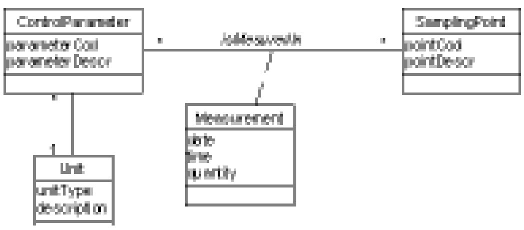

Figure 2 – Environmental Quality Parameters Pattern

Example: Figure 3 shows a segment of the conceptual data schema of the SIGPROGB Project, where the pattern was customized for the control of water quality parameters. In this case, each sampling point is located within a hydrological resource. Thus, the SamplingPoint class is associated with the HydroResource class that, in turn, has other associations in the context of the project. The diagram also presents especial features of GIS conceptual data modeling as, for instance, the use of graphic pictograms, proposed by (Caron and Bédard, 1993) and used in (Iochpe and Lisboa, 1998), for specifying the type of spatial object that represents each geographic entity.

5. GIS Patterns Catalogue

The reuse of good data modeling solutions greatly depends on how hard are analysis patterns to be found. Therefore, patterns must be well organized in easy-to-find patterns catalogues.

On the other hand, since the number of registered patterns would tend to grow quickly, once designers really buy the patterns approach, the GIS designers community should count on either “de facto” or official standards of both an unified GIS conceptual data model which prescribes a pattern description

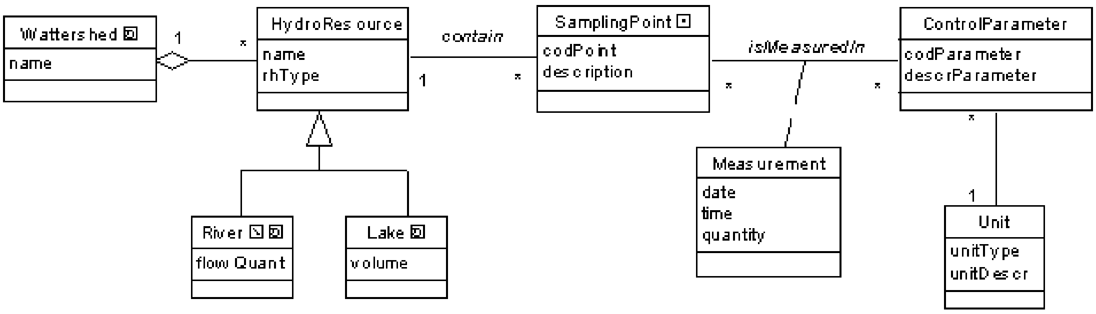

Figure 3 – Example of modeling water quality parameter

form and a set of classification rules for GIS analysis patterns.

Also important in this context is the concept of a *Pattern Language*. It was first proposed by Alexander et al (1977) and has been used by many design patterns authors. According to Coplien (1994), “a pattern language is a structured collection of patterns that build on each other to transform needs and constraints into an architecture”.

No analysis patterns language has been proposed until now. It would surely help GIS modeling designers to classify as well as describe sets of patterns on the basis of either common or related problems they can solve. This kind of classification can be very useful in the search of patterns by application domain. For instance, a pattern called *StreetNetwork* which describes object classes pertaining to a road network and a pattern called *Routing* which models street directions for each street segment in a network can be catalogued together into a pattern language intended to urban applications. This aggregation of patterns could be applied to help modeling a Bus Routes System.

A prototype that extends a clearinghouse system with a catalogue of GIS analysis patterns is under development at the Instituto de Informática of the Universidade Federal do Rio Grande do Sul. By accessing the clearinghouse’s query interface, it shall be possible for users to search for GIS analysis patterns much in the same way they would be searching for spatial data sets. We intend to create indices for patterns using the same mechanism which creates indices for spatial metadata. In addition, a patterns catalogue query subsystem shall be implemented.

An advantage that can be observed when metadata and analysis patterns are integrated in a clearinghouse system is that the metadata of the data produced for an application can make reference to the respective pattern which describes them. As a consequence, the pattern’s object schema can be used to update the database schema of the GIS in which the data will be incorporated.

6. Conclusions and Future Work

This paper explores the potential of analysis patterns in what concerns conceptual data modeling reuse in GIS applications. As the pattern approach is a new research theme even for the software engineering area, there is no consensus about a description form for analysis patterns. According to Riehle and Züllichoven (1996), the best way to describe a pattern is by describing its foreseen usage. Currently, we are investigating proper approaches for patterns description as well as best structures for a GIS patterns catalogue.

The results of this research shall be extended, in a further step, to consider pattern types of different abstraction levels as well. With a mechanism based on reuse to support the development of GIS applications, the whole software development cycle can become more efficient. Analysis patterns can allow analysts and designers to reuse the knowledge of experienced professionals mainly during the initial stages of GIS design (e.g.: requirements analysis and conceptual modeling) which have lead to low quality projects due to the lack of experience of nonspecialist designers.

Acknowledgements

This research was developed in the scope of SIGMODA Project, financed by grants of RHAE-CNPq and CAPES-PICDT.

References

Alexander, C.; Ishikawa, S; Silverstein, M.; Jacobson, M.; Fiksdahl-King, I. and Angel, S. (1977) *A Pattern Language*. New York: Oxford University.

Balaguer, F.; Gordillo, S. and Neves, F. das (1997) Patterns for GIS applications design. *Proceedings of PLoP’97 - Pattern Languages of Programming*, Monticello, Illinois, USA, 3-5 September.

Buschmann, F.; Meunier, R.; Rohnert, H.; Sommerlad, P. and Stal, M. (1995) *A system of patterns*. New York: John Wiley & Sons.

Caron, C. and Bédard, Y. (1993) Extending the individual formalism for a more complete modeling of urban spatially referenced data.

Computers Environment and Urban Systems, v.17, New York: Pergamon Press.

Coad, P. (1992) Object-oriented patterns. *Communications of the ACM*, 35:9.

Coad, P. (1997) *Object models: strategies, patterns, and applications*. 2.ed. New Jersey: Yourdon Press.

Coplien, J. O. (1994) *Software design patterns: common questions and answers*. AT&T Bell Laboratories Technical Memorandum.

Coplien, J. O. (1995) A generative development-process pattern language. In: James O. Coplien and Douglas Schmidt (eds.) *Pattern languages of program design*, Addison-Wesley.

Federal Geographic Data Committee (1997) *Content standard for digital geospatial metadata*. FGDC, Washinton (Revised April, 1997). Accessed in June, 1998. <http://www.fgdc.gov>

Fowler, M. (1996) Analysis patterns and business objects. *Proceedings of OOPSLA'96 - Workshop on Business Object Design and Implementation II*, ACM/SIGPLAN.

Fowler, M. (1997) *Analysis patterns: reusable object models*. Menlo Park: Addison Wesley Longman.

Gamma, E.; Helm, R.; Johnson, R. and Vlissides, J. (1994) *Design patterns: elements of reusable object-oriented software*. Reading: Addison Wesley.

Gerth, Th.; Schachtschabel, R. and Schönefeld, R. (1996) Using patterns in design and documentation of software. *Proceedings of WOON'96 - The white OO nights*. St. Petersburg, Russia, 20-21 June.

Gordillo, S.; Balaguer, F.; and Neves, F. das (1997) Generating the architecture of GIS applications with design patterns. *Proceedings of 5th ACM Workshop on Geographic Information Systems*. Las Vegas, NE, USA, 13-14 November.

Hay, D. C. (1995) *Data model patterns: conventions of thought*. New York: Dorset House Publishing.

Hay, D. C. (1996) Using data model patterns for rapid application development. *Proceedings of International Oracle User Week, Oracle*.

Iochpe, C and Lisboa F., J. (1998) BCH/GIS: a basic class hierarchy to support GIS conceptual design. *Proceedings of International Conference on Modeling Geographical and Environmental Systems with Geographical Information Systems (GIS)*. Hong Kong, China, 22-25 June.

Lisboa F., J. and Iochpe, C. (1996) Adapting the OMT object model to conceptual modeling of GIS applications. *Proceedings of 1ª SEGEO-RJ*, Rio de Janeiro, Brazil, 7-10 October. (In Portuguese)

Lisboa F., J.; Iochpe, C. and Garaffa, I. M. (1997) Conceptual data models for geographic applications: an experience with an inter-institutional GIS. *Proceedings of IV Simpósio Brasileiro de Geoprocessamento*, São Paulo, Brazil, 4-6 November. (In Portuguese)

Marr, A. J.; Pascoe, R. T. and Benwell, G. (1997) Interoperable GIS and spatial process modelling. *Proceedings of GeoComputation'97 & SIRC'97*, Dunedim, New Zealand, 26-29 August, University of Otago, pp 15-21.

Nebert, D. D. (1996) Information Architecture of a Clearinghouse. *In: WWW Access to Earth Observation/Geo-Referenced Data Workshop*, World Web Conference, 5-6 May. Accessed in March, 1998. <http://www.fgdc.gov/clearinghouse/>

Pree, W. (1995) *Design patterns for object-oriented software development*. Reading, MA: Addison-Wesley.

Rational Software (1997) *UML Notation Guide* Version 1.1. Accessed in June, 1998. <http://www.rational.com/uml/html/notation/>

Riehle, D. and Züllichoven, H. (1996) Understanding and using patterns in software development. *Theory and practice of object systems*, 2:1.

Weber, E. J.; Lisboa F., J.; Iochpe, C. and Hasenack, H. (1998) Geospatial metadata in Brazil: an experience in data documentation of an environmental GIS application. *Proceedings of GIS Planet 98 - International Conference & Exhibition on Geographic Information*, Lisbon, Portugal, 7-11 September.