

## A Software Architecture for Automated Geographic Metadata Annotation Generation

Luiz A. P. Paes Leme, Daniela F. Brauner, Marco A. Casanova, Karin. Breitman

Departamento de Informática – PUC-Rio  
Rua Marquês de São Vicente, 225 – 22.453-900 – Rio de Janeiro – RJ – Brazil  
{lleme, dani, casanova, karin}@inf.puc-rio.br

**Abstract.** *To facilitate data search and retrieval, a geographic data repository publishes metadata about the data resources it houses. Metadata generation is, however, a time consuming and error prone activity. Exploring unique characteristics of geographic objects, we first discuss a strategy that combines federations of gazetteers, thesauri and catalogs to harvest the information required for metadata annotations. Then, we propose a software architecture, ISO 19115:2003 compliant, for automated geographic metadata annotation generation. Finally, we describe the GeoCatalog tool, an implementation of the proposed architecture that demonstrates the viability of our approach.*

### 1 Introduction

The large volume of geographic data available today opens unprecedented opportunities for data interchange, facilitating the design of new geographic information systems (GIS) and claiming for the redesign of traditional ones. To secure interoperability among applications, however, it is fundamental to count on effective mechanisms to help locate and access relevant data. Current practice is based on the use of gazetteers and metadata catalogs that help applications discover the information they require.

A major holdback of this practice is the difficulty of manually creating metadata annotations in large scale, a process that is tedious, and sometimes unfeasible, depending on the volume of data under consideration. Fortunately, geographic objects have two valuable characteristics that help us automate the metadata annotation process. First, a geographic object is typically georeferenced, which functions as an approximation of a universal identifier for the object. Second, there are many gazetteers readily available on the Web that act as geographic object dictionaries and contain a significant amount of reliable information about geographic objects. Examples of such gazetteers are the ADL Gazetteer [Hill et al.1999] and the GEOnet Names Server [GNIS2005].

In this paper, we propose a software architecture for automated geographic metadata generation, a feature that every geographic catalog application should be equipped with. We also describe the GeoCatalog tool, an implementation of the architecture that demonstrates the viability of our approach.

The rest of this paper is organized as follows. Section 2 presents basic definitions. Section 3 provides a summary of the standards used in our proposal. Section 4 describes an example of the strategy for automated geographic metadata generation and cataloguing. Section 5 presents the proposed architecture. Section 6 introduces the

GeoCatalog tool. Section 7 discusses related work. Finally, section 8 contains the conclusions and directions for future work.

## **2 Definitions**

In this section, we briefly summarize the concepts that are central to our proposal.

A *data container*, or simply a *container*, is a recipient used to store data. It may be a file in the file system, a long field in a relational database table, or any other recipient which holds data. The notion of *data container type* generalizes the notion of file format.

A *data repository* is an identifiable collection of data containers. It may be a Web location, an FTP location, a file system directory, or a database table, for example.

A *gazetteer*, as defined in WordNet, is a geographical dictionary (as at the back of an atlas) containing a list of geographic names, together with their geographic locations and other descriptive information [Miller1995].

A *feature* is an abstraction of a real world phenomenon and a *geographic feature* is a feature associated with a location relative to the Earth [Percival2003]. In the familiar Computer Science jargon, a (geographic) feature is an object with a special attribute that describes the object location on the Earth surface, using a given *coordinate (geo)reference system* (CRS).

A *geographic name* is a proper name for a geographic feature, such as City of Rio de Janeiro, the Copacabana Beach, and the Sugarloaf Mountain.

*Metadata*, as also defined in WordNet, is data about other data. Examples of geographic metadata are scale, geographic projection, etc.

Finally, a *metadata catalogue* holds metadata describing data containers stored in data repositories [Nebert2002]. Typically, a catalogue does not store or manage the data containers themselves.

## **3 Standards**

We summarize the geographic metadata standards that are fundamental to understanding the architecture proposed in section 5 as follows.

The *ISO 19115:2003* [ISO/TC2112003] standard defines a metadata schema for geographic data and services. The standard defines metadata elements (id, extension, quality, temporal and spatial schema, spatial reference and data distribution), a conceptual schema and a common terminology for metadata. The elements were chosen to facilitate answering the following questions: “Is the data about some specific topic?” (what); “About some place?” (where); “About a specific time and period?” (when); and “Who should I contact to obtain further details or obtain a copy?” (who).

The OGC Catalogue Service (CS) 2.0 Specification [Nebert and Whiteside2005] defines metadata catalogue application interfaces, including metadata search and maintenance operations, resource search and retrieval, and session control. To be compatible with a service specification, a catalogue service should implement a minimum set of interface operations. In this scenario, several catalogues may provide

different services, e.g., one catalogue may provide querying operations, a second one may allow querying and update, a third one may implement querying in catalogue federations.

#### **4 An example of the strategy for Automated Metadata Extraction from Gazetteers**

In this section, we exemplify the strategy for automated metadata generation proposed in [Brauner et al.2006] that combines georeferenced data and gazetteers to generate useful metadata annotations.

As an example of how to generate *desc(C)*, suppose that we adopt the ADL Gazetteer and the ADL Feature Type Thesaurus. Consider the image of the City of Rio de Janeiro depicted in Figure 1. Table 1 instantiates this process.

**Table 1. Algorithm for extracting gazetteer entries related to a geographic data**

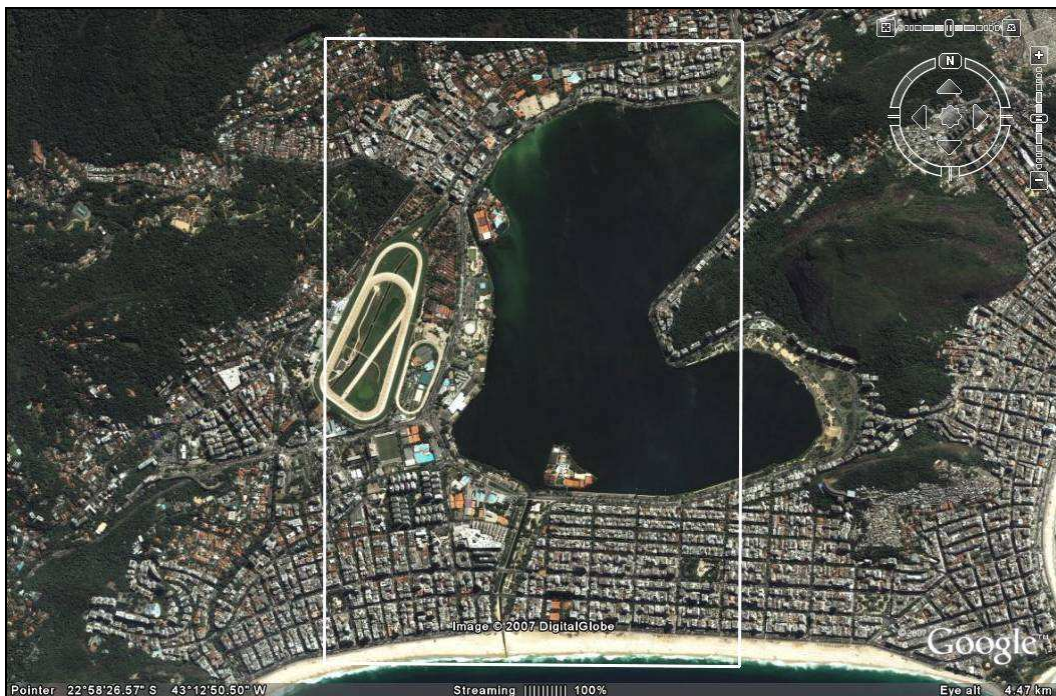
<ol style="list-style-type: none"><li>(1) Extract the georeferencing parameters from the information resource. In this case, the image fragment is consistent with a scale of 1:25,000 and has a bounding rectangle defined by the pair of coordinates ((43.224W, 22.960S), (43.204W, 22.988S)) – white rectangle in Figure 1.</li><li>(2) Assume that the user chooses to relate the image fragment with “hydrographic features”, a term of the ADL FTT that, in our running example, can be used to classify geographic datasets.</li><li>(3) Since the ADL Gazetteer entries have no associated scale information, ignore scale.</li><li>(4) Retrieve the geographic features within the bounding box parameters extracted in step (1) from the ADL Gazetteer and combine with the ADL FTT terms under “hydrographic features” (the term selected in step (2)). The query returns three entries, which are:<ul style="list-style-type: none"><li>• <i>Feature(“Rodrigo de Freitas, Lagoa - Brazil”, lakes)</i></li><li>• <i>Feature(“Leblon, Praia do - Brazil”, beaches)</i></li><li>• <i>Feature(“Leblon - Brazil, populated places)</i></li></ul></li><li>(5) Store the result of the query as a description of the image, that is, as a list of pairs (N,r), where N is a geographic feature returned in (4) and r is the topological relationship between the image and N (in this case, r is “within”)</li></ol>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### **5 A Software Architecture for Automated Geographic Metadata Annotation**

Based on the existing standards, combined with the strategy summarized in the previous section, we propose a software architecture for automated metadata generation.

The core of our proposal is the metadata *Harvest Service*, which directly retrieves metadata records from a set of external sources. It is depicted in Figure 2 by the thick line, bottom rectangle that encapsulates a set of three gray components. This service extends the OGC Catalogue Service reference *harvestResource* operation

[Nebert and Whiteside2005] acting as an automated preprocessing step to metadata cataloguing. The OGC standard specification does not make any considerations on how metadata is to be extracted from the external containers. We elaborate the OGC vague “load metadata records” directions, and propose a chain of subprocesses, each responsible for harvesting metadata from a distributed repository, including data containers, public gazetteers and metadata catalogues.

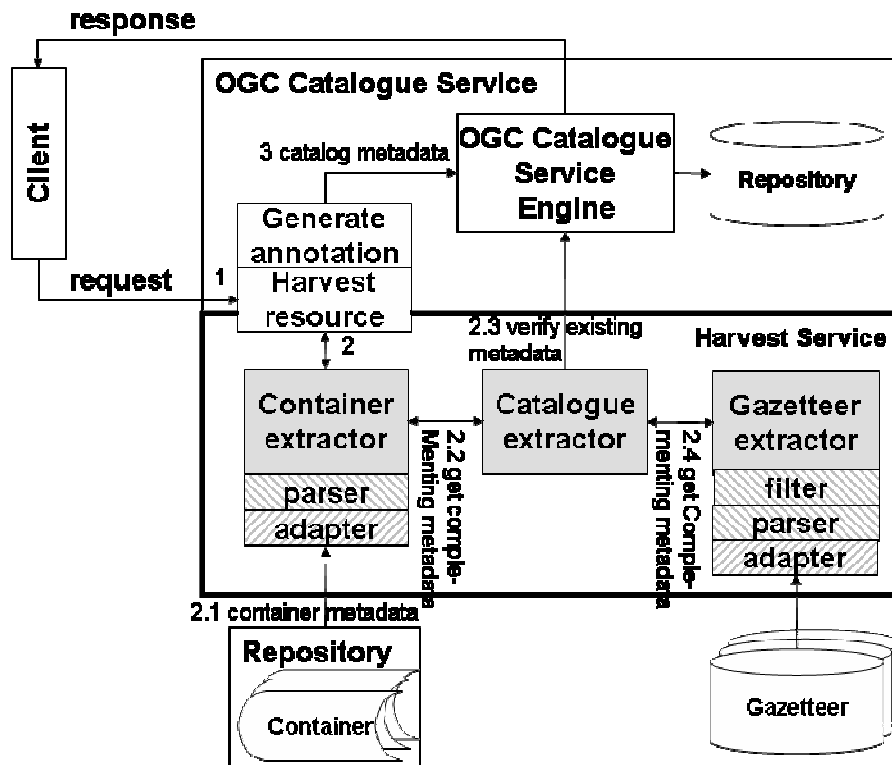


**Figure 1. Image of the City of Rio de Janeiro.**

The original *harvestResource* operation receives as input parameters a *source* (URL location) of data containers, a *resourceFormat* that specifies a container type, a *responseHandler* that specifies an URL where the operation response should be forwarded to, a *timeInterval* used to refresh captured metadata and a *resourceType* that identifies the type of repository to be crawled. In our architecture, the *resourceType* will be used as a data access configuration file. For example, if the repository is an FTP location, the URI would refer to a file containing specific connection information such as user, password, root directory, etc. On the other hand, if the repository is a file directory the URI would just contain a directory name.

The proposed architecture is divided into two major blocks, one responsible for harvesting and the other responsible for generating metadata annotations. In Figure 2 harvesting components are represented by the gray rectangles whereas metadata generation components are represented by white ones. We detail the proposed architecture components by describing their role in a prototypical metadata annotation generation process. The process initiates when a harvest request is received by the *Harvest Service* (step 1 in Figure 2) and forwarded to the *Container Extractor* (step 2). The last is responsible for inspecting as many *data container types* as available, using a collection of appropriate adapters. At this stage of the process, every *data container* is inspected for available metadata (step 2.1). We assume that *container data formats* are

specific to the geographic application domain, e.g. GeoTIFF, Shapefile, limiting the total of required parsers to a subset of the usual formats. In this process, a minimal set of data needs to be harvested from the data containers, including the geographic coordinates of the bounding box covered by the data, its scale and a classification in terms of the geographic feature(s) contained in the object in question, if available.



**Figure 2. Proposed architecture for automatic metadata annotation generation**

After extracting a minimal metadata set, the *Container Extractor* component delegates (step 2.2) to the *Catalogue Extractor* component the responsibility of querying the local catalogue in search of duplicates (already catalogued metadata for a similar data container – step 2.3). This process is accomplished by querying the local repository with the metadata already extracted from the data container in question. Duplication is very common especially in repositories that store temporal series of data containers, for example. In these cases, we want to avoid wasting computational effort in harvesting metadata that is already in the catalogue (created for other data containers of the same time series).

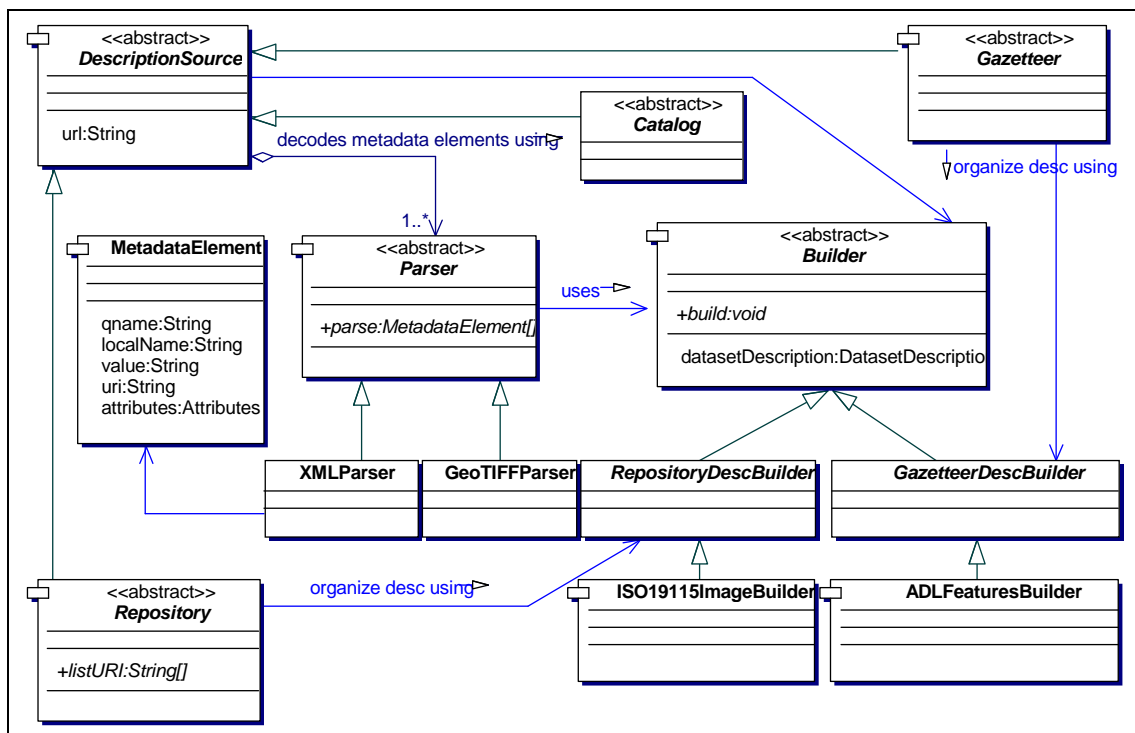
The next step uses the *Gazetteer Extractor* component, responsible for complementing the minimal metadata set with information retrieved from public gazetteers (step 2.4 in Figure 2), discussed in section 4. Please note that, similarly to the *Container Extractor* extension component, the *Gazetteer Extractor* also comprises a collection of adapters and parsers which provides the extractor the capability of interacting with gazetteers that follow different standards.

In Figure 3, we detail the architecture of the *Harvest Service* components. For the sake of simplicity, we assume that only the GeoTIFF format is in use, thus reducing the number of required parsers to one. Typically, the architecture will comprise as many

parsers as there are different data formats being used. The same is true for the gazetteers; we only show constructs for the ADL gazetteer (specialized by the *ADLFeaturesBuilder*). The complete architecture contemplates the instantiation of additional constructors to deal with gazetteers that use different metadata schemas.

The second block of components of the architecture is responsible for the generation of metadata annotations. These components are represented by the two white rectangles in Figure 2. To keep the architecture as general as possible, we propose an intermediate adapter layer that acts as the interface to the metadata schema in use.

The adapter layer is responsible for implementing the necessary methods to access information stored according to each particular schema. In Figure 3, we exemplify the adapter layer for the *ISO 19115:2003* schema. The adapter layer thus provides a level of flexibility that allows compliance of the proposed architecture with several metadata schemes, standardized or not. The decision of choosing which schema (profile) to adhere to is thus delegated to the group that is adopting the proposed architecture, which then becomes responsible for constructing the adapter to the schema of his (her) choice.



**Figure 3. Detail of the harvesting information components architecture**

Once the metadata annotation is generated, it is ready for inclusion in the catalogue. The annotation is then forwarded to the OGC Catalogue Service Engine component.

### 5.1 How to generalize the proposed architecture to other domains

The development of the proposed architecture was motivated by our need to scale up the process of developing metadata annotations for geographic data containers. The

## e-Science 2007 e-Science Workshop

proposed architecture, however, can be generalized to other application domains that use catalogues as a means to search and retrieve objects of interest. The basic requirements for extending our approach to any application domain can be generalized as follows:

1. The existence of a (good approximation for a) unique identifier for objects in that domain.
2. The existence of dictionaries, repositories or catalogues that contain descriptions of objects in the domain in question and that can be queried to enrich the metadata annotations. A minimum of structure is required, to enable the construction of software components that automate the metadata harvesting process.
3. Ideally the application domain in question may have some standard that can serve as the metadata schema for indexing objects in that domain. In case there are no standards available, the users (of the proposed architecture) will choose a structure for the annotation that better serves their needs.

A viable scenario is the instantiation of our architecture to the cultural heritage domain. Cataloguing our cultural heritage has naturally been a major activity of museums and other cultural institutions throughout the world. Today, most major museums make their collections available digitally over the Web. Many institutions have been working in standards for describing information about works of art.

Our architecture can be instantiated to this scenario by the adoption the following:

**Identifier:** to the best of our knowledge, there is not standard that provides a unique reference for a work of art. In this case, an acceptable approximation would be a combination of the creator(s), date and title of the work.

**Dictionaries:** the Getty Research Institute has done a remarkable work in developing thesauri and controlled vocabularies. The Art & Architecture Thesaurus (AAT) [Getty2006a] and the Union List of Artist Names (ULAN) are ISO compliant thesauri compliant that contain terms, names, and other information about people, places, things, and concepts relating to art, architecture, and material culture [Getty2006c, Getty2006b].

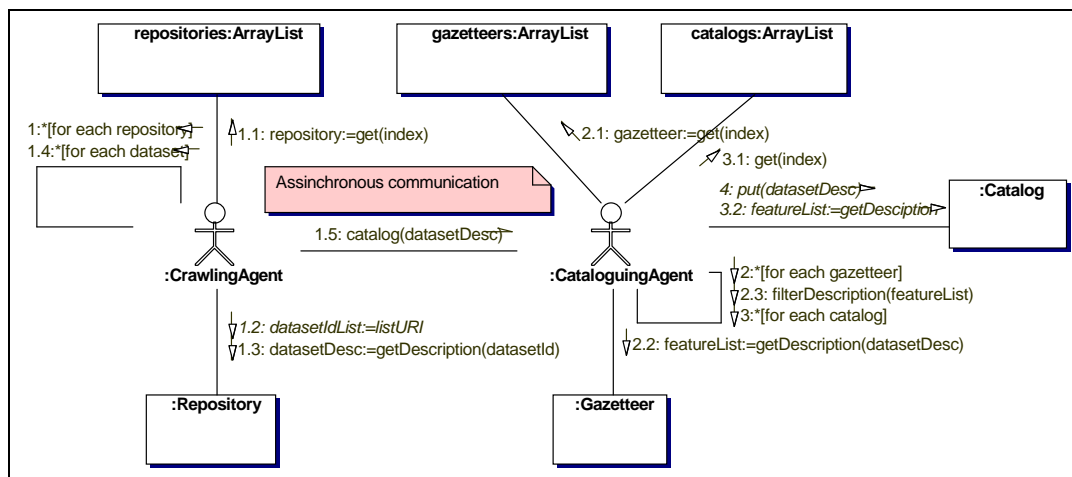
**Metadata:** the International Committee for Documentation of the International Council of Museums (ICOM-CIDOC) published the CIDOC Conceptual Reference Model [Crofts et al.2003, CIDOC2006], that provides definitions and a formal structure for describing the concepts and relationships used in cultural heritage documentation. The CIDOC CRM was accepted as a working draft by ISO/TC46/SC4/WG9 in September 2000 and is currently in the final stage of the ISO process as ISO/PRF 21127 (ISO 2006).

### **6 GeoCatalog tool**

In this section, we demonstrate the feasibility of our approach by presenting the GeoCatalog tool, built using the architecture proposed in section 5. GeoCatalog was implemented using software agents, which provide the application with two independent processes, one for crawling new data and other for cataloguing metadata. The first type of agents, *CrawlingAgents*, is responsible for crawling data repositories in search of new

data containers to be catalogued. Once a crawling agent finds a data container, it collects whatever metadata is available in the data container itself or in the repository where the container was found and sends a message to a cataloguing agent requiring that the new item be catalogued.

Once a request is received, a cataloguing agent uses metadata contained in the message to generate queries to registered gazetteers for relevant geographic features. Based on the additional data found in the gazetteers and after applying a set of filters, the agent produces metadata annotation for that data container and commits it to the catalogue application in use.



**Figure 4. Software Agents in the GeoCatalog application**

In Figure 4 we illustrate how GeoCatalog implements the example described in Table 1, section 4. In the left top corner is the XML input file for the Inmarsat satellite remote sensing image depicted in Figure 5a. Figure 5b shows a screen snapshot of GeoCatalog, processing the input file. Finally, Figure 5c depicts part of the resulting metadata annotation for the remote sensing image.

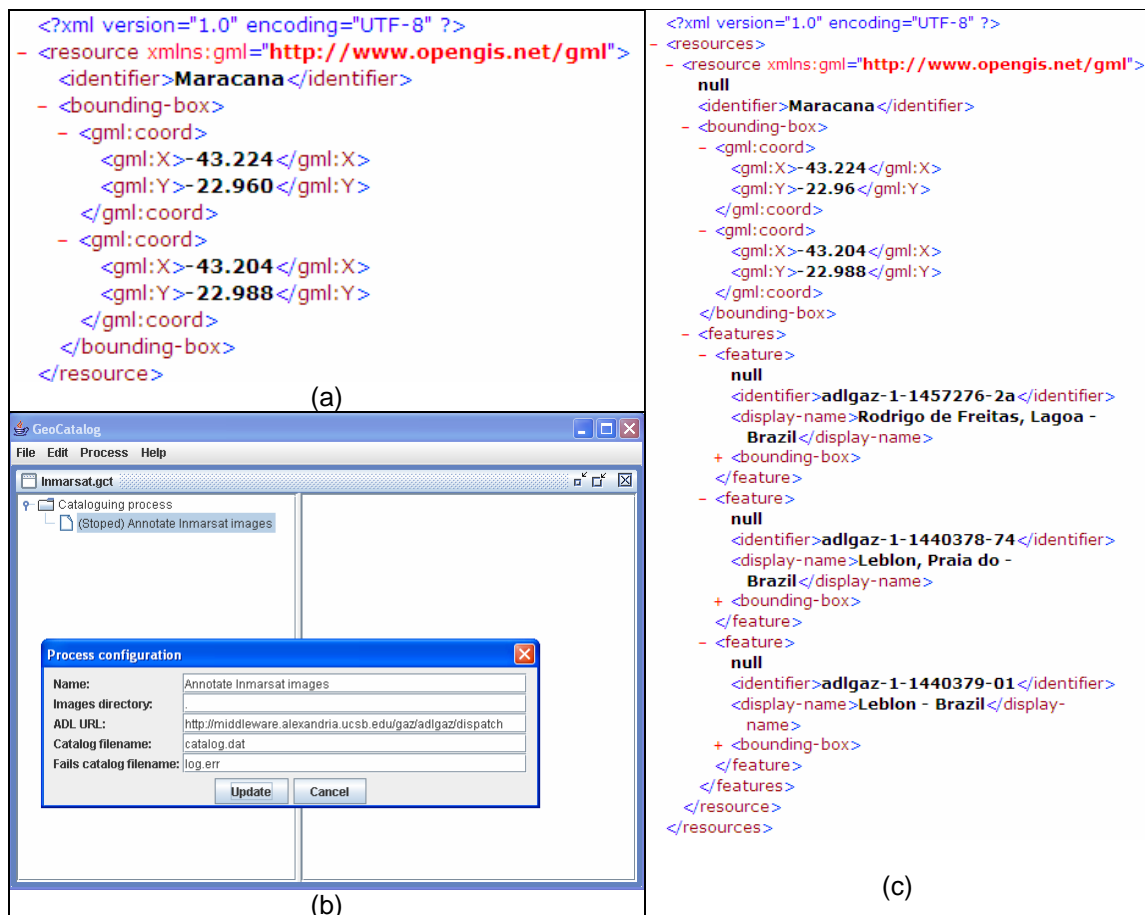
## 7 Related Work

A series of solutions have been proposed to help deal with data cataloguing. [Klien and Lutz2005] propose a method for automating the annotation process based on spatial relations. In [Hollink et al.2003], the authors propose a similar approach for the semantic annotation of art images. [Hollink et al.2004] expand the work to include spatial annotations relating objects depicted in images. In [Souza et al.2005] they propose a tool, to be used in conjunction with Web search engines, to allow data selection based on geographical parameters. [Hiramatsu and Reitsma2004] propose two tools to deal with geo-referenced information.

## 8 Conclusions

In this paper, we proposed a software architecture that is responsible for geographic data identification, and for automated metadata annotation generation and cataloguing. An important aspect of the proposed architecture is the adherence to well-known standards, such as the *ISO 19115:2003* and the OGC Catalogue Service (CS) 2.0 specification.

Our motivation comes from previous GIS projects by the authors, where collecting metadata proved to be a bottleneck difficult to overcome, given the large volume of geographic data containers involved. Furthermore, the new containers were quite often part of a time series, or they contained newly acquired data from the same geographic area. Hence, some of their metadata could be simply copied from other entries in the catalogue. The GIS projects also suggested that the quality of the metadata could be enhanced by relating the new data containers to geographic names from a standard gazetteer, or from a private gazetteer holding information about the company industrial installations.



**Figure 5. Example of the GeoCatalog application.**

## Acknowledgements

This work was partly financed by CNPq, through projects 550250/2005-0, 551241/2005-5.

## References

Brauner, D. F., Casanova, M. A., Breitman, K. K., and Leme, L. A. P. (2006). Using gazetteers to annotate geographic catalog entries. In Manolopoulos, Y., Filipe, J., Constantopoulos, P., and Cordeiro, J., editors, *Proceedings of the Eighth International Conference on Enterprise Information Systems: Databases and Information Systems Integration (ICEIS 2006)*, pages 215–220.

**e-Science**  
*2007 e-Science Workshop*

- CIDOC (2006). CIDOC conceptual reference model.
- Crofts, N., Doerr, M., and Gill, T. (2003). The cidoc conceptual reference model: A standard for communicating cultural contents. *Cultivate Interactive*, 9(7).
- Getty (2006a). Art & architecture thesaurus online (AAT). The J. Paul Getty Trust.
- Getty (2006b). Data standards and guidelines. The J. Paul Getty Trust.
- Getty (2006c). Getty vocabularies. The J. Paul Getty Trust.
- GNIS (2005). GNIS geographic names information system. U.S. Department of the Interior, U.S. Geological Survey, Reston, USA.
- Hill, L., Frew, J., and Zheng, Q. (1999). Geographic names: The implementation of a gazetteer in a georeferenced digital library. *D-Lib Magazine*, 5(1).
- Hiramatsu, K. and Reitsma, F. (2004). Georeferencing the semantic web: Ontology-based markup of geographically referenced information. In *Proceedings of Joint EuroSDR /EuroGeographics Workshop on Ontologies and Schema Translation Services*, Paris, France.
- Hollink, L., Nguyen, G., Schreiber, G., Wielemaker, J., Wielinga, B., and Worring, M. (2004). Adding spatial semantics to image annotations. In *International Workshop on Knowledge Markup and Semantic Annotation at (ISWC04)*.
- Hollink, L., Schreiber, G., Wielemaker, J., and Wielinga, B. (2003). Semantic annotation of image collections. In Handschuh, S., Koivunen, M., Dieng, R., and Staab, S., editors, *Proceedings Knowledge Markup and Semantic Annotation Workshop*, Knowledge Capture 2003.
- ISO/TC211 (2003). Geographic information - metadata. ISO international standard 19115. Technical report, ISO.
- Klien, E. and Lutz, M. (2005). The role of spatial relations in automating the semantic annotation of geodata. In *Conference on Spatial Information Theory*.
- Miller, G. (1995). Wordnet: A lexical database for english. *Communications of the ACM*, 38(1):39–41.
- Nebert, D. (2002). Catalog services specification, version 1.1.1. Technical report, Open GIS Consortium, Inc.
- Nebert, D. and Whiteside, A. E. (2005). Opengis catalogue services specification, version 2.0.0 with corrigendum opengis® implementation specification ogc 04-021r3. Technical report, Open Geoscience Consortium Inc.
- Percivall, G. (2003). Opengis reference model. *Open Geoscience Consortium*.
- Souza, L. A., Jr., C. A. D., Borges, K. A. V., Delboni, T. M., and Laender, A. H. F. (2005). The role of gazetteers in geographic knowledge discovery on the web. In *Proceedings of the Third Latin American Web Congress (LA-WEB05)*, page 157, Washington, DC, USA. IEEE Computer Society.