GRASS/ OSGeo-News
Open Source GIS and Remote Sensing informations Volume 1, June 2004

A generic approach to manage metadata standards

by Julien Barde and Duane Edgington and Jean-Christophe Desconnets

Inroduction

Informational Resources[1] (IR) management is a crucial part of environmental resources management. Indeed, the improvement of data processing and decision-making is strongly related to the ability to locate the relevant IR.

However, the exhaustive locating of relevant IR is a challenge for users as they face the following constraints:

- IR are *heterogeneous* (*language*, *semantic*, *syntax/formats*, *metadata*, *access constraints* because of their *rarity* and *cost*...),
- IR are *distributed* into *heterogeneous Information Systems (IS)* whose interoperability first involves syntactic and semantic/spatial matching issues (answers to a natural language query often require its translation into as many queries as different kinds of IS).

Thus, the key issue to improve data retrieval is a better management of the *{metadata element, value}* pairs, which constitute any metadata sheet. Once aware of existing IS, the priority to locate the relevant IR is the management of the matching between the *heterogeneous metadata elements (syntactic)*, as well as between their *heterogeneous values (semantic)*. The global scale of environmental domains and the multidisciplinary context of related studies strongly increase these constraints and the need of *semantic and spatial referentials* management.

[1] An *informational resource* (IR) is the whole of data, information, knowledge produced, needed or treated by users (regardless of their format: hardcopy or digital...). According to the users, this term covers a report, a map, a picture, a video, a dataset, a data serie, a database, a model....

Contents of this volume:

Metadata management

Metadata management is thus a priority before considering any data processing. Nevertheless, metadata management still faces the lack of referentials to homogenize: (i) the terminology of *metadata elements*, (ii) their *relationships* as well as (iii) their *values*. As a consequence, the quality of metadata management tools leans as much on the compliance with the reference standards specifications (syntactic: structure of metadata elements) as on the ability to use values from semantic referentials to edit instances.

The *heterogeneity of standardized metadata elements sets* interferes with *IS syntactic interoperability*:

- standards often use similar *core metadata elements*, designating the same *concepts* by using different *terms*. These elements answer the following questions: *Where*? *What*? *When*? *Who*?... and are essential to retrieve IR,

- moreover, standards with *redundant scopes* generate wider *matching issues* since the same kind of IR could be described with different standards (like *FGDC* and *ISO 19115*),

- setting up new *international metadata standards* makes the previous national/local standards obsolete and brings *archiving issues* (potentially related to the previous *matching issues*),

- the recent use of *XML Schemas* to standardize their mplementations decreases the redundancy of standards scopes and prevent wrong interpretations of their implementations [W3C (2004)]. Standards can be used as referential types libraries (eg the case of *OGC* standards like the update of *ISO 19115* with *ISO 19139*, *SensorML* ...).

The *heterogeneity of metadata elements values* interferes with *IS semantic interoperability* if values are not controlled (regardless of the chosen standard): the use of additional referentials is a key issue to improve IR descriptions and their retrieval by managing the *core metadata elements values* (which are used in priority in most of the queries). Among them, the management of the following descriptions are crucial as they are the most complicated and ambiguous:

- *terminologic description* management with common (multilingual) controlled vocabularies/semantic referentials to valuate *"keyword" like metadata elements*. Moreover, these referentials help to set up *shared vocabularies* in *pluridisciplinary* contexts,

- *spatial description* management with shared geographic / spatial referentials facilitated by friendly GUIs improves the complex use of geographic information (*GI*) (in particular the use of formats like *GML*, *WKT*...).

The need of multi-standards metadata management tools is increasing. Indeed, even a single institute or project often has to manage more than one kind of IR.

Moreover, by considering users' and software engineers' tasks and needs to manage metadata, it appears that most of them are similar regardless of the standard implemented.

User's needs and tasks

According to their roles (administrator...), whatever tools, *users tasks* to manage a metadata standard usually consists of:

- *profiling* the metadata standards for their specific uses,

- *editing* standards instances to describe their IR (by relating values to metadata elements of the chosen profile),

- *locating* (and eventually *acquiring* according to access rights) the relevant IR for a given work by using a single *multi-criteria search engine* which allows sophisticated *spatial queries*,

- *import/export* metadata standards instances (usually XML),

Users need assistance to perform these tasks easily with friendly tools which are currently lacking. Most of the time, users express the need for single centralized access and tools with GUIs whose ergonomy is friendly and homogeneous from one standard to another (since tasks are similar). Indeed, heterogeneous software/IS implementing different or similar standards strongly increase user's accommodation efforts (as users have first to become familiar with each software to perform these tasks and then consider they are wasting their time). Finally, users need complementary components for any complex valuation process (*Web mapping tool*, *controlled vocabularies*, *calendar*...) with complicated format (like *GML*...). These use cases are illustrated with a UML diagram in the related slideshow (here).

Software engineer's needs and tasks

In the same way software engineer's main tasks and needs remain similar from one standard to another.

Software (engineers) tasks consist of:

- *satisfying user needs* by *complying with standards*,

- managing the *matching between core metadata elements* of different standards to answer basic queries efficiently,
- integrating and managing existing *semantic and spatial referentials* to warranty the quality of IR descriptions and to manage query expansion process. Indeed, an efficient data retrieval involves the management of as many queries as existing IS. Answers to these queries are all more difficult since terms and geographic objects used are heterogeneous,
- providing a rich *spatial data infrastructure* to manage and eventually process related IR thereafter.

Software engineers need to *minimize their developments efforts* to implement metadata standards [Bernstein et al. (2000)]. They want to do so by answering similar user needs in the same (automated) way: by reusing a single scripts set and the same components (WMS...). This requires a generic approach (regardless of implemented standards) [Barde (2005)].

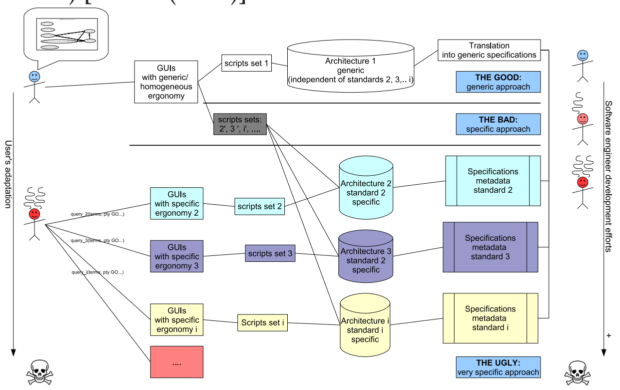

Figure 1: Benefits of a generic approach for multi-standards management

Generic approach vs specific approach

Traditional specific implementations lead to heterogeneous data storage systems by translating directly specifications into physical heterogeneous data models and thus require specific scripts sets to process them (see illustration in figure 1).

For example, by using a (manual or automated) generation of SQL from UML or XSD standards specifications, the resulting physical data models are going to be highly heterogeneous (as tables and columns names will match the metadata elements names). Scripts, to process their contents, have therefore to be adapted to these specific terms set to answer similar tasks. Scripts sets are thus heterogeneous from one standard to another.

According to the previous lists of tasks and needs, the figure 1 illustrates the benefit of a generic approach for both users and software engineers.

So far, existing tools don't cover these different needs as they mainly lean on specific approaches.

Generic models to manage efficiently {metadata element, value} pairs

We present in this section the ongoing generic models we are currently implementing to set up both a multi-standard metadata management tool and additional components which control the values of metadata elements by assisting the users at the same time. In particular these models will focus on the most crucial core metadata elements values which are related to thematic and spatial descriptions.

A generic model to manage heterogeneous metadata standards

The goal is to design a generic pattern (or conceptual model as shown in figure 2) to describe any metadata standard and then set up a generic metadata management system which allows the control of essential values. We suggest expressing a standard as an *inventory of structured metadata elements* with potential additional tools to fill their *content* with controlled *values* (according to standards specifications and/or software engineer's will).

This approach is close to DOM's[2] goal which involves similar concepts to manage *nodes* and their *relationships* as well as their *content* in any kind of document. However, we only focus on the specific case of metadata standards.

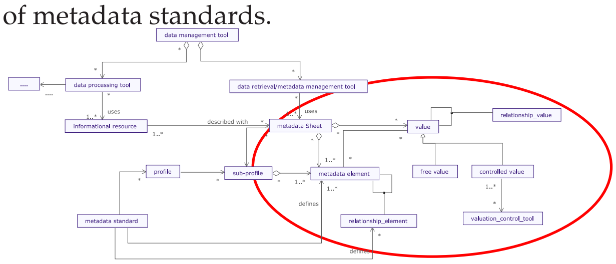

Figure 2: Generic expression of any metadata standard specifications

Nevertheless, a standard rarely aims to control the potential values of the core metadata elements, and even more rarely relationships between values (in particular terms and geographic objects, date/period...)). The control of such values is ruled by other specific standards. It is thus the role of the

[2] Document Object Model

software engineer to integrate these standards by setting up complementary tools to manage these specific values.

We will thereafter focus on the specific case of the management of spatial and thematic values. We suggest a new model to manage their relationships.

A generic model to manage heterogeneous (thematic and spatial) values

The different kinds of {Metadata element, value} pairs are more or less crucial for data retrieval. In particular, certain values are especially difficult to control. Among them, *thematic* and *spatial descriptions* are both crucial as they are related to core metadata elements, involved in most of the users queries (related to *Where*? and *What*? criteria). We aim to manage them in a generic way by focusing on the user's intention: by focusing on the management of underlying *spatial* and *thematic concepts* (by using a formalization of their properties and relationships, see figure 3).

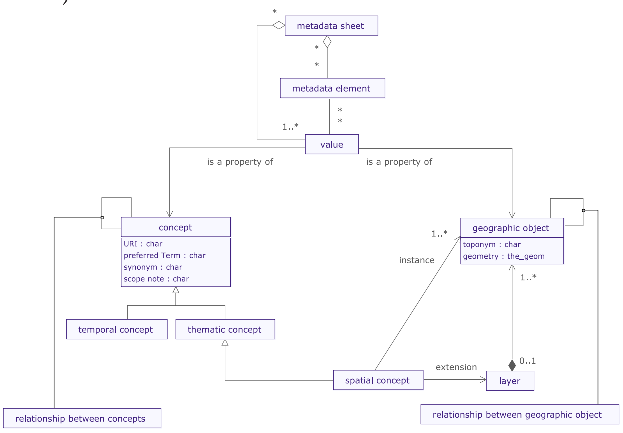

Figure 3: Properties and relationships of spatial and thematic concepts

Indeed, the use of terms as values related to core metadata elements is often ambiguous:

- users often formalize their IR descriptions or queries by using such *terms*: "*swordfish, sea temperature, Madagascar, spring*",
- semantic relationships management allows the system to relate different terms to expand these kinds of queries. For example by collecting other IR described with ("*Xiphias gladius...*") which is a *synonym* of "*swordfish...*") as it designates the same *concept* (in the same way that a picture, an image...),
- the case of a *toponym* brings a new problematic as this kind of term could be both considered as a keyword or a geographical description. In fact, the geographic object related to the term/toponym *"Madagascar"* could as well be designated graphically in a Web Mapping tool....

As illustrated in the figure 3, we suggest managing both *semantic and spatial relationships* between *thematic and spatial concepts* as well as *geographic objects* in the following way: "*a spatial concept as a kind of thematic concept whose instances are geographic objects*" [Barde et al. (2006)]. However it is important to consider that a *geographic object* is not necessarily related to a *term* or *toponym*.

The figure 4 summarizes the content of the previous generic models and give additional details to improve the management of both metadata elements and their values.

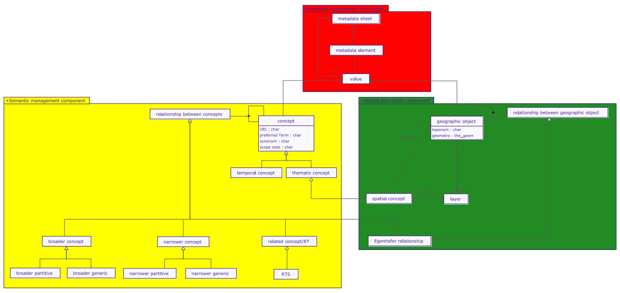

Figure 4: Summarization of the suggested generic approach

This model has been set up to be compliant with current reference standards implementation, for metadata, semantic and spatial information: standardized implementations of metadata standards (*XML Schemas*, *DTD*...), (Web) Semantic standards (*SKOS* - related to *ISO 2788* and *5964* standards-/*RDF*/*OWL*) and main GI standard formats. This generic model allows one to set up in a single architecture a *physical link between metadata elements and ontologies* to control their values (including spatial descriptions) and expand the queries efficiently.

In the same way, it is possible to set up additional controls for other crucial values: in particular *temporal* and *contacts descriptions* which answer the questions *When* and *Who*? Such control tools are usually *calendar* or *contacts directory* components (they manage date/period and human resources descriptions related to the IR).

The management of these additional referentials could be done independently of the metadata standards implemented. However, we aim to calculate the values of heterogeneous core metadata elements of the different metadata standards implemented in such a tool by using the same inventories of objects (managed in these referentials) as a basis for any standard. The management of these referentials in the same architecture facilitates the process. Thereafter by keeping the track of objects used to describe IR in a dedicated *generic common indexation table* which duplicates the main descriptions (*What*, *Where*, *Who*, *When*...), it will be possible to answer efficiently most of the users' requests (independently of the metadata standards used) by querying its records using richer values than standardized metadata element values (concepts URI instead of *terms*, 2D/3D geographic objects instead of *bounding boxes*...).

Model implementation with open source software

We present in this last section an implementation based on open source software.

Underlying technical choices

MDWeb is an open source software which is itself based on other open source software and standards. It implements this kind of architecture to set up a generic metadata management system. MDWeb:

- is a multistandard and multilingual metadata cataloging tool implementing a generic approach (like *M3Cat*, *MetaCat*...),
- is using a *three-tier* (client-server) architecture with:
  1. friendly *GUIs* (in Web browsers) with additional components (pop-ups) to assist metadata edition and searching:
     - *the spatial description* with Web Mapping tools which can be used as well to display the related GI: *Mapserver* / *Mapbuilder*,
     - *the thematic description* with Controlled vocabularies management GUIs to set up and browse of thesaurus / ontology: home made component.
  2. *applications scripts* (PHP/Javascript/XML with Apache Http server),
  3. *data storage*: *RDBMS* to manage metadata standards & spatial IR & related metadata & controlled vocabularies: *Postgres* with *Postgis* (WMS for remotely GI...). Import of *SKOS* files into Postgres by using *JENA* java API. *XML* repositories.

Additional details on the main characteristics of the suggested three-tier architecture for the physical data infrastructure can be found in the related presentation (here).

[3] Physical Data Model

Examples of a possible generic GUIs set

By using *MDWeb* as a basis to implement this approach, it is thus possible to meet users and software engineer's needs, in particular by having a single set of homogeneous GUIs, regardless of the implemented standard [Radha et al. (2005)]:

- *Import of any new metadata standard* by translating formal specifications into the PDM[3] (for now only XML Schemas specifications import is automated),
- Set up of *profiles* of imported standards,
- *Metadata sheet edition* with additional GUIs to assist (automatizations, control...) thematic and spatial descriptions of IR (as shown in figure 5),
- Generic/*multi-standard search engine*,
- *Import/export* of standardized (usually XML) metadata sheet.

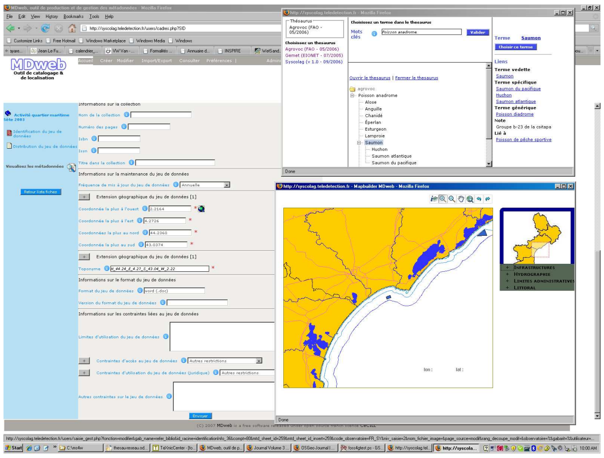

Figure 5: GUI to edit a metadata sheet

Conclusion and outlook

Data retrieval can be highly improved by managing metadata and their values in a better way. By implementating a generic approach (GUIs, scripts set, database) it is possible to manage into a single architecture :

- *heterogeneous metadata standards* (import, profiles, edition...),
- *heterogeneous values*: in particular *controlled terms* and *spatial descriptions* to describe *core metadata elements*,

- a *common indexation table* duplicating core metadata elements by using homogeneous values which can be used by used more efficiently by the *search engine* (no wrapper needed), especially to expand queries,
- *spatial IR described by metadata* can then be processed after being retrieved: either locally or remotely by using interoperable protocols or/and rich clients (*WMS*, *Qgis*, *Udig*…).

This kind of architecture is crucial to satisfy both *user's and software engineer's tasks and needs* by *minimizing adaptation and developments efforts* and by integrating at the same time the complementary tools to control crucial core metadata elements values.

Data retrieval is thus improved. In particular, by managing standardized semantic and spatial descriptions and their relationships in a same architecture, data retrieval can use *queries expansion* processes. It is thus possible to focus on specific use cases involving semantic and spatial relationships management like "find all the IR less than one mile of this geographic (platform/sensor…) object measuring the following physical parameter (temperature…)" by leaning on rich concepts, 2D or 3D geographic objects…. Moreover by using standardized semantic or spatial relationships (*W3C*, *OGC*…) the different kinds of queries can be exported and used in any kind of similar tool.

Generally, this implementation with an extensive use of *OGC standards and opensource software* increases its ability to interoperate with external IS.

Bibliography

Barde Julien
Monterey Bay Aquarium Research Institute (MBARI)
`http://www.mbari.org/staff/julien/`
`julien@mbari.org`

Bibliography

Bibliography

[Herborg et al. (2003)] L.M. Herborg, M.G. Bentley, A.S. Clare, S.P. Rushton (2003) The spread of the Chinese mitten crab (Eriocheir sinensis) in Europe; the predictive value of an historical data set. *Hydrobiologia* 503: 21-28.

[W3C (2004)] World Wide Web Consortium (W3C) (2004) XML Schema Part 0: Primer Second Edition. *http://www.w3.org/TR/xmlschema-0/*.

[W3C (2004)] World Wide Web Consortium (W3C) (2004) XML Schema Part 1: Structures Second Edition. *http://www.w3.org/TR/xmlschema-1/*.

[W3C (2004)] World Wide Web Consortium (W3C) (2004) XML Schema Part 2: Datatypes Second Edition *http://www.w3.org/TR/xmlschema-2/*.

[Barde (2005)] J. Barde (2005) Mutualisation de données et de connaissances pour la Gestion Intégrée des Zones Côtières. Application au projet SYSCOLAG. *universite Montpellier II* 285.

[Barde et al. (2006)] J. Barde, J. C. Desconnets, T. Libourel, P. Maurel (2006) Generic conceptual models for data and knowledge sharing. Application to environmental domain. *Hydroscience and Engineering, ICHE 2006* 16: 407-420.

[Bernstein et al. (2000)] Philip A. Bernstein and Laura M. Haas and Matthias Jarke and Erhard Rahm and Gio Wiederhold (2000) Panel: Is Generic Metadata Management Feasible? *The VLDB Journal* 660-662.

[Berkley et al. (2001)] Chad Berkley, Matthew Jones, Jivka Bojilova, Daniel Higgins (2001) Metacat: A Schema-Independent XML Database System. *SSDBM '01: Proceedings of the Thirteenth International Conference on Scientific and Statistical Database Management* 171. IEEE Computer Society

[Berkley et al. (2003)] Sergey Melnik, Erhard Rahm, Philip A. Bernstein (2003) Rondo: a programming platform for generic model management. *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data* 193-204. ACM Press

[Radha et al. (2005)] V. Radha, S. Ramakrishna, N. Pradeep Kumar (2005) Generic XML Schema Definition (XSD) to GUI Translator. *Distributed Computing and Internet Technology* 3816:290-296. IEEE Computer Society