THREDDS

S.Nativi
UNIDATA-UCAR
Boulder December 2001

A Proof of Concepts System

The Information and computational Models of a Proof of Concepts system for the THREDDS project

Stefano Nativi

Ver. 1.2

Draft

Internal diffusion

The Proof of Concepts System

Geographic Information Heterogeneity versus Interoperability

GI heterogeneity

Geographic Information (GI) has been characterised by heterogeneity for several aspects, such as:

- Conceptual modelling approach (e.g. composite approach versus geo-relational approach versus quad-tree approach);
- dataset granularity level (e.g. hierarchical versus unstructured data);
- metadata sharing approach (e.g. self-describing dataset versus well-known-structured dataset);
- dataset encoding format (e.g. ASCII versus binary encoding);
- dataset machine encoding (e.g. machine independent encoding versus machine dependent encoding);
- dataset encoding structure (e.g. HDF versus NetCDF, GeoTIFF versus Erdas LAN, ESRI Shape versus Autodesk MWF, etc.).

Such heterogeneity has been a drawback for GI sharing and integration among different Communities and has brought up the interoperability issue as a major effort to push GI market. And such need is even more important for the new computer mobile market.

GI interoperability opportunity

Within the Information Technology (IT) domain, the term "interoperability" signifies different combinations of various concepts depending on the context and the application.

As far as GI is concerned the most common interoperability solutions are better classified as *Integratability* achievements.

According to [1] integratability can be defined as: *the quality of possibly incompatible systems that makes it easier to adapt them or their exchanged data so that they will cooperate meaningfully for some purpose. Systems that are integratable can be made compatible with some nontrivial amount of effort, for example, by writing "glue code."*

In particular, the most common integratability solutions has been based on importing/exporting GI dataset.

Referring to the previously reported GI heterogeneity aspects, it is possible to distinguish among three main integratability levels:

1. Dataset structure integratability (resolving dataset encoding heterogeneity);
2. Metadata integratability (resolving dataset encoding and semantic heterogeneity);
3. Model integratability (resolving dataset encoding, semantic and modelling heterogeneity).

Naturally, the most common import/export functionalities achieve the dataset structure integratability, very few face semantic heterogeneity and even less model heterogeneity.

The present effort of GI community it is not only to achieve high-level system integratability, but also to develop system *Compatibility* (see [5] [6]).

According to [1], system *Compatibility* can be defined as: *the ability of systems to cooperate meaningfully for some purpose and/or to coexist without interfering with one another. Compatible systems work together as-is, not requiring any integration effort beyond some trivial configuration to let them know that they are supposed to work together*.

Standardised Reference Models

In order to achieve GI system high-level integratability, as well as compatibility, several international initiatives have been producing reference model specifications at various levels (i.e. abstract, implementation, recommendation).

Well-known specifications are: OGC OpenGIS, ISO 19100 standard series and FGDC specifications (see [6]).

These specifications wanted to constitute the foundations for implementing semantic and model integratability and tackle the more ambitious objective of system compatibility.

In summary, these specifications have been issuing abstract reference models which unifie: GI conceptual modelling approaches, GI metadata modelling, dataset encoding rules -platform independent, and several implementation specifications –platform dependent.

Naturally a valuable point is: how heterogeneous are these specifications coming up from different initiatives? That is: has the heterogeneity been shifted from the dataset structure to the modelling level?

As a matter of fact, a *Joint Steering Group to De-conflict Spatial Standards* was established in order to tackle such present risk.

The Web/XML technology revolution

The great success of Web technologies, especially as far as multi-platform possibilities are concerned, provided a new very promising framework for implementing high-level integratability and even compatibility among GI systems.

In particular, two technologies seem to be good foundations to achieve GI interoperability: Web Services and XML.

Web services

Web services are modular and reusable applications that can be invoked over the web (e.g. using a protocol like SOAP); they can be described using a service description language (i.e. like WSDL) and published into a registry (e.g. like UDDI) in order to allow service research, bounding and invocation (see [2]).

In particular, the following middleware technologies enable Web services:

Capability	Web Services
Network layer	TCP/IP
Web protocol	HTTP
Interface definition	WSDL
Meta language	XML
RPC approach	SOAP; XMLP
Registry/Repository	UDDI
Process flow	WSFL; ebXML
Modelling language	UML

The meta language used is the XML. Such language gained a tremendous consensus, especially as *Esperanto* for data and metadata encoding.

XML

Metadata is essential in order to enable collections of documents and data to be organised in catalogues, registries, libraries etc. When such collections are not held on just one system, XML is an ideal medium for metadata because it can be understood by many different applications and systems.

Moreover, metadata supports application integration and resource discovery. Most of approaches conceived to tackle such issues introduced an XML metadata solution.

XML metadata are here to stay, but it is important to bear in mind that XML metadata technology is a rapid evolving area, especially for the IT sector (see [3]).

GI and XML

It is natural that the main GI international reference models issued (or are issuing) dataset encoding rules for XML. Examples of such specifications are (see [6]):

Initiative	Specifications
OGC	GML 2.0 (and forthcoming GML 3.0)
ISO TC211	ISO 19118 standard
Database Promotion Center	G-XML 2.0 (and forthcoming 2.5)

Such specifications introduce rules for encoding specific application dataset in XML according to the respective reference models (e.g. OpenGIS and ISO 19100 standard series). Naturally, it is possible to introduce profiles of such encoding languages (i.e. to adopt only parts of the specifications and/or to introduce customised elements).

A main objective of these encoding specifications is to enable high-level integratability and even compatibility. Therefore, such XML-based specifications are -or will be - able to encode all the GI aspects previously listed. Naturally it is necessary to assume that inter-operating systems share a common reference model: the model which originated the encoding language.

Semantic Web

In order to achieve GI compatibility (especially, among different information communities), it is also necessary to utilise Web semantics resorts. The most common one is the ability to *learn* the constructs structure of a given XML dataset from the its XML Schema (e.g. W3C XSD) or DTD.

Other important resorts cover the possibility to infer constructs meaning of XML documents (see [3]).

From data format heterogeneity to XML-dialects

In order to easily achieve structure and metadata integratability, a large number of XML-dialects have been introduced, for encoding or wrapping most of the existing geographic data format.

In a certain sense it is possible to consider that every well-used geospatial data format was translated or encapsulated into an XML-dialect, or will be in a short time.

Such trend stems from the fact that organisations *discovered* the importance of self-descriptive dataset, allowing applications to automatically manage dataset import/export (i.e. structure and metadata integratability).

Moreover, XML fits Web and http-based technologies.

Eventually, defining customised XML tags -and XML Schema or DTD- turned out to be relatively easy and pleasing. In a certain sense is similar to work out a personalised HTML page. Authoring tools are relatively cheap and powerful.

Some Examples

For example, the following table reports some XML-based representation or wrapping or exporting/importing specifications for some of the most common geospatial data format.

Data Format	XML-based rappresentation/wrapping
ArcView® Shape SHP	DIMAP
ArcIMS® data	ArcXML (AXL)
AutoCAD® Land Development Desktop 2i data	LandXML
AutoCAD® Map 2000 data	LandXML
Autodesk MWF	Map Window XML (MWX)
DXF	DesignXML
DWG	DesignXML
CDF	CDFML
NetCDF	NetCDF XML, NcML
HDF	ESML
HDF5	H5gen XML
XDF	FITSML
GeoTIFF	GeoTIFF.Net, OGDI
VRF	OGDI
SPOT data	DIMAP
VEGETATION imagery	DIMAP
ERS imagery	DIMAP

Radarsat imagery	DIMAP
Landsat imagery	DIMAP
DXF	DIMAP
MIF/MID	DIMAP
Binary imagery	DIMAP, ESML
ASCII imagery	DIMAP, ESML
……..	….
……..	…..

It is easy to discover several others, and this trend is getting more and more popular.

Moreover, the previously introduced general-purpose encoding languages for geospatial datasets (i.e. GML, ISO 19118, G-XML) can be utilised for encoding several of the existing GI data formats.

It is possible to say that the heterogeneity concerning data format is now present in XML encoding, but there are at least four main improvements:

1. the syntax is unique (i.e. XML-dialects are all XML)

2. the dataset structure description is available along the dataset (DTD or XML Schema describes the structure of each dialect);

3. XML is a machine independent encoding and fits in with Web technologies;

4. XML is human readable.

Thanks to its unique syntax asset, applications are generally able to automatically import XML-dialects -simply parsing them.

Smart applications are able to read the associated XML Schemas –or DTDs- *learning* the structure and some of the meaning of XML-documents.

Integratability Vs Compatibility

XML-dialect heterogeneity is here to stay, and therefore it is better to develop compatibility at a higher level than data structure: the model level.

This is the way explored by the OMG Model-Driven Architecture initiative, and by OGC and ISO TC211, for general computational interoperability and for GI interoperability, respectively.

Therefore, the true compatibility should happen at the application model level; therefore, inter-operating systems must be able to sort out dataset semantic integratability issue (i.e. to understand and re-unify the meaning of dataset structures).

Such task can be achieved in a dynamic way (e.g. inferring semantics from XML Schemas, or from another XML meaning-describing document), or in a static way (e.g. providing the system with a configuration file for each *imported* XML-dialect). The first solution works out a compatibility system, while the second one implements modelling integratability.

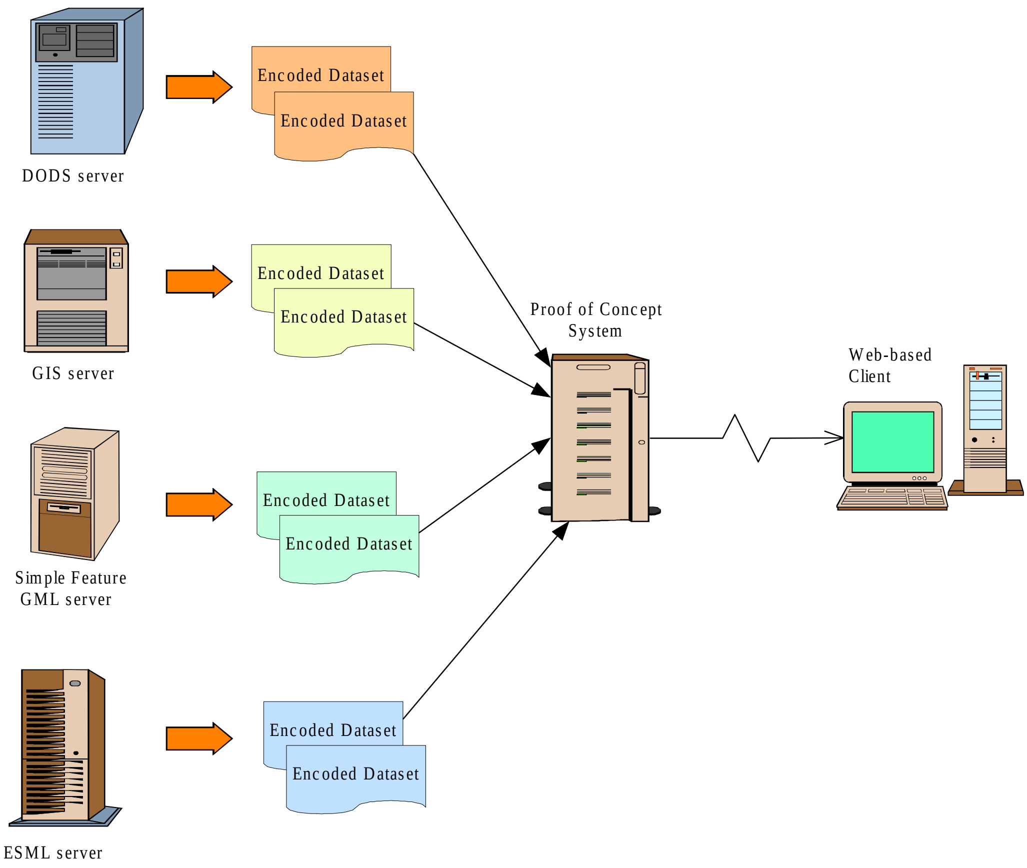
In any case the dataset structure level must be restrained as less as possible. More open the system is to XML heterogeneous dialects (i.e. structures), more information can be assimilated for satisfying system clients (e.g. users or other systems).

In order to achieve such ambitious goal, a system should be able to work at the meaning level –not only structure- of imported XML documents. Such capability allows the system to automatically interface virtually any XML-dialect which expresses a common meaning (see [3]).

The Proof of Concepts System Architecture

The following picture shows the general architecture of the proposed proof of concepts system. In particular, the depicted servers provide dataset (i.e. XML documents) in an off-line way.

The interaction between the system and the web-based client is implemented by a common Internet session.



Proof of concepts system architecture

This architecture fits the main objective of the proposed proof of concepts system: to test the feasibility of the proposed modelling integratability solution [5].

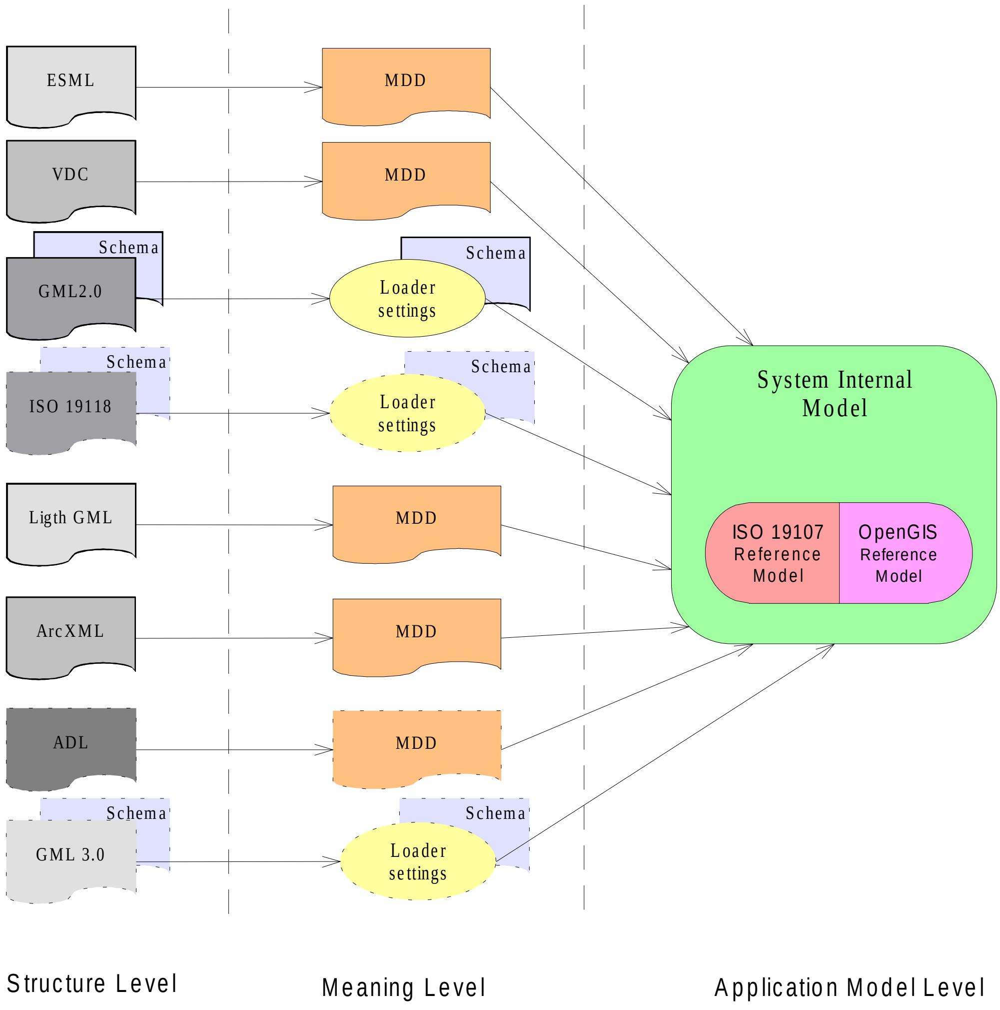
Modelling Integratability Solution

GI Metadata

The following schema shows the proposed computational solution for modelling integratability, as far as GI is concerned.

In particular, 3 information levels are considered:

1. The structure level: it is related to the representation of information;
2. The meaning level: it is related to the semantics of information;
3. The application level: it is related to the conceptual modelling of information.



Integratability solution

Here MDD stands for Meaning Definition Document, and can be any kind of XML-dialect document introduced to encode the *meaning* of the XML elements reported in the dataset. Actually, there is already an XML-dialect -called MDL- whose purpose is to capture XML elements meaning (see [3]); but here the concepts is intended in a more general sense.

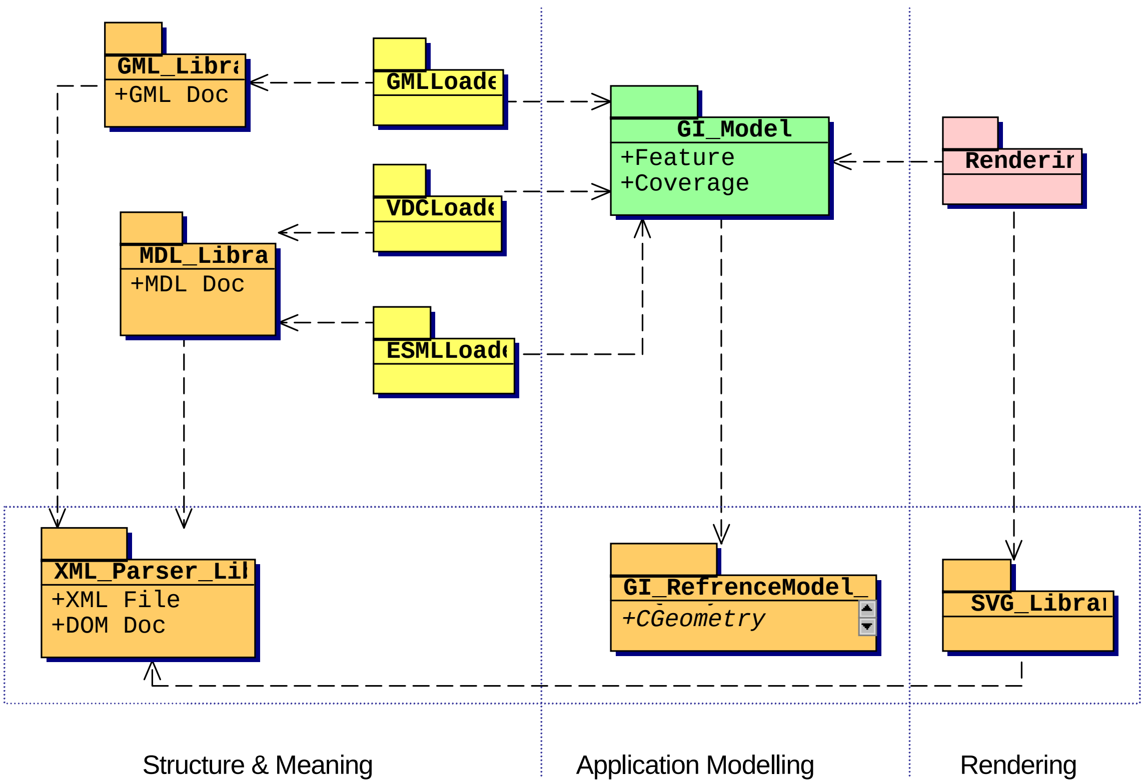
All the structures reported are XML-based, as well as MDDs which provide Loader modules with the necessary knowledge to *understand* the meaning of well-formed XML elements and project them to the THREDDS internal model.

Besides, several structures must be imported without the help of a MDD; it is mainly because the structures are either very complex or are not constrained by the general XML schema associated with the XML-dialect they are encoded in. In such cases the structures types are fully described in a more specific XML schema which varies from application to application. Therefore, the Loader module must be able to read both XML Schemas and infer from them the meaning of dataset structures.

It is at the Application Model level that the real compatibility occurs. That is possible since we assumed that the Internal Model is compliant with a general geospatial reference model (i.e. OpenGIS and ISO 19107, see [6]).

System Computational Model

The computational architecture of the proof of concepts system is showed in the following schema.



Libraries packages (the orange ones) are freely available libraries; they represent the enabling software on the top of which the system is implemented.

The *Loader* packages (i.e. the yellow ones) are packages to be developed in order to link the Structure and Meaning level to the Application Modelling level.

The GI Model package (i.e. the green one) implements the GI application model of the system (see [5]); it is the key component to be developed in order to carry out the modelling compatibility.

The Rendering package (i.e. the pink one) is a package to be developed in order to render application business objects (i.e. geospatial features and coverages). Several rendering formats and technologies can be adopted for such task, the SVG format was chosen as it is a well-promising XML-based technologies; furthermore, they are available free libraries to encode/decode SVG documents, and web-based clients to display SVG documents.

Naturally, it is always possible and useful to consider other rendering solutions.

It is important to clearly separate GI data modelling and rendering levels.

References

[1] David Flater, “Impact of Model-Driven Standards” OMG Model Driven Architecture, 2001.

[2] Sridhar Iyengar, “Model Driven Architecture (MDA) meets Web Services”, OMG Model Driven Architecture, 2001.

[3] K. Ahmed, et al., “XML Meta Data“, WROX Press Ltd., 2001.

[4] THREDDS Project Home Page, http://www.unidata.ucar.edu/projects/THREDDS.

[5] S. Nativi “Information Interoperability and Application Model”, http://www.unidata.ucar.edu/projects/THREDDS/Nativi/ApplicationModel/ApplicationModelFrameset.htm.

[6] S. Nativi, “Standardisation Framework for Geographical Information Resources”, http://www.unidata.ucar.edu/projects/THREDDS/Nativi/Standards/StandardsFrameset.htm.