



Metadata State of Play

Compliance Testing and Interoperability Checking

S. Viganó
TXT e-solutions

M. Millot
European Commission DG JRC,
Institute for Environment and Sustainability

EUR 23036 EN - 2007

The mission of the Institute for Environment and Sustainability is to provide scientific-technical support to the European Union's Policies for the protection and sustainable development of the European and global environment.

European Commission
Joint Research Centre
Institute for Environment and Sustainability

Contact information

Michel Millot

Address:
European Commission, Joint Research Centre,
Institute for Environment and Sustainability
Spatial Data Infrastructure Unit
T.P. 262
Via E. Fermi, 2749
I-21027 Ispra (VA)
ITALY

E-mail: michel.millot@jrc.it
Tel.: +39-0332786146
Fax: +39-0332786325

<http://ies.jrc.ec.europa.eu>
<http://www.jrc.ec.europa.eu>

Stefano Viganó

Address:
TXT e-solutions
Via Frigia, 27
I-20126 Milano (MI)
ITALY

E-Mail: stefano.vigano@txt.it
Tel. +39-02257711
Fax.+39-022578994

<http://www.txtgroup.com/it/index.shtml>

Legal Notice

Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use which might be made of this publication.

***Europe Direct is a service to help you find answers
to your questions about the European Union***

**Freephone number (*):
00 800 6 7 8 9 10 11**

(*) Certain mobile telephone operators do not allow access to 00 800 numbers or these calls may be billed.

A great deal of additional information on the European Union is available on the Internet. It can be accessed through the Europa server <http://europa.eu/>

JRC 41737

EUR 23036 EN
ISBN 978-92-79-07690-9
ISSN 1018-5593
DOI 10.2788/53688

Luxembourg: Office for Official Publications of the European Communities

© European Communities, 2007

Reproduction is authorised provided the source is acknowledged

Printed in Italy

Table of Contents

1. Introduction.....	5
2. Purpose and Scope of the Document	6
3. Normative Reference	7
3.1 Normative Background	7
4. Requirements Synthesis	11
5. Conformance.....	15
5.1 ESRI - Commercial.....	16
5.1.1 Introduction.....	16
5.1.2 Product Overview	17
5.1.3 Data Sheet	18
5.2 Metadata Parser (MP) - Freeware	19
5.2.1 Introduction.....	19
5.2.2 Product Overview	19
5.2.3 Language support.....	22
5.2.4 Data Sheet	23
5.3 ISO Metadata Editor 4.0 (IME) - Freeware	24
5.3.1 Product Overview	24
5.3.2 Data Sheet	25
5.4 Java Metadata Interface - Commercial	26
5.4.1 Product Overview	26
5.4.2 XML Validation.....	26
5.4.3 Data Sheet	27
5.5 Oracle XML Schema Validation - Commercial	28
5.5.1 Product Overview	28
5.5.2 Data Sheet	29
5.6 HiSoftware Products: TagGen® Office 2005 - Commercial.....	30
5.6.1 Product Overview	30
5.7 CatMDEdit – Free (GNU).....	32
5.7.1 Data Sheet	32
5.8 SIDIGER Project	33
5.8.1 Product Overview	33
5.9 Other tools.....	34
5.9.1 MetaScribe: A New Tool for Metadata Generation.....	34
6. Critical Review	35
7. Test Cases	36
7.1 ISO Metadata Editor 4.0 (IME)	36
7.2 Oracle XML sample.....	38
7.3 Java Metadata Interface	42
8. Conclusions.....	45
Annex A: Terms, Definitions and Abbreviations	46
Annex B: Inspire References	47
Implementation phases and milestones.....	48
Annex C: Bibliography	53

1. Introduction

The rapid increase of spatial data, proprietary software and the implementation of location based technologies to many modern industrial and business activities has created a need to maximise the open exchange of information between geospatial enabled parties. Since the mid nineteen nineties a whole series of international, regional and national spatial data infrastructures (SDI) have been established.

The DIRECTIVE 2007/2/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 14 March 2007 establishes an Infrastructure for Spatial Information in the European Community (INSPIRE)

The overall aim of the INSPIRE proposal is to improve the way in which spatial data held by public authorities supports environmental policy, by improving the harmonisation of spatial data and the interoperability of spatial services and ensuring greater sharing of the data between public authorities and on-line access by the public.

The Directive requires Member States to take a set of measures, ranging from the type of data shared between users and application up to the network of services that allow searching for, viewing and accessing these data.

The present study focuses mainly on some of these measures, and precisely, on certain types of information regarding the data. This information about data is called *metadata*.

Since metadata can be used to semantically describe other information, the INSPIRE programme has defined exactly what type of description will be required for compliancy (see Inspire Metadata Implementing Rule), that is:

- Metadata on metadata is needed to know what the status is of the metadata itself;
- Discovery metadata that can be implemented at two levels.
 - Level 1 metadata ensures the needed linked between the expert and the non-expert users.
 - Level 2 metadata are detailed enough for high-level discovery of spatial resources by expert users.

In addition, metadata could serve other purposes such as:

- Evaluation: it includes level 2 metadata and a set of metadata elements needed to evaluate the fitness for use of the spatial resource. The INSPIRE evaluation metadata elements is a minimum set to be extended by the different INSPIRE user communities to adapt the evaluation metadata to the resource specificity. The evaluation metadata is supported by ISO 19115 and ISO 19119, and possible user community extension of ISO 19115 and ISO 19119.
- Use: it involves complementary standards such as ISO 19110, ISO 19111, ISO 19117

2. Purpose and Scope of the Document

In the first chapter the document introduces the object of this survey, i.e. the spatial data metadata, with respect to the European Community INSPIRE programme.

After a general description of what metadata is up to this day and what can be used for in the GIS commercial and not profit field, the document then describes the INSPIRE requirements for compliancy. The third chapter therefore describes the normative background related to standards.

The document thereafter (chapter 5) focuses the subject on the present market; the existing software packages and solutions are taken into account with an analytic approach.

The chapter 6 tries to review the metadata state of play under a critical point of view, underlying what is present and what is missing.

In the following chapter the author describes a few test cases related to some of the software described in chapter 5.

The document finally closes with reference annexes for the used terminology, bibliography and the INSPIRE requirements.

3. Normative Reference

3.1 Normative Background

The main international organisations that look after the definition of the common rules for the geographic information are:

- ISO/TC211
- CEN/TC287
- Open Geospatial Consortium
- INSPIRE
- DGIWG
- Dublin Core
- W3C

The TC211 is the ISO Technical Committee that from 1994 takes care of standards for the geomatic and geographic information. At present, the published standards (IS) of the 19100 series are about 30. More standards are being defined or going to be approved.

The TC287 is the Technical Committee sibling of the TC211 at the CEN level, that is to say in Europe. At present, in the TC287 there is only one working group, the Working Group (WG) 5 “Spatial Data Infrastructure”.

The Open Geospatial Consortium (OGC – known also as the OpenGeospatial Consortium) is an international no-profit organisation, based on the voluntary consent, that looks after the definition of the technical specifications for geospatial and localisation (location based) services. OGC works through programmes defined by own members, in tight contact with governs, private firms and the university in order to implement interfaces of GIS software applications with an open and extensible architecture. These technical specifications defined by OGC are public (AS) and available free-of-charge.

The Digital Geospatial Information Working Group (DGIWG) has been created in 1985, in the military context, in order to technically support the digital geospatial information exchange. DGIWG develops and manages a set of standards for the digital geospatial information, with the object of accessing, exchanging and usage of geographic data among defence organisations of the member countries. DGIWG works in collaboration with ISO/TC211 and other international organisations specialised with geographic information. Only the national agencies of the member countries can participate to DGIWG. Now the main task of DGIWG (in collaboration with ISO and other standards’ organisations and further with NATO) is the definition of data models rather than data itself. These models can be described like exchange formats or application interfaces.

The Dublin Core initiative was born in the OCLC context (On line Computer Library Center), the U.S.A. libraries community that gather together authors, editors and owners of rights over documents on the network, in order to give the right tools for accessing the digital resources. This initiative suggested a minimal set of descriptive elements that could be given by the author or the publisher of a digital resource. The thereafter-constituted consortium has thus developed an architecture for metadata that could satisfy the requirements of information authors and sellers. This minimal set (proposed on December 1996) was made of fifteen base elements and it has been now extended to derived elements or qualifiers; this standard is still in progress but the initial core of 15 elements has never been changed. Each element has been defined with a set of ten attributes

compliant to the ISO/IEC 11179 standard. The minimal set of 15 elements has been defined as the SO 15836 standard in 2003 under the impulse of the U.S.A. standardisation agency (National Information Standards Organization). This promotion is an important and official acknowledgment to the usage of the Dublin Core standard, already translated in more than 20 languages and used all over the world to integrate information of different types.

The World Wide Web Consortium (W3C) develops technologies that ensure the interoperability (specifications, guidelines, software and applications) to drive Internet to the high most potential level, working as a forum for information, communications and common activities.

INSPIRE (INfrastructure for SPatial InfoRmation in Europe)

The Inspire programme has referred to the following international standards:

- ISO 19115 Metadata
- ISO 19139 Metadata XML Schema Implementation

ISO 19115:2003 → Semantic Interoperability

- Designed:
 - to support geographic information;
 - to work with wider information technology standards and practices;
 - to support locate, evaluate, extract, employ;
 - to support datasets, series, features, attributes.....;
 - to serve the global community, in a multinational, multilanguage environment;
 - based on a foundation of national, regional, and special information community standards and experiences
- Developed through a rigorous, consensus ISO process
- Provides a foundation for national, regional, and global interoperability

ISO 19139 → Technical Interoperability

Geographic Information – Metadata XML Schema Implementation

- Defines geographic metadata XML encoding (gmdXML)
- Uses ISO 19118 rules to produce ISO 19139 XML from ISO 19115 UML Schema
- Identification of additional 19100 entities: 19103, 7, 8, 9, 18
- gmd XML encoding rules
- Takes advantage of GML and other XML encodings
- Clauses defining:
 - Cultural and linguistic support
 - multiple languages, etc
 - Metadata for transfer
 - Dataset and Aggregate dataset metadata
 - Codelists

ISO 19106 - Standardized Profiles

“...set of one or more base standards or sub-sets sets of base standards... that are necessary for accomplishing a particular function.”

Conformance to profile means conformance with base standard(s) (Conformance Level 1 is a strict conformance level)

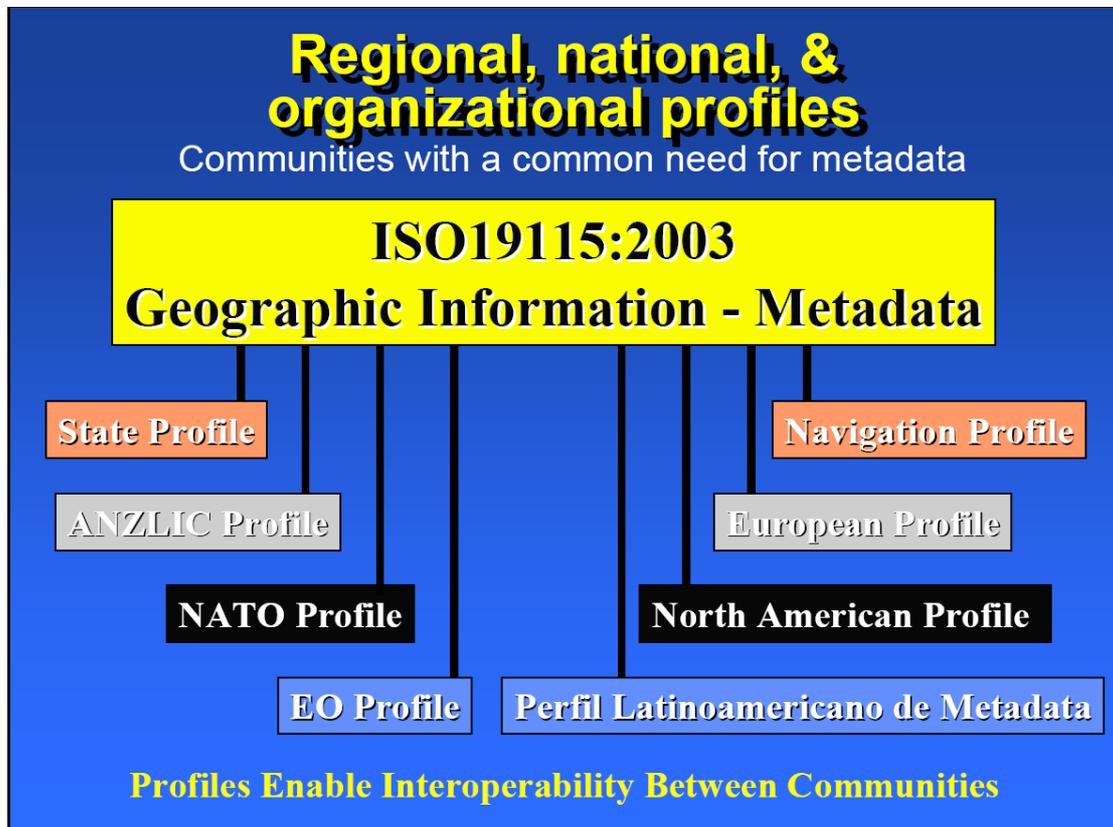


Figure 1: Regional, National and Organisational Profiles

Other profiles

- UK Gemini Profile (Geo-spatial Metadata Interoperability Initiative) AGI
- French Profile
- ISO19115 NAP (North American Profile)
- UNGIWG (UN Geospatial Information Working Group)

GI standardisation initiatives

- mainly 3 organisations:
 - ISO dealing with GI in ISO/TC211¹
 - CEN with TC287
 - OGC²

¹ www.isotc211.org

- standards and specifications to be used as reference material for DTs and CT

Open Geospatial Consortium

OGC in short is an international industry consortium of 333 companies, government agencies and universities participating in a consensus process to develop publicly available geoprocessing specifications. Open interfaces and protocols defined by OpenGIS Specifications support interoperable solutions that "geo-enable" the Web, wireless and location-based services, and mainstream IT, and empower technology developers to make complex spatial information and services accessible and useful with all kinds of applications.

GIS products are broken down by the OGC into two categories, based on how completely and accurately the software follows the OGC specifications.

Compliant Products are software products that comply with OGC's OpenGIS® Specifications. When a product has been tested and certified as compliant through the OGC Testing Program, the product is automatically registered as "compliant" on this site.

Implementing Products are software products that implement OpenGIS Specifications but have not yet passed a compliance test. Compliance tests are not available for all specifications. Developers can register their products as implementing draft or approved specifications, though OGC reserves the right to review and verify each entry.

² www.opengis.org

4. Requirements Synthesis

The section below is an excerpt of the Chapter 4 of the first version of the “Draft Implementing Rules for Metadata”.

Different levels of requirements are expressed concerning the availability of metadata elements related to spatial resources:

- Metadata on metadata is needed to know what the status is of the metadata itself. It is described in the General IRs defined in chapter 6;
- Discovery metadata is composed of two levels. Level 1 discovery metadata ensures the needed linked between the expert and the non-expert users. Level 2 discovery metadata are detailed enough for high-level discovery of spatial resources by expert users.
- Evaluation metadata includes level 2 metadata and a set of metadata elements needed to evaluate the fitness for use of the spatial resource. The INSPIRE evaluation metadata elements is a minimum set to be extended by the different INSPIRE user communities to adapt the evaluation metadata to the resource specificity. The evaluation metadata is supported by ISO 19115 and ISO 19119, and possible user community extension of ISO 19115 and ISO 19119.
- Use metadata involves complementary standards such as ISO 19110, ISO 19111, ISO 19117 and others. In practice use metadata may often be found in documentation for applications rather than associated with data. The interrelations between the different level metadata needed to cover the INSPIRE Implementing Rules and the user communities metadata is illustrated in Figure 3.

The interrelations between the different level metadata needed to cover the INSPIRE Implementing Rules and the user communities metadata is illustrated in the Figure 2 (Use of ISO 19115 and ISO 19119).

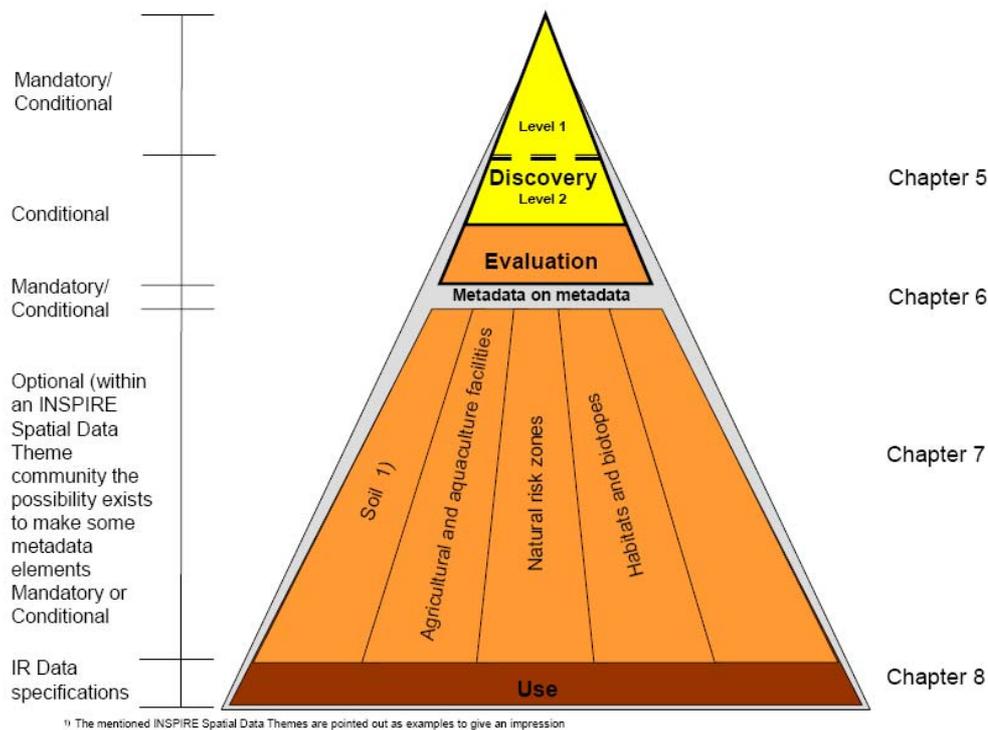


Figure 2: Metadata Levels for Inspire Program

Highlights

- Ensures the needed links between the expert and the non expert users (chapter 5 – discovery level 1)
- Metadata are detailed enough for high level discovery of spatial resources by expert users (chapter 5 – discovery level 2)
- Metadata elements needed to evaluate the fitness for use of the spatial resource (chapter 7).
- *Use* metadata involves complementary standards such as ISO 19110, ISO 19111, and ISO 19117 (chapter 8).

The section below is an excerpt of the Chapter 5 from the “Draft Implementing Rules for Metadata” INSPIRE official document.

Metadata for discovery serve the following purposes:

- to provide information about the resources a user is interested in
- to enable organisations to publish their data holdings

The discovery activity involves an interaction between a client application (e.g., a search engine) and a metadata repository. In this interaction, it is fundamental to consider:

- the user query expressed through the search interface of the search engine and provided in a form compatible with the metadata repository interface;
- the query analyser which evaluates each of the metadata sets of the metadata repository against the query criteria;

- the resulting metadata sets (i.e., the metadata sets matching the query criteria expressed by the user) as they are provided first by the repository interface to the result interface, then by the result interface to the user.

The manager of the metadata repository is fully responsible for the repository content model, which will not be discussed further. In the same way, the search engine providers are responsible for defining the user-side of the search interface and of the result interface.

Consequently, these IRs focus on:

- the content of the queries provided by the Search Interface to the Repository Interface;
- the core content of the resulting metadata sets provided by the Repository Interface to the Result Interface.

From this perspective, the metadata elements involved in the discovery of spatial resources can play two roles: metadata elements that constitute search-parameters and metadata elements whose main role is to describe a resource in response to a query.

This IR therefore distinguishes between two categories of metadata:

- Search metadata
They allow a user to formulate a query. They help specify what kind of data the user is looking for (e.g., which part of the world this data covers, and when the data was created). Search metadata will be a small set of metadata elements, which shall be supported by every metadata query tool and every metadata catalogue.
- Response metadata
They constitute a documentary feedback from the data/service provider, more fully describing a resource. Response metadata can be used for evaluation and to answer questions like:
 - What data and/or service are available within an area of interest?
 - Who is responsible for the resource and the associated metadata?
 - Who can I contact to get access, use and pricing of the resource?
 - Does the identified resource contain sufficient information to enable a sensible analysis to be made for my purposes?
 - What is the quality of the identified resource?
 - How and where can I obtain the resource?

Response metadata are documentation provided with a resource to ensure that users other than the data providers themselves use a resource correctly and wisely. They may also contain evaluation metadata that specify the necessary processes to obtain and use the data.

Response metadata help end users as well as data providers to document, store, reuse, maintain and archive a resource effectively.

This abstract metadata set for discovery distinguishes two levels of conformance for the response metadata elements:

- Discovery level 1 elements are metadata that provide basic and essential descriptions of a resource.
- Discovery level 2 elements are metadata that describe a resource in more detail. Level 2 elements are not essential for the first response. They give additional information about a resource as specified in Article 5 and 11 of the INSPIRE directive or are essential for an evaluation of the metadata information itself.

5. Conformance

A lookup of metadata tools and support tools on the web for existing software solutions has given different result according as the software application context taken into account.

Different grouping methods can be applied to the existing tools and solutions. Grouping criteria can refer for example to the involved metadata: like tools for GIS metadata rather than generic metadata.

The GIS Metadata tools then can be grouped according to the standards they are compliant to. A tool can be compliant for example to Dublin Core only, to FGDC or generically to ISO.

There is also a cross breakdown of tools and utilities in two main sections:

- freeware solutions, most of them developed and maintained by universities or development groups;
- Commercial solutions, usually developed by well-known software manufacturer (e.g. ESRI, or Intergraph).

The present chapter will adopt the last approach, looking for two or more solutions in the two main groups of freeware and commercial software. To make the look-up easier and faster, we have started with sites that keep an up to-date list of existing software.

The *Metadata Tools for Geospatial Data* is an independent metadata software evaluation page maintained by Hugh Philips and hosted by the Wisconsin Land Information Clearinghouse (WiscLINC). The *State Cartographer's Office* site (hosting the previous page), publishes a web page³ that summarizes the existing and more used tools used for documenting geospatial data and serving geospatial metadata. Most of these tools are compliant to the *Content Standards for Digital Geospatial Metadata* (CSDGM).

This page includes a list of available tools, giving for each one a general rating, the metadata standard to which the tool is compliant and the associated GIS platform. The page is updated regularly to include new tools and exclude the old ones. For this reason the list is a valid starting point for the evaluation of software solutions related to GIS metadata.

A second comprehensive web page⁴ about metadata tools can be found at the Federal Geographic Data Committee ([FGDC](#)) main site.

There is also a list of useful tools developed in the last years by [Intelec](#) and available in a dedicated web page⁵.

All these pages are a good starting point to look for solutions with given functionalities of metadata validation.

The following paragraphs will focus on a subset of the existing solutions, trying to analyze and evaluate the most common and successful ones.

³ <http://sco.wisc.edu/wisclinc/metatool/#TOOLBOX>

⁴ <http://www.fgdc.gov/metadata/geospatial-metadata-tools/?searchterm=SMMS%20review>

⁵ http://www.intelec.ca/technologie_a.html

5.1 ESRI - Commercial

The investigation has started from ESRI that is one of the most popular commercial brand in the GIS universe of software products. This brand develops GIS solutions addressing almost all the relevant requirements of GIS users and scenario. Detailed information of the ESRI products can be found on the main web home page⁶.

5.1.1 Introduction

ESRI (formerly Environmental Systems Research Institute) is a Principal member of Open Geospatial Consortium, Inc. (OGC) and active in the organization including their Board of Directors and Planning and Technical Committees. ESRI staff participation in OGC Programs, test beds, pilot projects, and specification initiatives includes:

- Standards, Interoperability, and Community Outreach and Adoption programs
- Chairing/Co-chairing many OGC Working Groups
- As editors of the Open Location Services (OpenLS) and OGC Simple Features specifications

ESRI fully encompasses Open Geospatial Consortium, Inc. (OGC) specifications and standards as well as comprehensive IT standards such as those related to ISO, W3C, ANSI, CEN, and many other leading de-facto industry standards. This includes adherence to and leadership work in areas such as XML, SOAP, SQL, etc.

ESRI actively participates in ISO TC 211. ESRI is able to represent the US in ISO TC 211 through its membership in ANSI-INCITS L1. ESRI participates both in the development of the standards and through using the standards in our products and services. ESRI has taken a leadership role in the development of the following standards:

- ISO 19115 Metadata
- ISO 19125 Simple Feature Access - Parts 1 & 2
- ISO 19139 Metadata XML Schema Implementation
- ISO 19115-2 Metadata - Extension for imagery

Supported ISO/TC 211 International Standards:

- ISO/TC 211 Geographic Information Standards ESRI's Implementation
- ISO 19106:2004—Profiles ESRI is using the concepts in this standard to produce user community profiles of metadata and the simple features GML.
- ISO 19107:2003—Spatial Schema ESRI is using the concepts in this standard in the development of ISO 19125 and ISO 19115; basic concepts defined in this standard are implemented in ArcGIS and in the design of geodatabases; this is the foundation for simple features GML.
- ISO 19108:2003—Temporal Schema ESRI implements the concepts in this standard in metadata and wherever temporal information is appropriate.

⁶ <http://www.esri.com>

- ISO 19109:2005—Rules for application schema. Using the concepts described in this standard, ESRI defines application schemas using a Conceptual Schema Language (e.g., UML). The ArcGIS Data Interoperability extension applies the concepts of mapping from one application schema to another for data transfer as defined in the standard. ESRI also uses the same concepts as the general feature model (GFM) as defined in the standard.
- ISO 19110:2005—Methodology for Feature Cataloguing ESRI is participating in several standards organizations to develop an encoding for feature catalogues based on the concepts in this standard. Encoded catalogues can be used as additional metadata.
- ISO 19111:2003—Spatial Referencing by Coordinates ESRI implements the basic concepts defined in this standard through GML as well as participates on the project team for the development of the new amendment for this standard.
- ISO 19112:2003—Spatial Referencing by Geographic Identifiers ESRI uses the concepts defined in this standard in the implementation of gazetteers as well as wherever spatial referencing by geographic identifiers is used.
- ISO 19113:2002—Quality Principles ESRI implements the concepts in these standards in Production Line Tool Set (PLTS) for ArcGIS and in database production services.
- ISO 19114:2003—Quality Evaluation Procedures
- ISO 19115:2003—Metadata ESRI implements this standard with ArcIMS Metadata Server—9.1, GIS Portal Toolkit 2.0, and in ArcGIS Desktop. ESRI played a lead role in supporting the development of this standard.
- ISO 19118:2005—Encoding ESRI is using the concepts defined in this standard in the development of SF-GML and ISO 19139.
- ISO 19119:2005—Services ESRI is using the concepts defined in this standard in its implementation of the OGC W*S specifications and is testing implementation of their use in developing OGC profiles for Catalog Services—Web (CS—W).
- ISO 19123:2005—Schema for coverage geometry ESRI uses the concepts defined in this standard for exchanging and interfacing with several raster, matrix, and tin structures.
- ISO 19125:2004—Simple Feature Access—Parts 1–2 ESRI implements this standard with ArcSDE and ArcGIS Desktop. ESRI played a lead role in supporting the development of this standard.

5.1.2 Product Overview

We have taken into account the ArcGis Desktop 9.2 release.

ArcGIS desktop products share the same core applications (ArcMap and ArcCatalog), the same user interface and development environment. This suite is made of different products and we have focused on the optional extensions that a final user can add to them to perform extended tasks.

ArcGIS Desktop Extensions allow you to perform tasks such as raster geoprocessing, three-dimensional visualization, and geostatistical analysis.

In detail, there are some tools that can be used to validate metadata against a model and to generate XML output conformant to required specifications.

The following tools are available within the geoprocessing Conversion Toolbox of the ArcMap 9.2 version:

- Metadata Translator: converts metadata from one standard's format to another using the ArcGIS metadata translation engine.

- XSLT Transformation: uses an XML Parser to transform an XML metadata document using an XSLT style sheet and export the result to a standalone file.
- Metadata Publisher: publishes an XML metadata document to a metadata catalog.
- USGS MP Metadata Translator: validates FGDC CSDGM metadata content and exports the content to standalone files using the USGS metadata parser utility, mp

A full description of the listed tools is available at the product web page⁷.

The *Metadata Translator* converts metadata from one standard's format to another using the ArcGIS metadata translation engine. For example, metadata can be exported in ISO 19139 format. This translation engine can be configured to support any metadata standard or profile.

The ArcGIS metadata translation engine can export metadata content from ArcCatalog to a standalone XML file. For example, the exported file might contain the same content but formatted to follow a different metadata standard such as ISO standard 19139, "Geographic Information — Metadata — Implementation Specification".

5.1.3 Data Sheet

Name/Version Reviewed	ArcGis Version 9.2
Date and Version of last known release	April, 2007 – version 9.2
Other relevant information and tips	
Function	A set of utilities to validate metadata against the ISO 19139 specifications.

⁷ [http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=An overview of the Metadata toolset](http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=An%20overview%20of%20the%20Metadata%20toolset)

5.2 Metadata Parser (MP) - Freeware

A second choice for a common software tool that satisfies the metadata validation requirements is the freeware *Metadata Parser*. Given the free-from-cost nature of this tool, there is not an official web page with the product detailed information. The tool is referenced in different GIS-related sites and the best link to start from is the web page⁸ maintained of Peter Schweitzer who is also the author of this tool.

5.2.1 Introduction

Peter Schweitzer is a geologist with a rich curriculum in the field of presentations about metadata. These presentations focus on scientific data management and the concomitant problems faced by organizations that seek to make scientific information available to the public.

You can find a comprehensive list of them at the web page specified in the footnote below⁹.

The author has worked on different tools, all free and, predictably, all provisional.

The *Metadata Parser* (M.P. for short) is one of these tools: the tool parses formal metadata, checking syntax in detail, and generating text, two forms of HTML, SGML, XML, and DIF. There is a version for UNIX and one for Microsoft Windows.

5.2.2 Product Overview

This program is a compiler to parse formal metadata, a utility which checks hierarchically indented text metadata (or SGML metadata, if it is 'perfect') against the CSDGM (Content Standard for Digital Geospatial Metadata). The error report flags deviations from the production rules for the CSDGM and element values not allowed in the domain of each element.

In addition to producing a hierarchically indented (2 space indent) text output, it also produces a standard presentation HTML output with a 'table of contents,' an HTML presentation of the metadata in the form of questions and answers (FAQ style), an SGML output, an XML output, and a DIF (Directory Interchange Format) output.

If the metadata does satisfy the CSDGM, but does not use hierarchical indentation to indicate parent/child relationships in the metadata (e.g. numbers are used instead), M.P. will choke on it. However, that same metadata may be effectively pre-processed with CNS, after which it may pass M.P. successfully.

The operational characteristics (input/output) of this tool are controlled through a configuration file. How to customize the configuration file is described in the tool's metadata and also (in somewhat more detail) through a web page devoted to the configuration file. Some notable items which can be controlled through the configuration file include: updating a metadata file standard and production of Dublin Core elements in META tags in the HEAD of output HTML documents.

The program generates output suitable for viewing with a web browser or text editor. Configurable output formats, accepts and passes local extensions to the metadata standard if they are defined in a user supplied extension file.

⁸ <http://geology.usgs.gov/peter/>

⁹ <http://geo-nsdi.er.usgs.gov/talk/>

It runs on Linux and UNIX systems and on PC's running all versions of Microsoft Windows (95 and up including XP). MP generates a textual report indicating errors in the metadata, primarily in the structure but also in the values of some of the scalar elements (that is, those whose values are restricted by the standard).

The compiler, its source code, executables for UNIX (Solaris and Linux) and Microsoft Windows, and its own formal metadata are available at the link in the footnote¹⁰.

A separate document shows the revision history of the tool.

The table 1 below displays a sample extracted from the tool Error Log:

```
mp 2.7.1 - Peter N. Schweitzer (U.S. Geological Survey)
Warning: ambiguous indentation at line 48
Warning (line 92): element Citation_Information found in textual value, some
information may be lost.
Upgrade (line 201): value of Metadata_Standard_Version set to FGDC-STD-001-1998
Error (line 48): Place is not permitted in Identification_Information
Error (line 46): Theme_Keyword_Thesaurus is not permitted in Keywords
Error (line 47): Theme_Keyword is not permitted in Keywords
Error (line 45): Theme is required in Keywords
Error (line 68): no element recognized in "Orginator: Wisconsin State
Cartographer's Office"; text is not permitted in Citation_Information
Error (line 67): Originator is required in Citation_Information
Error (line 91): no element recognized in "Source Citation:"; text is not
permitted in Source_Information
Error (line 90): Source_Citation is required in Source_Information
Error (line 91): element of unknown type has child (unknown)
Error (line 165): Online_Option is not permitted in Digital_Form
Error (line 164): Digital_Transfer_Option requires at least one Online_Option or
Offline_Option
Error (line 184): improper value for Metadata_Date
11 errors: 2 unrecognized, 4 misplaced, 4 missing, 1 bad_value
```

Table 1: MP Error Log

The symbols *cfile*, *ffile* and so forth represent the full names of files appropriate for the used operating system, and may include directory path separators (on Windows this is the backslash \ and on Unix the slash /).

Syntax error messages indicate the nature of discrepancies between the input file and the standard, and the line numbers of the relevant elements in the input file. If the command line parameter *-e efile* is not specified, syntax errors are written to the *stderr*, which is usually the console (for MS-DOS) or the terminal from which the compiler is launched.

5.2.2.1 Input

M.P. can read:

- Indented text

Since the FGDC Content Standard for Digital Geospatial Metadata, as the name implies, specifies only the contents of metadata files and not their encoding, it was necessary to

¹⁰ <http://geology.usgs.gov/tools/metadata/>

devise a specification for metadata encoding in order to develop and use this compiler. That encoding format is purely textual and the fidelity of the compiler to this format is fanatical.

- XML or SGML

The FGDC metadata standard was written before XML existed. At that time, SGML did exist but its use was not common outside some specific communities such as the publishing community (remember Ventura Publisher?). Since then, XML has become more widely accepted and supported, and mp was modified to read well-formed XML documents that use specified element names arranged in accord with the Document Type Definition.

Note: M.P. does not read word-processor documents; it only reads plain text, SGML, and XML files!

5.2.2.2 Output

Aside from the error report, output is generated only in the formats you specifically request (meaning that no other output is generated by default). Use the command-line options to specify the output file names and formats that you want.

Because the command line can become long and complex, you can simplify the process by using a configuration file.

5.2.2.3 Notes on output formats

Text output, if requested, follows the encoding format. This provides a check of the compiler; any such program should be able to reproduce its input without significant loss of information.

- HTML output, if requested, uses descriptive lists to arrange the elements hierarchically. The HEAD element of the metadata contains META elements corresponding to the Dublin Core.
- FAQ-style HTML output, if requested, uses the general arrangement of information found in Metadata in Plain Language and re-expresses the metadata in a manner that is easier to read. This format is not parseable in subsequent software processing. To see how M.P. writes standard metadata elements in this output format, consult a specially built input file, the configuration file used to process it with mp, and the FAQ-style output that M.P. generated for it.

Dublin core elements are added to the HEAD element as META tags.

NOTE: M.P. now provides in its HTML output a link to each of the other output formats that you requested when running mp. These links are relative to the current directory by default, and will work correctly when someone retrieves a metadata record directly through a web server. However, HTML metadata records retrieved through the Clearinghouse gateway interface come tagged with the URL of the gateway; consequently these links will not work by default with HTML records found through the gateway interface. To make these links work without regard to the retrieval method, place a BASE tag into the HEAD element of the output HTML code. As you might guess, M.P. can do this for you, but it needs to know the URL where your metadata will be available as web pages. It gets this information from a configuration file entry. To use a different configuration file, the `-c config_file` command

line option must be used, substituting for *config_file* the name of the actually used config file.

- SGML output uses the eight-character tags proposed by the FGDC Clearinghouse Working Group. The SGML output is designed to work with a SGML Document Type Definition (DTD) that I have developed and tested.
- XML output uses the eight-character tags given in the 1998 version of the CSDGM. The XML output is designed to work with a XML Document Type Definition (DTD) that I have developed and tested. NOTE: the DTD won't be displayed in the main browser window because it consists of DTD entries, all of which begin with < and end with >. The web browser thinks these are some weird kind of HTML tag, so it ignores all of them, leaving you with nothing. To see the DTD, choose Page Source from the View menu of the browser. Or if you choose Save As from the File menu, you can open the DTD in a text editor.
- Directory Interchange Format (DIF) output will require editing to fix inconsistencies between the DIF and FGDC metadata standards, and to add information required by DIF that is not clearly identified in the FGDC scheme, such as Entry_ID.

5.2.3 Language support

M.P. is able to interpret metadata element names written in several languages. To use this feature, include the -l option on the command line or the language option within the input section of the config file. In each case the language is specified using a 2-letter abbreviation from the following table:

- en - English (default)
- es - Spanish
- ca - Catalan
- id - Indonesian
- fr - French

Translations should be regarded as a work in progress. A separate table shows the current translation of element names in the various languages supported.

This tool can be tested by exploiting a web service available at the Geospatial Metadata Validation Service web page¹¹.

The *Geospatial Metadata Validation Service* checks a geospatial metadata record to be FGDC-compatible using mp (metadata parser). Provide a metadata record that is formatted as indented text, XML, or SGML. Results will show discrepancies between the structure of your metadata record and the expectation of the FGDC metadata standard (FGDC-STD-001-1998) and will show several potentially useful re-expressions of the metadata record itself.

¹¹ <http://geo-nsdi.er.usgs.gov/validate.php>

5.2.4 Data Sheet

Name/Version Reviewed	mp (Metadata Parser) Version 2.7.3
Date and Version of last known release	November 4, 2004 – version 2.8.18
Other relevant information and tips	Use the language switch (-l) on XML metadata to generate metadata with element names in languages other than English (Spanish, Indonesian). To update your 1994 CSGM metadata to 1998 CSDGM metadata with mp >> Modernizing Your Metadata, AKA: Dinner and a Metadata
Function	A utility to validate metadata against the CSDGM (1998) and convert it to standardized text, HTML, SGML, XML, or DIF form.

5.3 ISO Metadata Editor 4.0 (IME) - Freeware

A look up for more software candidates, with reference to the list described at the beginning of this chapter, has come up with the *ISO Metadata Editor (IME)*.

This tool is maintained by the Spanish National Institute for Aerospace Technology (INTA); references of the software and related documentation can be found at the INTA Earth Observation Department web page¹².

The tool functionalities are mainly devoted to the metadata editing but there is also a very important feature that makes this software potentially useful for the scope of this study: the validation against a metadata ISO schema.

5.3.1 Product Overview

Quoting the advertising exposed on the INTA site web pages, IME is “...*an application focused on make easier the understanding and working with ISO19115 and ISO19139 standards, and validate the interoperability of xml files metadata*”.

The software package includes the ISO19139 schema files that has been published in the l'Equipe D'Experts en Normalisation (EDEN) de l'IGN site¹³.

The software exposes the following features:

- Profiles edition and modification.
- Info about correct/incorrect metadata selection related with: specified classes, mandatory elements and entities with only one child allowed.
- Metadata location.
- Metadata paths identification.
- Metadata edition (XML attributes and elements values).
- Codelists options are showed.
- "Dummy" data fill in.
- Templates creation (IME format) to group global data.
- XML files opening and saving.
- XML validation with ISO19139 schema.
- XSLT file creation.
- HTML metadata file creation.
- External XML files validation with ISO19139.
- XSLT transformation applied over external XML files to obtain HTML.
- Multilingual profile.
- ISO19115 Corrigendum modifications included.

The functionalities that participate in metadata validation, according to the goals of this survey, have been highlighted in yellow.

IME can work with different schema files versions. Once a final release gets published, it will be only needed to update these schema files in the directory "schema19139" of the application.

¹² http://www.crepad.rcanaria.es/metadata/en/index_en.htm

¹³ <http://eden.ign.fr/xsd/isotc211>

The software provides guidance for metadata management, through the following steps:

1. Profile Definition (metadata subset, selected from the set referenced by the ISO19115 standard, that will allow us to cover the geographic information associated with our data)
2. Metadata Edition according to the data type defined by ISO19115 standard
3. XML file generation according to the ISO19139 schema file
4. HTML generation to facilitate the data visualization. An XSLT Transformation is required to make the XML to HTML conversion

The picture below displays an example of the validation step.

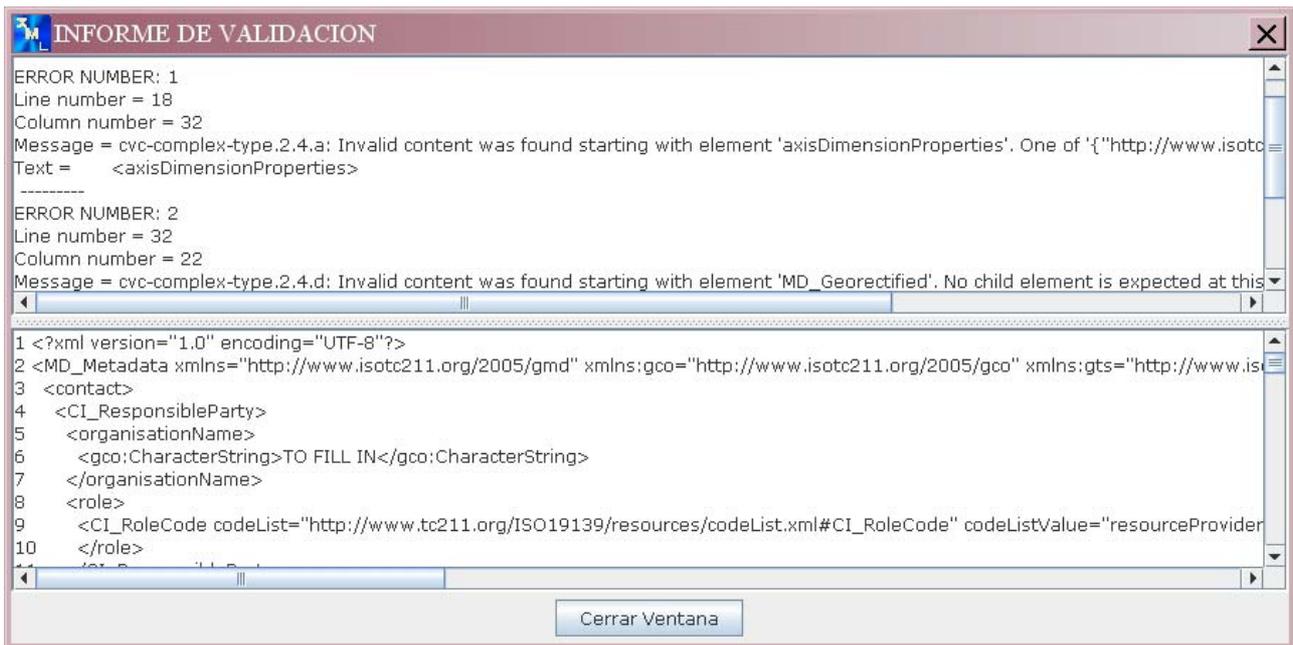


Figure 3: IME Validation Sample

5.3.2 Data Sheet

Name/Version Reviewed:	IME (ISO Metadata Editor) Version 4.0
Date and Version of last known release	October 14, 2006 – version 4.0
Other relevant information and tips	Requires JRE 5.0+ and MS Windows or Linux
Function	It manages ALL metadata information according to ISO 19115 requirements and using the ISO19139 schema files.

5.4 Java Metadata Interface - Commercial

While the solutions described in the previous paragraphs have been implemented as fully implemented software with given functionalities for metadata handling, this paragraph introduces a different approach to the problem under discussion: the usage of a developing environment. The present investigation has taken into account the Java Metadata Interface (JMI) and the new XML functionalities of the Standard Edition of the Java 2 Platform. Detailed information of this product can be found in the Java site¹⁴.

5.4.1 Product Overview

The JMI specification enables the implementation of a dynamic, platform-independent infrastructure to manage the creation, storage, access, discovery, and exchange of metadata. JMI is based on the Meta Object Facility (MOF) specification from the Object Management Group (OMG), an industry-endorsed standard for metadata management. The MOF standard consists of a set of basic modelling artifacts described using UML. Models of any kind of metadata (called metamodel) can be built up from these basic building blocks. JMI defines the standard Java interfaces to these modelling components, and thus enables platform-independent discovery and access of metadata. JMI allows for the discovery, query, access, and manipulation of metadata, either at design time or runtime. The semantics of any modelled system can be completely discovered and manipulated. JMI also provides for metamodel and metadata interchange via XML by using the industry standard XML Metadata Interchange (XMI) specification.

Some of the new features of the *Java 2 Platform, Standard Edition (J2SE) 5.0* release, code-named Tiger, are the added XML validation package and the *XPath* libraries. Before the Tiger release, the Java API for XML Processing (JAXP) SAXParser or DocumentBuilder classes were the primary instruments of Java technology XML validation. The new Validation API, however, decouples the validation of an XML document from the parsing of the document. Among other things, this architecture allows Java technology to support multiple schema languages. The new validation API has been implemented in the new Java API for XML Processing (JAXP) 1.3 package, available either as stand alone or already included in the J2SE 5.0 release.

More information about the JAXP 1.3 package can be found in the dedicated web page¹⁵.

5.4.2 XML Validation

JAXP 1.3 introduces a new schema-independent Validation Framework (called the Validation APIs). This new framework gives much more power to the application dealing with XML schema and can accomplish things that were not possible before. The new approach makes a fundamental shift in the way XML processing and validation are performed. Validation used to be considered an integral part of XML parsing, and previous versions of JAXP supported validation as a feature of an XML parser: a SAXParser or DocumentBuilder instance.

The new Validation APIs decouple the validation of an instance document as a process independent of parsing. This new approach has several advantages. Applications that rely heavily on XML schema can greatly improve the performance of schema validation. Perhaps more importantly,

¹⁴ <http://java.sun.com/products/jmi/>

¹⁵ <http://java.sun.com/developer/technicalArticles/xml/jaxp1-3/>

many previously unsolvable problems can now be solved in an efficient, easy, and secure way. Let's look at what you can do with the new Schema Validation Framework.

Validate XML Against Any Schema

Though JAXP 1.3 requires support only for W3C XML schema language, you can easily plug in support for other schema languages, such as RELAX NG. The Validation APIs provide a pluggability layer through which applications can provide specialized validation libraries supporting additional schema languages. This is achieved using a SchemaFactory class that is capable of locating implementations for the schema languages at runtime. The first step is to specify the schema language to be used and obtain the concrete factory implementation:

```
SchemaFactory sf = SchemaFactory.newInstance(<SCHEMA LANGUAGE>);  
<SCHEMA LANGUAGE> could be W3C XML Schema, Relax NG etc.
```

Table 2: SchemaFactory Instance Definition

If this function returns successfully, it means that an implementation capable of supporting specified schema language is available. Getting the SchemaFactory implementation is the entry point to the Validation APIs. This step goes through the pluggability mechanism that has long been at the core of JAXP. You can write the code in such a way that applications can switch between W3C XML Schema and RELAX NG validation without changing a single line of code.

For a detailed description of the software coding required to implement a validation functionality please refer to the paragraph 7.2.

5.4.3 Data Sheet

Name/Version Reviewed:	Java API for XML Processing (JAXP) 1.3
Date and Version of last known release	October, 2005 – version 1.3
Other relevant information and tips	Official documentation can be found at the Sun pages dedicated to the Java Jaxp technology.
Function	The Java API for XML Processing (JAXP) enables applications to parse and transform XML documents independent of a particular XML processing implementation

5.5 Oracle XML Schema Validation - Commercial

One more solution that can suit the current evaluation study is a database platform with XML support. Oracle 10g, along with the Microsoft SqlServer 2005 are the most popular DBMS among the commercial products while MySQL is a similar solution in the non-commercial market area.

Oracle is far the most known and used DBMS platform and the latest 10g version is a suit of products addressing the most different enterprise needs and demands. The present paragraph will focus on the *Oracle XML Developer's Kit*. Detailed information can be found at the Oracle web site¹⁶.

5.5.1 Product Overview

Oracle XML Developer's Kit 10g is a set of components, tools and utilities in Java, C, and C++ which is available in Oracle Database 10g, Oracle Application Server 10g and on OTN that eases the task of building and deploying XML-enabled applications with a commercial redistribution license.

Unlike many shareware and trial XML components, the production Oracle XDK provides the XML support that is proved in demanding enterprise production environments including Oracle SOA, Oracle JDeveloper/ADF, Oracle XML DB and Oracle XML Publisher. For Oracle customers, Oracle XDK is officially supported 24x7. Oracle XDK consists of the following components:

- **XML Parsers:** create and parse XML using DOM (including 3.0), SAX and JAXP interfaces. Directly access XMLType in the Oracle Database 10g with unified C DOM interfaces.
- **XSLT Processors:** transform or render XML. Now with XSLT 2.0 Java support.
- **XSLT VM:** and Compiler provides high performance C XSLT transformation engine using compiled stylesheets.
- **XML Schema Processors:** supports XML schema validation. Now includes validation interfaces for stream-based processing.
- **XML Java Beans:** parse, transform, diff, retrieve and compress XML documents via Java components.
- **XML Class Generator:** Now supporting JAXB. Automatically generates classes from DTDs and XML schemas to send XML from Web forms or applications.
- **XML SQL Utility:** generates XML documents, DTDs and XML schemas from SQL queries in Java and inserts XML documents into Oracle databases.
- **XSQL Servlet:** combines XML, SQL, and XSLT in the server to deliver dynamic web content and build sophisticated database-backed web sites and services.
- **XML Pipeline Processor:** invokes Java processes through XML control files.
- **TransX Utility:** makes it easier to load globalized seed data and messages into Oracle databases.

The XML Schema Validation can provide a flexible and portable form of data validation for use in applications. You may implement the XML validation process in your client-side or mid-tier applications, but if you want either:

¹⁶ <http://www.oracle.com/technology/tech/xml/xdkhome.html>

- Absolute control of data validation whenever the data is updated/inserted
- Make use of the data management capability of the Oracle database

This data validation process is implemented inside a trigger callable from PL/SQL procedures on the server-side. Since there is not a built-in PL/SQL API to do XML Schema validation, we can create one using Java Stored Procedures.

With the advent of XML Schema, there is the requirement to validate XML documents against their associated schemas. The following description gives an introduction to the XML schema validation process with the XDK for Java and discusses how to build an Oracle Java Stored Procedure to perform the schema validation on the server-side of the Oracle Database.

XML Schema Validation provides a flexible and portable form of data validation for applications. You can implement the XML validation process in either the client-side or the mid-tier applications, but if you want either:

- Absolute control of data validation whenever the data is updated/inserted
- Make use of the data management capability of the Oracle database

Then putting data validation process inside a trigger or your PL/SQL procedures on the server-side is a good solution. Since there is not a built-in PL/SQL API to do XML Schema validation, we can create one using Java Stored Procedure.

The first step in building a Java Stored Procedure for XML Schema validation is to download the components and decide the environment requirements. The required components are:

- XML Schema Processor for Java [xschema.jar]
- XML Parser for Java [xmlparserv2.jar]

Both of the components are included in Oracle XML Developer's Kit for Java. Please refer to the Resource section to download the latest version of XDK for Java. The Oracle Database (8.1.6 version and above) is also needed which fully support Java Stored Procedures. If you download the XDK for Java and have an Oracle 8.1.6 Database or above, you can follow the following steps to build up the Java Stored Procedure and take advantage of XML Schema for data validation:

- Create the Java Class with the Schema Validation Function
- Load and resolve the Java Class into Oracle Database
- Publish the Java by defining the specification

5.5.2 Data Sheet

Name/Version Reviewed:	Oracle 10g XML Developer's Kit
Date and Version of last known release	April, 2006 – version 10.2.0.2
Other relevant information and tips	Available for Java, C, and C++
Function	XML validation

5.6 HiSoftware Products: TagGen® Office 2005 - Commercial

TagGen® Office¹⁷ is a complete commercial software solution for developing and maintaining metadata schemas, creating meta tags, verifying compliance and creating a metadata index for the web enterprise solution.

The software has been developed by HiSoftware, a leading provider of software, services, and On-Demand solutions that test, repair, monitor and enforce audited Web content, quality, and regulatory compliance.

5.6.1 Product Overview

With the TagGen® Office solution it is possible to easily create and maintain metadata. In detail, the core functionalities are:

- Batch Capability
- Update 1 or 100 pages in seconds with the click of a button
- Protect your documents with specific intellectual property elements:
- Publisher author, company
- Copyright statement, and copyright tags
- Update site-specific default elements
- Expiration, Archive, Reply-to
- Custom controlled vocabularies
- Generate keywords for all your Web documents in your site
- Use batch to remove all named metatags from your documents if you want to start fresh!

This product also includes

- Interoperability Support Features
- Support for use of tokens "Z" technology
- Support for GILS Schema.
- Support for WAGILS Schema.
- Support for Dublin Core Schema.
- Dublin Core Editor supports implementation of full Dublin Core metatags.
- GILS Compatible.

More functionalities for the management of customized metadata and properties include:

- In-line Custom Field Editor creates drop-down menus or date format for your controlled vocabulary!
- Ability to hide tabs in the interface that you do not use- Create customized GUI!
- Type recognition for your custom fields
- The system ships with the standard recognized schemas: GILS, WAGILS, AU-GILS, Dublin Core, IMS, MS Extended, A-Core, NBII, and Publish
- TagGen includes a base template from which you can derive new schemas
- The Grid Base (like Microsoft(r) Excel(r) Spreadsheet) stores element names and allows you to set up system default values for the elements.

¹⁷ http://www.hisoftware.com/taggen_office/Index.html

- The user can create the custom schema and then protect the names of the elements from being overwritten while allowing the values to be edited.
- Language, Author, Classification and Custom Subject tag's Meta name can be customized
- The value cells automatically grow as needed during data entry so you can see everything that you are typing in.
- Unlimited custom support to meet all of your metadata needs!

Other Features:

- Add Page Properties
- Site Properties
- All required search engine metadata
- Import existing metadata automatically
- Custom lists and drop-down menus where frequent values are added to the list by default
- Keyword and Description character counter and the keyword density checker
- Save keywords
- Includes JITI Builder to Manage MS Office Document Metadata

Verification of HTML metadata is necessary for many reasons:

- Confirm compliance to internal policy
- Confirm compliance to global policy
- Confirm search engine readiness
- PICS/Child Safety Practices
- Confirm privacy statements.

TagGen® Office - Customization Features

With TagGen you can add customized metadata schemas for specific search engines, your corporate Intranet, or your Web site. The custom metadata schema generator allows you to easily create metadata element groupings or to derive a new group from an existing group. TagGen Office includes the ability to easily create dropdown menus and date formats to better manage your controlled vocabularies. TagGen Office also has the ability to hide tabs in the interface that you do not use, customizing your GUI!

Highlight: TagGen Office includes a Dublin Core Editor. In the technical sheet web page¹⁸ more detailed information can be found.

¹⁸ http://www.hisoftware.com/fact_sheetcc.htm

5.7 CatMDEdit – Free (GNU)

CatMDEdit has been developed by TeIDE¹⁹, a Spanish consortium constituted by the R&D groups of the University of Zaragoza (Computer Science and Systems Engineering Dept.), the Universitat Jaume I (Dept. of Information Systems), and the Polytechnic University of Madrid (Topography and Cartography Dept.). The application includes:

- Support for edition and visualisation of metadata entries in a catalog;
- A thesaurus management tool, allowing the management of thesauri supported by a thesauri database. The main functions of this tool are: creation/deletion/modification of thesauri; edition/visualisation of terms in a hierarchical and alphabetical structure; and import/export from/to text files in different formats;
- An XML Import/Export tool, enabling the exchange of metadata records in XML format conforming to different standards such as CSDGM, ISO19115 and Dublin Core. Besides, the tool also facilitates more readable presentations of metadata records in HTML format. The import process has two possibilities: it can either create a new metadata record, or update the content of a metadata record previously selected from the metadata repository;
- A Metadata Generation tool, enabling the semi-automatic generation of metadata for several types of resources. For instance, this tool is able to obtain descriptive information from ESRI shapefiles. Additionally, the tool can also extract metadata corresponding to the relational structure of tabular sources (e.g. Excel, Access, Oracle...);
- A Collection Metadata Edition Tool, enabling the edition and visualization of metadata describing collections of datasets (later called units) that can be considered as a unique entity. Some examples of such collections are spatial aggregations, like mosaics of aerial images or temporal series which aggregate geographic resources taken in similar conditions but at different instant times. The objective of this tool is to manage jointly the metadata at collection level (shared by the all the units in the collection) and the specific characteristics of each unit;
- A Metadata Validation Tool, facilitating the validation of metadata elements inserted by the user, i.e. checking the conditionality (optional, mandatory or mandatory-if-applicable) of metadata elements and reminding what mandatory elements have not been filled in;
- A Contact Management Tool, allowing reusing contact information (e.g. name, address, telephone...), which is needed in several metadata fields. Thanks to this tool, the contact information about a person is only inserted once and used whenever it is required;

5.7.1 Data Sheet

Name/Version Reviewed:	CatMDEdit 3.8.0
Date and Version of last known release	January, 11 2005 – version 3.8.0
Other relevant information and tips	
Function	Metadata editor tool that facilitates the documentation of resources, with special focus on the description of geographic information resources

¹⁹ <http://teide.unizar.es/> Please note that the specified linked page no longer exists.

5.8 SIDIGER Project

The present study has also considered existing solutions implemented in the context area of GIS communities, consortium or international organisations. The search on this area has come up with the tool developed within a INSPIRE pilot project²⁰ and detailed below. The tool furthermore is based upon an other open-source tool named CatMDEdit and also described in a dedicated paragraph.

5.8.1 Product Overview

SDIGER is a pilot project on the implementation of the Infrastructure for Spatial Information in Europe (INSPIRE). The main site Geo-Portal is comprehensive of information, services and documents

It was funded by the Statistical Office of The European Communities, that aims to demonstrate the feasibility and advantages of solutions for sharing spatial data and services proposed by the INSPIRE position papers and to estimate the costs of implementing interoperability-based solutions on the basis of real cases, together with the problems, obstacles which might be encountered during the subsequent large-scale implementation of INSPIRE.

SDIGER consists in the development of a Spatial Data Infrastructure (SDI) to support access to geographic information resources concerned with the Water Framework Directive (WFD) within an inter-administration and cross-border scenario that involves: two countries, France and Spain; and, the two main river basin districts at both sides of the border, the Adour-Garonne basin district and the Ebro river basin district.

Within the SIDIGER project, an open source metadata management tool with support to the aforementioned metadata profiles has been provided. This application is based on the *CatMDEdit* open source tool²¹.

The tool has been adjusted to the requirements of the SDIGER project in order to give support for the creation of metadata according to the metadata profiles defined for the project. In order to facilitate the use of the tool, three guidelines have been created to explain the cataloguing according to:

- [SDIGER - WFD metadata profile](#)
- [SDIGER - INSPIRE metadata profile](#)
- [SDIGER - Dublin Core application profile for Geographical Data Mining](#)

²⁰ <http://www.ideo.es/sdiger/>

²¹ <http://catmdedit.sourceforge.net/>

5.9 Other tools

5.9.1 MetaScribe: A New Tool for Metadata Generation

The NOAA Coastal Services Center's MetaScribe tool is designed to reduce significantly the effort required to produce metadata compliant with the Federal Geographic Data Committee (FGDC) Content Standard for Digital Geospatial Metadata (CSDGM). This software tool takes advantage of the fact that, within a collection of records for a given data type, the records are generally very similar in content, with only a few fields or phrases changing from one record to the next. MetaScribe delivers the greatest savings when used to create a collection of similar records. In fact, this tool is not appropriate for the creation of multiple metadata records with little redundant content.

MetaScribe is template driven (see example template). The user must create a metadata template, which is uploaded into MetaScribe. Building a template is not a trivial task; however, once a template is created for a given data type, the user can create multiple records quickly and easily.

MetaScribe uses two proofing tools to check the records it creates: *cns* (Chew and Spit) and *mp* (Metadata Parser). These programs, created by the U.S. Geological Survey (USGS), will report any errors within a record and require that those errors be fixed before a final metadata record is produced. This illustrates the importance of creating a proper template that will produce error-free records.

6. Critical Review

The software described in the paragraphs of Chapter 5 all satisfy in different ways the requirement stated at the starting of this study, which is the ability to perform a check on the metadata conformance and interoperability.

Each of this software, anyway, is somewhat limited in the functionalities exposed or in the implemented technology and a few considerations must be given to better understand these limits. As a matter of fact, before expressing any choice we should first consider our goals.

If we are generally looking for metadata validation and conformance to given schemas, then almost all the previous solutions can suit our needs. We can in fact use the software with some configuration or schema files in order to recognize the metadata structure we want to validate. These files are sometime distributed with the same product (e.g. the Dublin or ISO 19115 schemas), other times they must be created by the user.

In the first case the predefined configuration almost always addresses the most common standards, like ISO or FGDC. The limit is evident, since a user needs the correct configuration file to perform validation against a desired standard. Whenever a schema, required to validate against a standard, is not available, then the software cannot be of use at all.

The second case gives to users the ability to create home-made configurations using for example a base model. The strength of this case is the flexible usage of the validation tool, since the user has the opportunity to apply to new standards or to the evolution of existing ones. The weak side of this approach is the complex information underlying a metadata schema. The user must first know and then define the schema structure required by the validation process.

The metadata conformance to an unavailable schema requires therefore a full understanding of the standard and a good knowledge of the XML/XSD technology in order to fully exploit the software capabilities.

If, on the other hand, we look for software compliancy to a given metadata standard, then we find only few products in line with our requirement. Most of them address, inevitably, the American market and support therefore the FGDC standard. Looking at the existing scenario one can understand that this issue is not a minor detail, for the reason hereinafter described.

The Inspire European proposal aims to give a background of common interfaces and services to a market (in the most general meaning) that is still in its youth, even if in ferment and every day more rich of new actors and solutions. Europe must anyway face the competition of existing solutions (both in the standard and in the software areas) driven by the American market and originated from their requirements and demands.

It may therefore be difficult, for the European proposal to make the new metadata standard easily and quickly adopted by the software market. On the other hand there are some products described in chapter 4 (like oracle) that can be already used to implement legacy solutions for the new Inspire specifications.

7. Test Cases

For the required test cases the author has chosen a subset of the reviewed solutions out of the ones described, for the difficulties related to their availability and the base knowledge required to use them. The next paragraphs will present some of the possible testing scenarios.

7.1 ISO Metadata Editor 4.0 (IME)

The application has been downloaded from the CREPAD metadata web page²².

After the installation, the author has selected the Validate XML sub-function from the XML entry in the main menu (see figure 4).

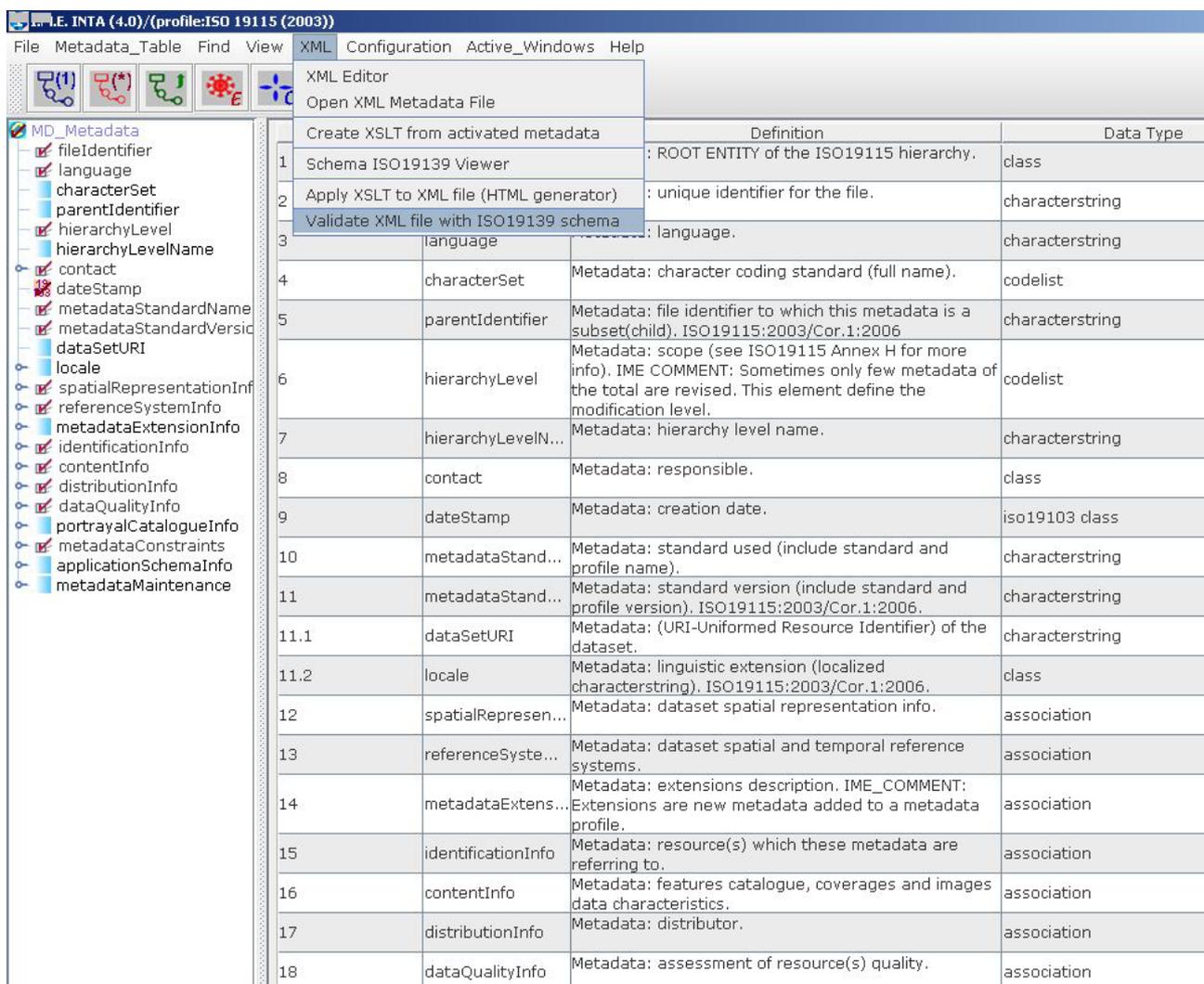


Figure 4: IME Metadata Validation

²² http://www.crepad.rcanaria.es/metadata/v4_0/ime_v4_0_setup.exe

The application thereafter opens a dialog (figure 5) where you must select

- the XSD file related to the chosen schema
- the XML file to be verified

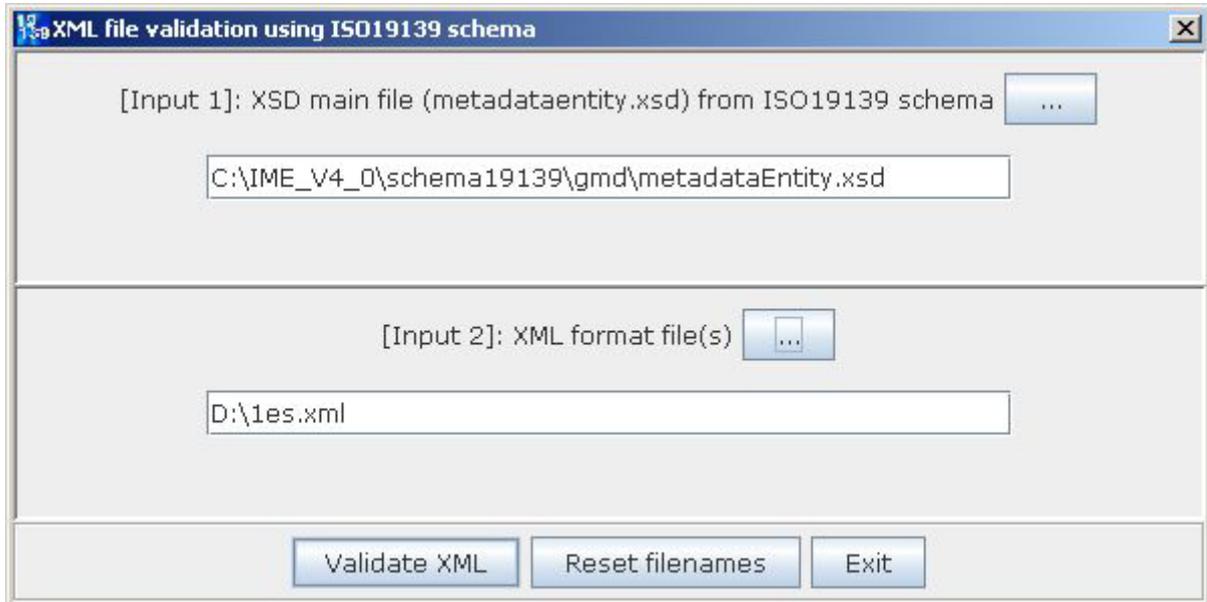


Figure 5: IME Validation with ISO19139 Schema

The validation results, on completion, are exposed in the new window where you can review the errors and perform the appropriate actions.

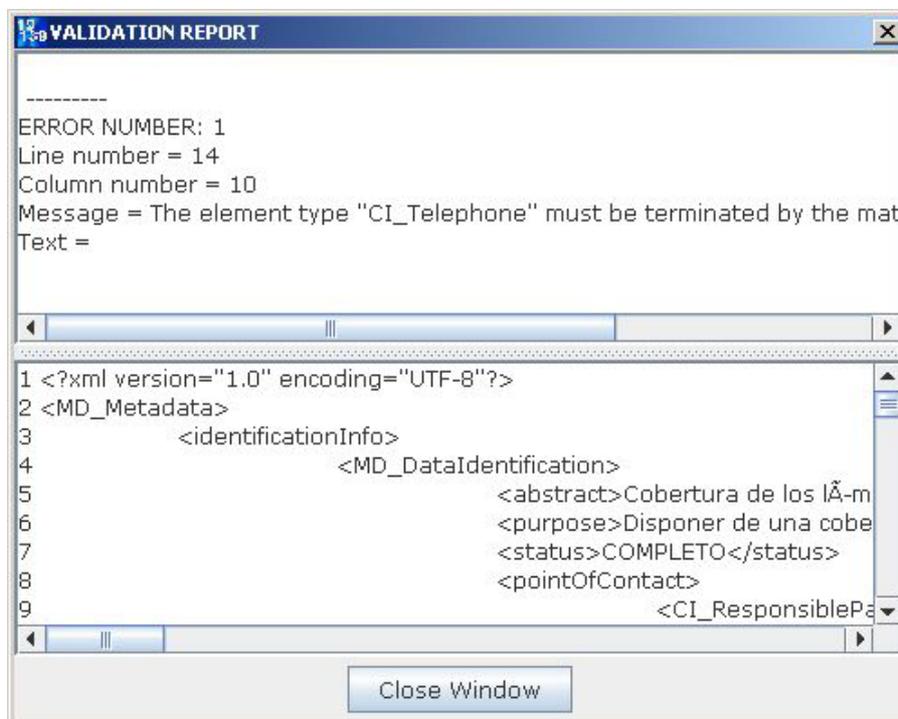


Figure 6: IME Validation Final Report

7.2 Oracle XML sample

The following example shows a way to perform xml validation with proper pl/sql procedures.

First we must create the database objects...

```
create table schema_tab(id number, xsd varchar2(4000));
create table xml_tab(id number, xml varchar2(4000));
```

Table 3: Table Creation

... and populate them with data.

```
insert into schema_tab values(1, '<schema targetNamespace="http://www.example.com/Report"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:r="http://www.example.com/Report"
  elementFormDefault="qualified"> <annotation>
<documentation xml:lang="en">
  Report schema for Example.com
  Copyright 2000 Example.com. All rights reserved.
</documentation>
</annotation> <element name="purchaseReport">
<complexType>
<sequence>
  <element name="regions" type="r:RegionsType">
    <keyref name="dummy2" refer="r:pNumKey">
      <selector xpath="r:zip/r:part"/>
      <field xpath="@number"/>
    </keyref>
  </element> <element name="parts" type="r:PartsType"/>
</sequence>
<attribute name="period" type="duration"/>
<attribute name="periodEnding" type="date"/>
</complexType> <unique name="dummy1">
  <selector xpath="r:regions/r:zip"/>
  <field xpath="@code"/>
</unique> <key name="pNumKey">
  <selector xpath="r:parts/r:part"/>
  <field xpath="@number"/>
</key>
</element> <complexType name="RegionsType">
<sequence>
  <element name="zip" maxOccurs="unbounded">
    <complexType>
    <sequence>
    <element name="part" maxOccurs="unbounded">
    <complexType>
    <complexContent>
    <restriction base="anyType">
    <attribute name="number" type="r:SKU"/>
    <attribute name="quantity" type="positiveInteger"/>
    </restriction>
    </complexContent>
    </complexType>
    </element>
    </sequence>
    <attribute name="code" type="positiveInteger"/>
    </complexType>
    </element>
    </sequence>
  </complexType> <simpleType name="SKU">
  <restriction base="string">
  <pattern value="\d{3}-[A-Z]{2}"/>
  </restriction>
</simpleType> <complexType name="PartsType">
<sequence>
  <element name="part" maxOccurs="unbounded">
    <complexType>
    <simpleContent>
```

```

        <extension base="string">
          <attribute name="number" type="r:SKU"/>
        </extension>
      </simpleContent>
    </complexType>
  </element>
</sequence>
</complexType></schema>
');

insert into xml_tab values(1,'<purchaseReport
xmlns="http://www.example.com/Report"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.example.com/Report report.xsd"
period="P3M" periodEnding="1999-12-31"> <regions>
<zip code="95819">
  <part number="872-AA" quantity="1"/>
  <part number="926-AA" quantity="1"/>
  <part number="833-AA" quantity="1"/>
  <part number="455-BX" quantity="1"/>
</zip>
<zip code="63143">
  <part number="455-BX" quantity="4"/>
</zip>
</regions> <parts>
<part number="872-AA">Lawnmower</part>
<part number="926-AA">Baby Monitor</part>
<part number="833-AA">Lapis Necklace</part>
<part number="455-BX">Sturdy Shelves</part>
</parts></purchaseReport>');

insert into xml_tab values(2,'<purchaseReport
xmlns="http://www.example.com/Report"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.example.com/Report report.xsd"
period="P3M" periodEnding="1999-11-31"> <regions>
<zip code="95819">
  <part number="872-AA" quantity="1"/>
  <part number="926-AA" quantity="1"/>
  <part number="833-AA" quantity="1"/>
  <part number="455-BX" quantity="1"/>
</zip>
<zip code="63143">
  <part number="455-BX" quantity="4"/>
  <part number="235-JD" quantity="3"/>
</zip>
</regions> <parts>
<part number="872-AA">Lawnmower</part>
<part number="926-AA">Baby Monitor</part>
<part number="833-AA">Lapis Necklace</part>
<part number="455-BX">Sturdy Shelves</part>
</parts></purchaseReport>');

```

Table 4: Populate Tables With XML Data

In table 5 (below) there is a sample Java code that implement the XML validation.

```
import oracle.xml.parser.schema.*;
import oracle.xml.parser.v2.*;
import java.net.*;
import java.io.*;
import org.w3c.dom.*;
import java.util.*;
import oracle.sql.CHAR;
import java.sql.SQLException;
public class SchemaUtil
{
    public static String validation(CHAR xml, CHAR xsd)
    throws Exception
    {
        //Build Schema Object
        XSDBuilder builder = new XSDBuilder();

        byte [] docbytes = xsd.getBytes();
        ByteArrayInputStream in = new ByteArrayInputStream(docbytes);
        XMLSchema schemadoc = (XMLSchema)builder.build(in,null);

        //Parse the input XML document with Schema Validation
        docbytes = xml.getBytes();
        in = new ByteArrayInputStream(docbytes);
        DOMParser dp = new DOMParser();

        // Set Schema Object for Validation
        dp.setXMLSchema(schemadoc);
        dp.setValidationMode(XMLParser.SCHEMA_VALIDATION);
        dp.setPreserveWhitespace (true);

        StringWriter sw = new StringWriter();
        dp.setErrorStream (new PrintWriter(sw));

        try
        {
            dp.parse (in);
            sw.write("The input XML parsed without errors.\n");
        }
        catch (XMLParseException pe)
        {
            sw.write("Parser Exception: " + pe.getMessage());
        }
        catch (Exception e)
        {
            sw.write("NonParserException: " + e.getMessage());
        }
        return sw.toString();
    }
}
```

Table 5: XML Validation Function

In the next piece of code there is the PL/SQL function that performs XML Schema validation

```
CREATE OR REPLACE FUNCTION SchemaValidation(xml IN VARCHAR2,xsd IN VARCHAR2)
return varchar2
IS LANGUAGE JAVA NAME
'SchemaUtil.validation(oracle.sql.CHAR,oracle.sql.CHAR) returns java.lang.String';
/
show errors;
```

Table 6: PL/SQL Function for Validation

Finally, there is the code required by the Java Stored Procedure to do XML Schema validation

```
-----  
-- Write XML Buffer to Output  
-----  
CREATE OR REPLACE PROCEDURE printBufferOut(xmlstr IN OUT NOCOPY VARCHAR2) AS  
  line   VARCHAR2(20000);  
  nlpos  INTEGER;  
  
BEGIN  
  LOOP  
    EXIT WHEN xmlstr is null;  
    nlpos := instr(xmlstr,chr(10));  
    line := substr(xmlstr,1,nlpos-1);  
    -- print line  
    IF(length(line) <250) THEN  
      dbms_output.put_line(' | '||line);  
    ELSE  
      dbms_output.put(' | ');  
      LOOP  
        EXIT WHEN line is null;  
        dbms_output.put_line(substr(line,1,250));  
        line := substr(line,250+1);  
      END loop;  
    END if;  
  
    xmlstr := substr(xmlstr,nlpos+1);  
    IF (nlpos = 0) THEN  
      dbms_output.put_line(' | '||xmlstr);  
      EXIT;  
    END if;  
  END LOOP;  
END printBufferOut;  
/  
show errors;  
CREATE OR REPLACE PROCEDURE dbvalid(xmlid IN NUMBER, xsdid IN NUMBER) IS  
  p_xml varchar2(4000);  
  p_xsd varchar2(4000);  
  p_out varchar2(4000);  
  
begin  
  select xml into p_xml from xml_tab where id=xmlid;  
  select xsd into p_xsd from schema_tab where id=xsdid;  
  p_out := SchemaValidation(p_xml,p_xsd);  
  printBufferOut(p_out);  
end;  
/  
show errors;
```

Table 7: Example Call of Validation Function

7.3 Java Metadata Interface

This test case has been made with the Package `javax.xml.validation` functionality. The package provides an API for validation of XML documents. Validation is the process of verifying that an XML document is an instance of a specified XML schema. An XML schema defines the content model (also called a grammar or vocabulary) that its instance documents will represent. The following example demonstrates validating an XML document with the Validation API (for readability, some exception handling is not shown):

```
// parse an XML document into a DOM tree
DocumentBuilder parser = DocumentBuilderFactory.newInstance().newDocumentBuilder();
Document document = parser.parse(new File("instance.xml"));
// create a SchemaFactory capable of understanding WXS schemas
SchemaFactory factory = SchemaFactory.newInstance(XMLConstants.W3C_XML_SCHEMA_NS_URI);
// load a WXS schema, represented by a Schema instance
Source schemaFile = new StreamSource(new File("mySchema.xsd"));
Schema schema = factory.newSchema(schemaFile);
// create a Validator instance, which can be used to validate an instance document
Validator validator = schema.newValidator();
// validate the DOM tree
try {
    validator.validate(new DOMSource(document));
} catch (SAXException e) {
    // instance document is invalid!
}
```

Table 8: Sample Usage of JavaX Validation

The JAXP parsing API has been integrated with the Validation API. Applications may create a Schema with the validation API and associate it with a `DocumentBuilderFactory` or a `SAXParserFactory` instance by using the `DocumentBuilderFactory.setSchema(Schema)` and `SAXParserFactory.setSchema(Schema)` methods.

XML Validation

The simplest way to validate an XML document is to use a `Validator` object. This object will perform a validation against the `Schema` object from which the `Validator` was created. `Schema` objects are typically created from `SchemaFactory` objects. The static `newInstance()` object allows you to create a `SchemaFactory` using a preset XML schema. The code in table 9 demonstrates this.

```
SchemaFactory factory =
    SchemaFactory.newInstance(XMLConstants.W3C_XML_SCHEMA_NS_URI);
Schema schema = factory.newSchema(new File("mySchema.xsd"));
Validator validator = schema.newValidator();
```

Table 9: SchemaFactory Instance Definition

Calling the `validate()` method on the `Validator` object performs the actual validation. This method takes at least a `javax.xml.transform.Source` object, of which you can use a `SAXSource` or a `DOMSource`, depending on your preference.

```
DocumentBuilder parser =
    DocumentBuilderFactory.newInstance().newDocumentBuilder();
Document document = parser.parse(new File("myXMLDocument.xml"));
validator.validate(new DOMSource(document));
```

Table 10: Source XML Validation

In the next table 11 there is a simple source example that shows how to validate an XML document using a [World Wide Web Consortium \(W3C\) XML Schema](#), sometimes referred to as WXS.

```
try {
    // Parse an XML document into a DOM tree.
    DocumentBuilder parser =
        DocumentBuilderFactory.newInstance().newDocumentBuilder();
    Document document = parser.parse(new File("myXMLDocument.xml"));

    // Create a SchemaFactory capable of understanding WXS schemas.
    SchemaFactory factory =
        SchemaFactory.newInstance(XMLConstants.W3C_XML_SCHEMA_NS_URI);

    // Load a WXS schema, represented by a Schema instance.
    Source schemaFile = new StreamSource(new File("mySchema.xsd"));
    Schema schema = factory.newSchema(schemaFile);

    // Create a Validator object, which can be used to validate
    // an instance document.
    Validator validator = schema.newValidator();

    // Validate the DOM tree.
    validator.validate(new DOMSource(document));

} catch (ParserConfigurationException e) {
    // exception handling
} catch (SAXException e) {
    // exception handling - document not valid!
} catch (IOException e) {
    // exception handling
}
```

Table 11: XML Validation with WXS

The `newInstance()` method takes in a constant to indicate which type of schema it can expect. Currently, the only schema that is required is the W3C XML Schema. This is an object-oriented schema language that provides a type system for constraining the character data of an XML document. WXS is maintained by the W3C and is a W3C Recommendation (that is, a ratified W3C standard specification).

The present example executes the source code on the following XML file:

```
<?xml version="1.0"?>
<birthdate>
```

```
<month>January</month>
<day>21</day>
<year>1983</year>
</birthdate>
```

Table 12: XML Sample

In addition, the example includes the following W3C XML Schema document as the XML validation schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd" />
  <xs:element name="birthdate">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="month" type="xs:string" />
        <xs:element name="day" type="xs:int" />
        <xs:element name="year" type="xs:int" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Table 13: XML Validation With W3C Schema

If the validation is successful, the program will run without incident. However, if there is a spelling error on the `month` element:

```
<amonth>January</amonth>
```

Table 14: XML Forced Error

At this point, the `Validator` will throw a `SAXException`, the first few lines of which are shown in the next table.

```
ERROR: 'cvc-complex-type.2.4.a: Invalid content was
found starting with element 'amonth'. One of '{"":month}'
is expected.'
org.xml.sax.SAXParseException: cvc-complex-type.2.4.a:
Invalid content was found starting with element 'amonth'.
One of '{"":month}' is expected.
  At ... (Util.java:109)
  at ... (ErrorHandlerAdaptor.java:104)
  ...
```

Table 15: Validation Exception

8. Conclusions

This survey has proved that users can find, at present, different solutions as on the software market of commercial products as in the freeware source areas. These solutions can address the main requirements related to management (including validation and interoperability) of metadata resources.

The wide range of this software, their price-model (i.e. commercial rather than shareware/freeware) and the involved technologies establish a background that, although not fully comprising all the possible demands, by all means is a good starting point within the GIS area that is now expanding and will surely keep on growing in the next years.

The more and more wide importance of GIS data and solutions at different level of public organisations and commercial activities on one hand and the need of common interfaces, specifications, and interoperability services on the other hand will boost the development of new solutions and the enhancement of existing ones in order to better suit the increasing needs.

Even if the existing software solutions related to GIS and in general to metadata are still able to improve and expand, the time has come for a proper diffusion and better usage of them.

The present study has also noticed an orientation of the market standards and solutions toward the requirements and needs of the American organisations and commercial companies. This tendency is necessarily going to influence the development of new commercial software. However, the application of the INSPIRE directive in the public organisations and administrations of the European Community countries will counterbalance this tendency, boosting the acceptance and spreading of the new metadata standard. This standard will give to European countries a proper ground to share data and services related to GIS resources.

This scenario will thus imply the development of new software, or the extension of existing one, that will be able to face toward the new challenges set up by the INSPIRE programme and fulfil the market expectation.

Annex A: Terms, Definitions and Abbreviations

Acronyms

ANSI	American National Standards Institute
CEN	Comité Européen de Normalisation
CT	Consolidation Team
DT	Drafting Team
DGIWG	Digital Geographic Information Working Group
EU	European Union
FGDC	Federal Geographic Data Committee
GIS	Geographic Information System
GML	Geography Markup Language
INSPIRE	INfrastructure for SPatial InfoRmation in Europe
IRs	Implementing Rules
ISO	International Organization for Standardization
JRC	European Commission Directorate General Joint Research Centre
LMO	Legally Mandated Organizations
NSDI	National Spatial Data Infrastructure
OGC	Open Geospatial Consortium
SDI	Spatial Data Infrastructure
SDIC	Spatial Data Interest Communities
UML	Unified Modelling Language
UNGIWG	UN Geospatial Information Working Group
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

More Acronyms

CEN TC287	CEN Technical Committee 287 Geographic Information
ISO/IEC	ISO/International Electrotechnical Commission
ISO/TC211	ISO Technical Committee 211 Geographic information/Geomatics

Annex B: Inspire References

The main information related to the Inspire programme can be found at the following link:

http://inspire.jrc.it/reports/transposition/INSPIRE_IR_WP2007_2009_en.pdf

Key components of the Inspire framework

- Reference and thematic data,
- Metadata that describes who has these resources, what are their characteristics, and the conditions for access and use,
- A Data policy framework setting the rules for access and sharing,
- Co-ordinating mechanisms to make sure that it all works.

Crucial points addressed by the Inspire programme.

1. Spatial data gaps.
Full European coverage for certain data sets in accordance with minimum quality criteria is essential for efficient use of data from a variety of sources. However, there remain important gaps in Europe even for the most essential spatial data sets.
2. Data documentation is often lacking.
In many cases data documentation does not exist, making it impossible to find back possibly valuable information; existing data documentation is available in a variety of formats
3. Spatial data sets are not compatible/interoperable.
Most uses of spatial data require data from various sources (e.g. combine environmental information with basic topographic data, combine information on environment and health), but data from various sources are often not compatible. This requires repeated adaptation of data sources or discourages the use of the spatial data.
4. GIS initiatives in Europe are often incompatible.
Technology progress allows us today an integrated discovery, access and use of spatial data from different sources, located at different sites. Several communities have set up their own mechanisms for exchanging spatial data (e.g. regional spatial data infrastructures or thematic spatial data infrastructure like the bio-diversity clearinghouse mechanism), but often these initiatives are not co-ordinated across the boundaries of the communities involved, leading to duplication and forgone potential economies of scale. In Europe, an overarching initiative is needed that will bring together the existing and emerging initiatives into one consistent framework.
5. Barriers for use.
Important barriers exist of a procedural, legal or financial nature for access and use of spatial data, even between public sector bodies. There is often no culture of sharing of information between public sector bodies. Therefore possibilities for reuse of information between different levels of government are limited, leading to duplication of data collection and maintenance. In addition, many public bodies apply prohibitive charges or licensing conditions for the reuse of spatial data (including to other public bodies).

Implementation phases and milestones

The implementation has been articulated in 3 phases, ranging from Commission proposal (7/2004) to Community Directive.

1. Preparatory phase (2004-2006)
 - Co-decision procedure
 - Preparation of Implementing Rules (IR)
2. Transposition phase (2007-2008)
 - Directive enters into force
 - Transposition into national legislation
 - INSPIRE Committee starts its activities
 - Adoption of Implementation Rules by Committology
3. Implementation phase (2009-2013)
 - implementation and monitoring of measures
 - adoption of second set of Implementing Rules

The next table 15 summarise the Inspire implementation milestones.

Milestone	Delta	Description
2007	X	Entry into force of INSPIRE Directive
2007+3m	X+3 months	Establishment of the INSPIRE Committee
2007	X+1 year at the latest	Adoption of Implementing Rules for the creation and up-dating of the metadata Adoption of Implementing Rules for network services Adoption of Implementing Rules on third parties use of the upload services Adoption of Implementing Rules for monitoring and reporting Adoption of Implementing Rules governing access and rights of use to spatial data sets and services for Community institutions and bodies
2009	X+2 years at the latest	Adoption of Implementing Rules for the use of spatial data sets and services by third parties
2009	X+2 years	Adoption of Implementing Rules for harmonised spatial data specifications and for the exchange of Annex I spatial data
2009	X+2 years	Provisions of Directive are brought into force in MS (transposition date)
2009	X+2 years	Designation of responsible public authorities for spatial data sets and services
2009	X+2 years	Implementation of sharing framework of spatial data sets and services between public bodies
2009	X+2 years	Implementation of provisions on monitoring
2009	X+2 years	Network services are operational
2010	X+3 years	Metadata available for spatial data corresponding to Annex I and Annex II spatial data
2010	X+3 years	Member States' First Report to the Commission. From then onwards MS have to present reports every 3 years
2011	X+4	New or updated spatial data sets available in accordance with Implementing Rules for harmonised spatial data specifications and exchange for Annex I spatial data

2012	X+5	Adoption of Implementing Rules for harmonised spatial data specifications and for the exchange of Annex II and Annex III spatial data
2013	X+6	Metadata available for Annex III spatial data
2014	X+7	New or updated spatial data sets available in accordance with Implementing Rules for harmonised spatial data specifications and exchange for Annex II and Annex III spatial data
2014	X+7	Commission's report to the EP and the Council. From then onwards the Commission has to present reports every 6 years

Table 16: INSPIRE Implementation Milestones

The next section summarise the planned tasks assigned to the involved working groups.

Spatial Data Interest Communities

- Identify user requirements
- Provide expertise to INSPIRE Drafting Teams
- Participate in the review process of draft Implementing Rules
- Develop, operate and evaluate implementation pilots
- Develop guidelines, awareness raising and training

Legally Mandated Organisations

- Collaborate within the SDICs, or autonomously in providing technical specifications
- Help to identify user needs
- Contribute to the analysis of the technical and operational feasibility of implementation of proposed draft Implementing Rules
- Provide feedback on the cost/benefit consequences of Implementing Rules at Member State level.

Drafting Teams

- Analyse and review the reference material
- Write draft INSPIRE Implementing Rules
- Provide recommendations to the CT (in case of conflicting technical specifications)
- Provide suggestions to the CT for testing any proposed specification

Consolidation Team

- Coordinate and support DTs' activities
- Coordinate with the SDIC, which includes:
 - Ensure coherence between different Implementing Rules
 - Final edit harmonised draft Implementing Rules ensuring their overall coherence
- Coordinate Pilot and Prototype projects
- Coordinate the review processes
- Follow the link to other directives under preparation and to ensure feedback from the co-decision process.

- Link to legal services of the European Commission

Pilots and Prototypes

- Development of the EU geo-portal
 - Development by JRC
 - Based on standards (ISO, CEN, OGC, W3C)
- WFD as pilot for implementation
 - Need for adoption of a common GIS dataset
 - Harmonisation of encoding
- Eurostat INSPIRE pilot: SDIGER
 - Feasibility of interoperable solutions
 - Cross border case: WFD for Adour - Garonne
- EEA SDI prototyping
 - EIONET as example of SDIC

The following table shows the number of registered SDICs, LMOs and Projects involved in the Inspire drafting and review phase.

Who/What	Registered (num.)	Link
SDICs	195	http://www.ec-gis.org/inspire/ir/list_registered_sdics.cfm
LMOs	110	http://www.ec-gis.org/inspire/ir/list_registered_lmos.cfm
Projects	28	http://www.ec-gis.org/inspire/ir/list_registered_projects.cfm

Table 17: Registered SDICS, LMOs & Projects

The figure 7 below displays the Inspire target goal, that is the creation of an Infrastructure for Spatial Information.

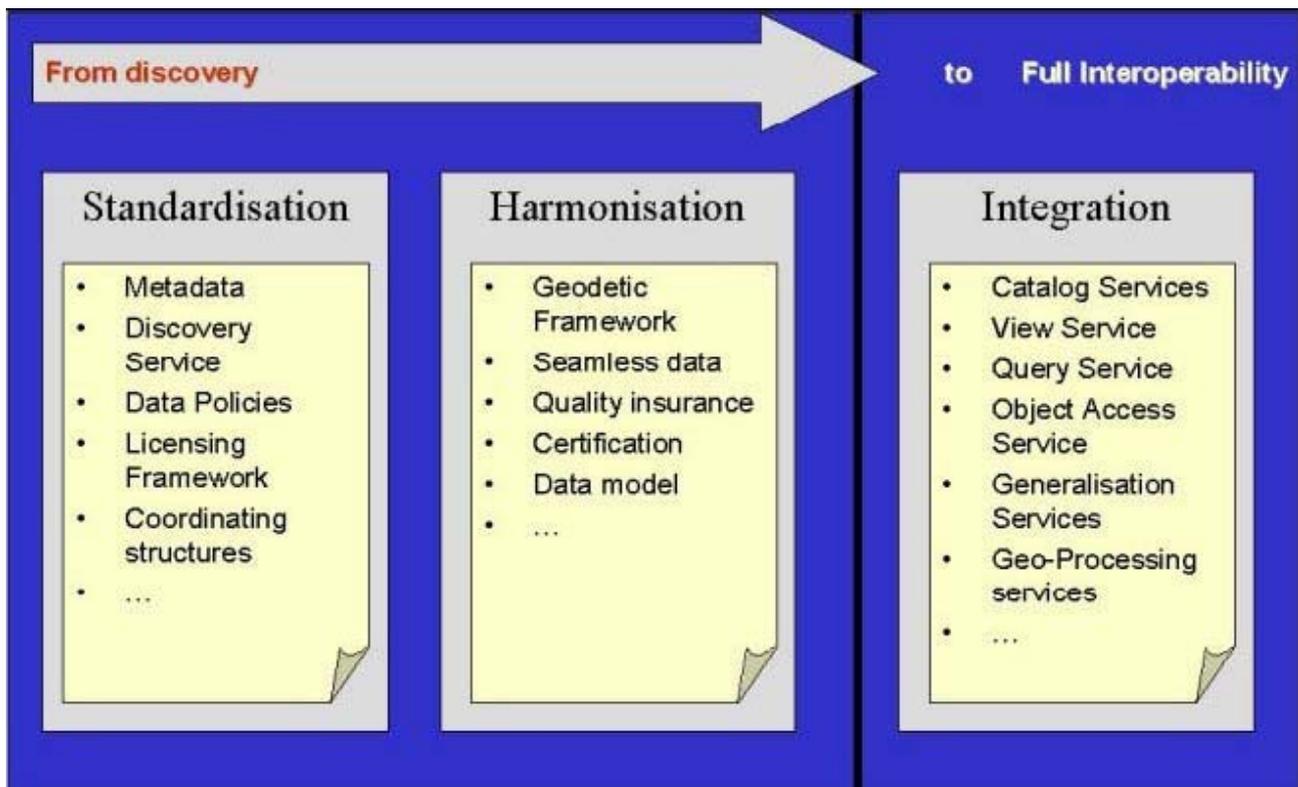


Figure 7: INSPIRE Scenario

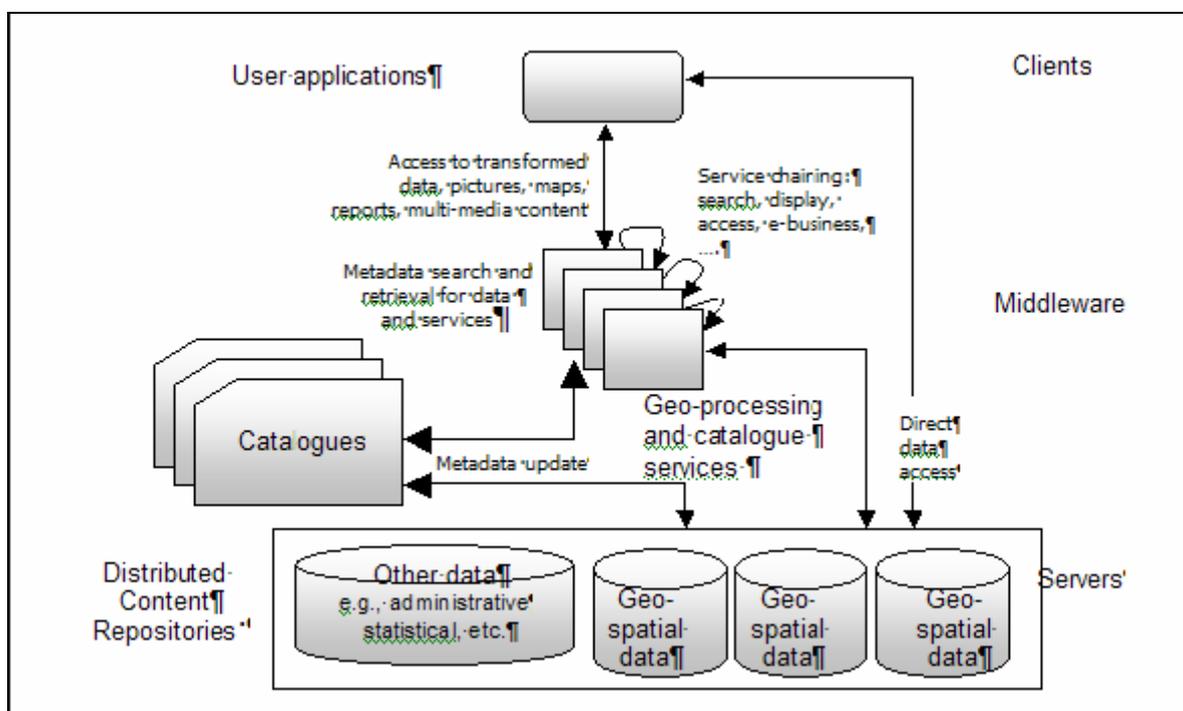


Figure 8: INSPIRE Architecture Reference Model

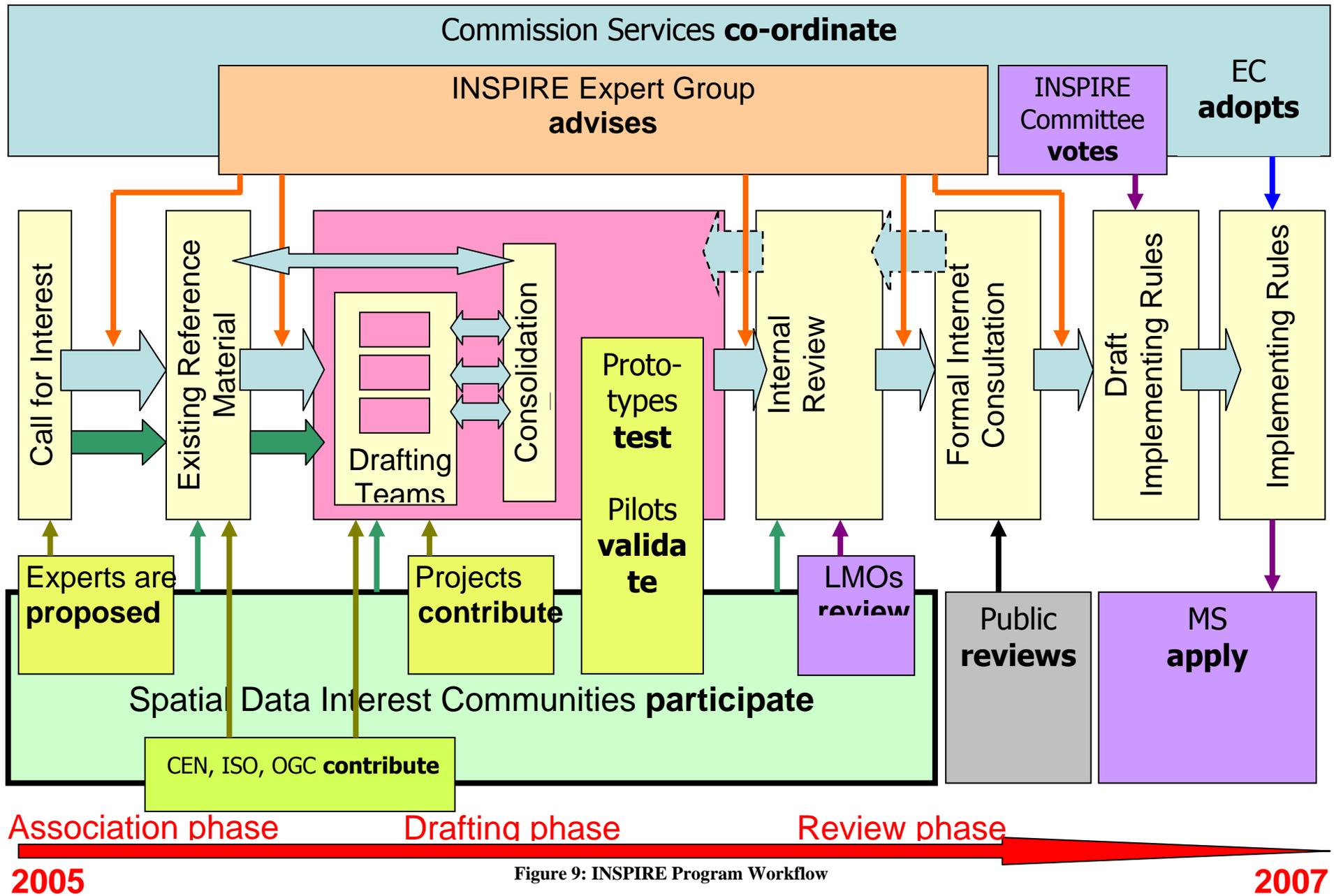


Figure 9: INSPIRE Program Workflow

Annex C: Bibliography

DT Metadata – Draft Implementing Rules for Metadata, INSPIRE – 2007
Reference: draftINSPIREMetadataIRv2_20070202.doc

INSPIRE Metadata Survey Results, JRC-IES - Joanna Nowak and Massimo Craglia - 2006
Reference: INSPIRE_Metadata_Survey_2006_final.pdf

INSPIRE Web Site Homepage:

Web link: <http://www.ec-gis.org/inspire/home.html>

INSPIRE Metadata Implementing Rules – GI&GIS in Italia - 2007

Reference: inspire_metadata_irs_osservazioni_v1.doc

Web link: www.freegis-italia.org

Commercial and Open-Source GIS Software, FreeGIS-Italia Web Site

Web link:

http://www.freegis-italia.org/index.php?option=com_content&task=section&id=6&Itemid=86

Spatial Data Standards and GIS Interoperability, ESRI - 2003

Reference: ESRI spatial-data-standards.pdf

GIS Standards and IT Interoperability, ESRI Web Site

Web link: <http://www.esri.com/software/standards/index.html>

Software QA and Testing Resource Centre, Rick Hower Web Site

Web link: <http://www.softwareqatest.com/qatweb1.html>

Metadata Tools for Geospatial Data, Wisconsin NSDI Clearinghouse Web Site

Web link: <http://sco.wisc.edu/wisclinc/metatool/#TOOLBOX>

Metadata Tools, Colorado Plateau Web Site

Web link: <http://www.mpcer.nau.edu/metadata/MetadataTools.htm>

European Commission

EUR 23036 EN – Joint Research Centre – Institute for Environment and Sustainability

Title: Metadata State of Play, Compliance Testing and Interoperability Checking

Author(s): S. Viganó, M. Millot

Luxembourg: Office for Official Publications of the European Communities

2007 – 57 pp. – 21 x 29.7 cm

EUR – Scientific and Technical Research series – ISSN 1018-5593

ISBN 978-92-79-07690-9

DOI 10.2788/53688

Abstract

The document contains an inventory of existing solutions for compliance testing and interoperability checking of metadata taking into account the draft INSPIRE metadata implementing rules, the assumptions that the metadata will be available in the XML format following an INSPIRE metadata schema definition file, may contain other items that are data theme specific and not included in the Implementing Rules and will be accessible for compliance testing and interoperability checking through a push and pull mechanism. Even if the emphasis is on geographical information and therefore on Geographical Information Systems, the document describes applicable solutions outside the geographical Information System domain.

How to obtain EU publications

Our priced publications are available from EU Bookshop (<http://bookshop.europa.eu>), where you can place an order with the sales agent of your choice.

The Publications Office has a worldwide network of sales agents. You can obtain their contact details by sending a fax to (352) 29 29-42758.

The mission of the JRC is to provide customer-driven scientific and technical support for the conception, development, implementation and monitoring of EU policies. As a service of the European Commission, the JRC functions as a reference centre of science and technology for the Union. Close to the policy-making process, it serves the common interest of the Member States, while being independent of special interests, whether private or national.



ISBN 978-92-79-07690-9

