

Open Source and Open Standards for using integrated geographic data on the web

A. Felicetti¹ and M. Lorenzini¹

¹PIN, University of Florence, Prato, Italy

Abstract

This article describes the first results of our research concerning the development of a complete Open Source system based on W3C and ISO 19100 standards to integrate and manage spatial and non-spatial archaeological information on the Web. The system is based on MAD, a web tool originally developed to manage archaeological semantic datasets encoded in RDF using the CIDOC-CRM ontology. Geographic functions have been implemented to integrate spatial archaeological information for the management of unstructured documents, such as excavation diaries and reports, in a spatial context. The system will allow the creation and distribution of rich geospatial relationships across the Web and the use of geographic data in a Semantic Web scenario. The Geographic Markup Language (GML) has been used in our system to store geographic data related to archaeological records. GML information has been created using Open Source GIS software starting from vectorial data (.shp or .dxf). Brand new GML documents can be also created starting from non-spatial data. The advanced query system in MAD allows the creation of Semantic Web enriched data combining spatial and non-spatial information and using ontologies. Data serialized by the MAD system can be exported in SVG or visualized using map server web applications. The flexibility of GML features will also allow the implementation of complex query-on-map functions to visually query and generate dynamic maps. The tool can be also used to host and serialize KML archaeological files to be used in Google Earth and Google Maps applications.

Categories and Subject Descriptors (according to ACM CCS): H.3.5 [Information Storage and Retrieval]: Online Information Services

1. Introduction

The activity carried on by the EPOCH team working on documentation standards and databases focuses on the implementation of CIDOC-CRM-compliant data structures and tools covering all the needs of heritage documentation. In particular, the team has developed MAD, an XML-native DBMS [Fel06]; it has showed [ND06] how to incorporate 3D ontologies into the CIDOC-CRM one by means of an extension of CIDOC-CRM based on the identification of a CRM entity with the root of a 3D ontology, exemplifying the case of X3D; and has provided [AMA] a methodology and a tool (named AMA) for mapping diverse documentation systems onto a CIDOC-CRM compliant structure to guarantee their interoperability. An important characteristic of archaeological documentation systems resides in the presence of spatial features. Although not always included in

the documentation — in fact spatial features have less relevance for museum documentation, for example spatial information is paramount when on-field data are considered. This is clearly witnessed by the widespread usage of GIS to store excavation data as well as survey data, and so on. On the other hand, current GIS systems rely on RDBMS to store the information about the objects referenced in the geographic system, so it seems that XML-based documentation systems (as MAD, for example) cannot avail of GIS nor include spatial search functions among their features. Actually CIDOC-CRM provides for spatial entities, such as Place and Coordinates. However, these appear to be aimed at giving a general spatial reference, as it may be encountered in historical sources, rather than a geometric or geographic reference as usually processed by GIS. In a technical report of 2001 [Doe01] Martin Doerr compared OpenGis abstract specifications with CIDOC-CRM, concluding that there is

some overlap but the different scopes of the two may require some work for an harmonization that could be beneficial to both. After this report appeared, GML was developed, an XML specification of geographic entities, concepts and relations strictly related to OpenGis. In the present paper we try to incorporate part of the GML specifications into CIDOC-CRM identifying them with the specific documentation entity Information Object [CRM]. The space-related entities in CIDOC-CRM are considered as part of the “descriptive” documentation, to be processed by a different tool – a human being or a parser – to guarantee coherence: the perspectives of CIDOC-CRM spatial entities and GIS entities look rather different, as confirmed by Doerr’s notes. We will not consider the time concepts, which look substantial to CIDOC-CRM and rather incidental to GML. We hope to deal with these in a future paper. In conclusion, our goal is to extend CIDOC-CRM in the simplest way, taking some (standard) parts from GML as an extension of a CIDOC-CRM entity, to enable archaeologists to use CIDOC-CRM-compliant systems also when they need spatial features [LN07]. In other words, the “absence” of GIS functionality will no more be an excuse to avoid using CIDOC-CRM in current research activities, as MAD will be able to fully support spatial functionality.

2. The GML technology

The format we have chosen to represent the geographic information for our datasets is GML (Geography Markup Language) [GML], an XML-based grammar for the modelling, exchange, and storage of geographic information. GML grammar specified in one or more XML Schemas, is able to describe spatial objects using a variety of coordinate reference systems, geometry, topology, time, units of measure and generalized values to build a set of geographic features that can be considered a digital representation of the real world [CLPA].

GML is going to become the standard format for many geographic web applications able to do many of the same things that usually would have needed a desktop GIS to be accomplished. With GML, for example, visualization does not require any GIS tool because a simple transformation of the GML to another XML based format, like Scalable Vector Graphics (SVG) or eXtensible 3D graphics (X3D), can be accomplished through some scripting and a XSL stylesheet [XLS] describing how to represent the GML features. Geographic data in GML can be also sent to any device with an XML interface and displayed on any XML-enabled device, like the new generation PDAs and mobile phones.

This is possible because GML has a standardized syntax to describe geographic features using XML Schema [XSD] files to specify the rules and formats for encoding spatial features, such as lines and polygons. Users can also define more application specific schemas or document type definitions by extending the available GML schemas. GML can

take advantage of the XLink [XLK] and XPointer [XPO] mechanisms to create chains of GML documents by linking features coming from two or more GML data stores. This allows the construction of a virtual dataset by setting up links between.

3. MAD: the data integration environment

The platform we are working on to test the advantages of an integrated system is MAD, a web application originally designed to manage semantically encoded archaeological datasets to be used in a semantic web integrated context [SW]. The MAD system is actually a high performance, scalable native XML-based management server with both data- and document-centric capabilities. MAD’s strength comes from its core technology, a highly flexible native XML database, eXist [EXI], with a model that is optimal for managing and storing any kind of XML data, including semantic information (encoded for instance in RDF [RDF] or OWL [OWL]) and geographic formats (like GML and KML) and providing the high level of efficient persistence that XML applications and transactions require to be reliable [Mei03]. One of the key benefits in using MAD is that the server does not need to know the schema or structure of data prior to processing and storing it. This ability to make the XML Schemas optional is a vital innovation when dealing with geographic data, because the structures of this kind of documents frequently change and mapping schemas for the purpose of linking to a new data source is both difficult and time consuming.

3.1. Storing GML information in MAD

GML information coming from GIS software – e.g. the Open Source GIS GRASS [GRA] or any commercial system that can export data to GML – or from other GML web repositories can be directly stored into MAD and made immediately available for operations using the eXist web interfaces. All the necessary XML Schemas related to GML information can be stored as well if provided together with data or can be directly retrieved over the web using the namespace mechanism. The XML Schemas can be used for the validation of documents and query results and for the creation of on-the-fly templates to serialize geographic data. Nevertheless the presence of the XML Schemas in the MAD repository is not strictly necessary for the MAD server to work with geographic data, since the namespace mechanism itself is enough, in most of the cases, to guarantee data validity and usability. New GML documents can be also imported from distributed geographic archives or created dynamically as results of XLink/XPointer operations or internal geographic queries, and stored in MAD to be reused for future works.

3.2. The MAD query system

Once spatial and non-spatial data are stored into MAD, a wide range of query types can be written and executed in or-

der to extract meaningful sets of XML-encoded fragments to be used in different contexts. The query framework of MAD offers many possibilities to explore and combine semantic and geographic data through an advanced set of graphical user interfaces provided for all the necessary data management operations. MAD can also act as an XML information provider, returning query results to incoming requests of remote applications since stored data can be accessed and queried through many different protocols, including XML-RPC, SOAP, REST-style HTTP or direct access via the XML:DB API. Structural queries for pure XML data and semantic queries for ontology-based information are already fully implemented and working [Hun03]; the geographic query framework is still under development due to some well known difficulties in querying GML data [JAF03].

The preferred tool to query XML information in our system is XQuery [XQR], a W3C standard language based on XPath [XPA] and very similar to SQL in syntax but specifically designed to query the XML data structure. W3C XQuery is fully supported by MAD together with some other powerful functions directly provided by eXist as XQuery extensions, used to create documents from scratch and to delete, move and rename existing documents and collections in the database. SPARQL [SPA] and RQL [RQL] semantic query languages are also supported by MAD and are efficiently used to extract subgraphs from RDF encoded documents.

3.3. Spatial queries in MAD

The problems with geographic data relates to the nature of GML data: since GML is features-based every GML document is a list of spatial properties and geometries, and we need to perform spatial analysis (area of a polygon, length of a line) and highlight spatial relationships (containment, overlapping) when querying this kind of data. Languages that are currently used for querying XML data are fully dedicated to tree manipulations, can only get alphanumeric features and are not suitable for spatial relations. XQuery for instance does not highlight the spatial properties of the set of tags representing points or polygons but can only select parts of the GML sub trees.

Several solutions have been tested for querying GML data and many language based on XQuery have been tested to deal with features and to return meaningful objects [BC04]. One of the most promising solutions is to write all the necessary spatial functions in XQuery itself (the XQuery grammar proposed by W3C allows the definition of user external functions [XQF]), but this is quite difficult, because XQuery is a functional language that doesn't offer data structures for implementing geometric algorithms.

OpenGIS Web Feature Service requests (WFS) [WFS], used for describing data manipulation operations on geographic features, provides some useful query results to re-

quests sent via HTTP; but its efficiency depends mostly on the underlying GIS capabilities [OGI].

The experimental solution we are testing to perform spatial queries in our system, relates to the XML documents' indexing strategies of the eXist XML database and to the extensibility of the XQuery system through a set of Java modules very simple to update and maintain. Recently eXist's developers have updated the indexing system providing a new mechanism to index XML data [Mei06]. The new mechanism is modular and should ease the development of related custom functions. A Java module for indexing spatial data has been added to the indexing mechanism for enhancing the storage of geometric characteristics of GML geometries. Together with this geographic indexing module, an XQuery set of spatial functions has been developed to query GML features (under the <http://exist-db.org/xquery/spatial> namespace) [Bri07] and typical geographic queries can be now written using the XQuery syntax and the spatial namespace. The following query, for instance, selects all the polygons included in a given boundary.

```
declare namespace
gml="http://www.opengis.net/gml";
declare namespace
spatial="http://exist-
db.org/xquery/spatial";

spatial:within(//gml:Polygon,
<gml:Polygon
xmlns:gml='http://www.opengis.net/gml'>
  <gml:outerBoundaryIs>
    <gml:LinearRing>
      <gml:coordinates>
        2978200,187600 287400,187600
        2878400,188000 278200,188000
        2978200,187600
      </gml:coordinates>
    </gml:LinearRing>
  </gml:outerBoundaryIs>
</gml:Polygon>
)
```

The syntax is very clear: after the necessary namespaces declaration for GML and the spatial functions of eXist, the `spatial:within` function is used to select all the `gml:Polygon` elements that falls within the boundary defined by the `gml:LinearRing`. The query returns another GML dataset containing the resulting polygons that can be visualized as shown in figure 1.

Other typical spatial queries can select buffers or intersections

```
declare namespace
gml="http://www.opengis.net/gml";
declare namespace
spatial="http://exist-
db.org/xquery/spatial";

spatial:buffer(
```

```

<gml:Polygon
xmlns:gml='http://www.opengis.net/gml'>
  <gml:outerBoundaryIs>
    <gml:LinearRing>
      <gml:coordinates>
        268100,187600 268400,187600
        268400,188000 278200,188000
        268100,187600
      </gml:coordinates>
    </gml:LinearRing>
  </gml:outerBoundaryIs>
</gml:Polygon>, 500
)

```

```

declare namespace
gml="http://www.opengis.net/gml";
declare namespace
spatial="http://exist-
db.org/xquery/spatial";

spatial:intersects(//gml:Polygon,
  <gml:Polygon
xmlns:gml='http://www.opengis.net/gml'>
  <gml:outerBoundaryIs>
    <gml:LinearRing>
      <gml:coordinates>
        278201,187600 278401,187600
        278401,188000 278201,188000
        278201,187600
      </gml:coordinates>
    </gml:LinearRing>
  </gml:outerBoundaryIs>
</gml:Polygon>
)

```

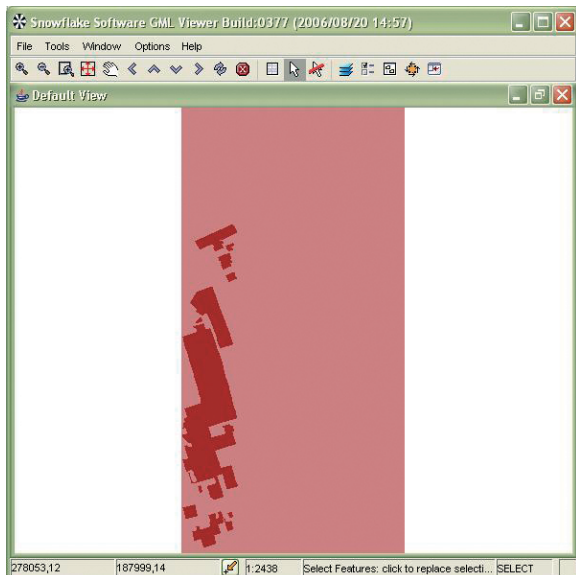


Figure 1: Results of the sample *spatial:within* query.

The results of these queries are still GML fragments that

can be reused, visualized or sent to other applications for further operations.

4. Data aggregation

The complexity and the power of the MAD query system can generate complex set of XML fragments by recursively executing chains of queries for the generation of aggregated data for the semantic web. This operation is made easy by the fact that the original GML model was based on the World Wide Web Consortium's RDF. The OGC introduced XML Schemas for GML structure mainly to define more easily the XML relational structure in order to enhance the connection of the various existing geographic databases. The new XML Schema based GML however shows many features of RDF, including the use of child elements as properties of the parent ones and the use of references for remote properties. Thanks to this feature it is very easy to nest fragments of GML into RDF CIDOC-CRM encoded documents simply instantiating a CIDOC-CRM E73. Information Object and creating a relation with the place the GML object refers to [CDG03], as clarified by the following code fragment:

```

<?xml version="1.0" encoding="ISO-8859-7"?>
<rdf:RDF xml:lang="en"
xmlns:rdf="http://www.w3.org/1999/02/
22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/
rdf-schema#"
xmlns:crm="http://cidoc.ics.forth.gr/
rdfs/cidoc_v4.2.rdfs#"
xmlns:gml="http://www.opengis.net/gml">

```

```

<crm:E53.Place rdf:about="US1020">
  <crm:P67B.is_referred_to_by>
    <crm:E73.Information_Object
      rdf:about="gmlModel_US1020">
      <gml:Polygon srsName="osgb:BNG">
        <gml:outerBoundaryIs>
          <gml:LinearRing>
            <gml:coordinates>
              278534.100,187424.700
              278529.250,187430.900
              278528.700,187431.650
              278527.250,187433.600
            </gml:coordinates>
          </gml:LinearRing>
        </gml:outerBoundaryIs>
      </gml:Polygon>
    </crm:E73.Information_Object>
  </crm:E53.Place>
</rdf:RDF>

```

The possibility to create RDFized GML code embedding fragments of GML inside RDF, allows RDF documents to use GML statements for the description of spatial features in a semantic web environment [BL98] to be visualized by semantic enabled browsers with geographic extensions, like the W3C Tabulator [TAB].

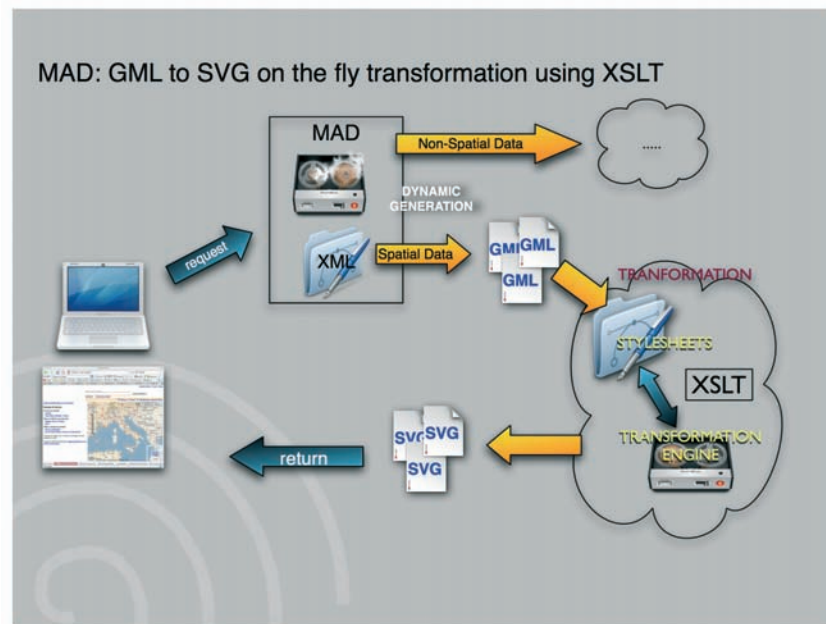


Figure 2: The SVG serialization of GML data in MAD

5. Data Presentation

The presentation framework of MAD is entirely driven by a pipeline mechanism build through a set of logic components able to read and understand HTTP Requests and to perform operations accordingly. For the construction of our pipeline we used some of the functions provided by the Cocoon framework, an advanced, flexible and easy to configure Java-based engine for data processing, based on XML technology and using simple XML configuration files to work [COC]. The pipeline takes in charge the query results, selects the sequential chains of XSL stylesheets to be dynamically applied to XML fragments and documents returned by queries and formats the content according to users' and applications' requests. This mechanism allows, for instance, the selection of geographic information contained in non-spatial XML documents and their transformation into GML datasets to be serialized for the final visualization.

Typically, Scalable Vector Graphics (SVG) can be used as the drawing object language because of its XML nature [SVG]. As long as the new browsers supports vector graphics, then an SVG map can be created transforming the GML fragments and displayed without any additional software. SVG drawings can be dynamic and interactive, as the Document Object Model (DOM) for SVG allows straightforward and efficient vector graphics animation via Javascript and a rich set of event handlers such as onmouseover and onclick directly assigned to any SVG graphical object by the MAD pipeline. For old browsers any SVG can viewer can be used to turn SVG code into an image (PNG, TIF, JPEG) us-

ing for instance the free downloadable plug-in from Adobe Corporation [ADO]. Many Open Source applications exist to be used as a server side render application, like the Apache Batik (see figure 2) [BAT].

The alternative solution is the use of a map server able to read and render GML natively using for instance a GML enabled web server tool like MapServer to convert GML documents in a set of drawing objects and to render them as a map on the browser. The same presentation mechanism is also able to provide complex and detailed files in the KML format, ready to be used in all the Google Maps/Google Earth applications, simply using the KML Schema to serialize transformation results [KML].

6. Testing the application

In order to test the capabilities of our system we used the archaeological datasets recorded during the excavation of the Anglo-Saxon mixed rite cemetery at Cleatham, in the parish of Manton, North Lincolnshire (UK), made by Kevin Leahy between 1984 and 1989 (National Grid reference SE932008). The database contains the documentation of about 1528 burials (urns and inhumations) of the early Saxon period and is available to download from the Archaeology Data Service web site as a series of comma delimited text files (.csv) [ADS]. The dataset contains detailed descriptions of inhumations and urns, including information on fabrics, classifications, decorations and locations (coordinates). Another section of the database records the finds collected

during the excavation activity and the coordinates of their original position.

GML data have been created from the dataset's geographic information - using the GRASS GIS software - and put into MAD together with the semantic version of the archaeological documentation derived from the .cvs archives. A GML example of an encoded set of urns follows:

```
<gml:FeatureCollection>

<gml:Feature gml:id="R63">
  <gml:description>
    Burnished urn
  </gml:description>
  <gml:name>urn_23</gml:name>
  <gml:location>
    <gml:Point gml:id="u23">
      <gml:pos>6.60 35.20</gml:pos>
    </gml:Point>
  </gml:location>
</gml:Feature>
[...]
```

```
<gml:Feature gml:id="R73">
  <gml:description>
    Unfinished urn
  </gml:description>
  <gml:name>urn_26</gml:name>
  <gml:location>
    <gml:Point gml:id="u26">
      <gml:pos>32.80 30.70</gml:pos>
    </gml:Point>
  </gml:location>
</gml:Feature>

</gml:FeatureCollection>
```

We have then tested the geographic query features of MAD over the new GML data to find set of inhumations and urns matching given spatial criteria. The following XQuery fragment, for instance, selects all the urns positioned at a maximum distance of 10 meters from the inhumation defined by the inhum43 polygon.

```
declare namespace
gml = "http://www.opengis.net/gml";
for $i/gml:Feature/gml:Location
/gml:Point[@gml:id="u*"]
in spatial:whithin
(
  <gml:Polygon gml:id="inhum43">
    <gml:outerBoundaryIs>
      <gml:LinearRing>
        <gml:pos>
          8 95.7 10 95.7 10 96.7 8 95.7
        </gml:pos>
      </gml:LinearRing>
    </gml:outerBoundaryIs>
  </gml:Polygon>, 10
)
return $i
```

The query returns a subset of gml:Feature objects like the following (see figure 3).

```
[...]
<gml:Feature gml:id="R518">
  <gml:location>
    <gml:Point gml:id="u518">
      <gml:pos>11.6 86.7</gml:pos>
    </gml:Point>
  </gml:Location>
</gml:Feature>
[...]
```

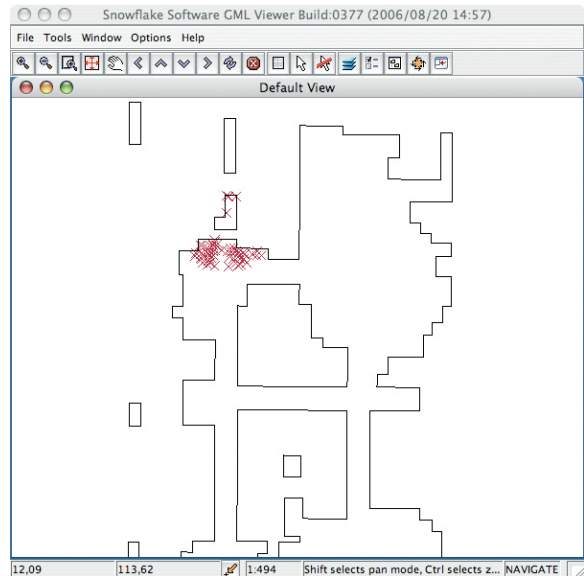


Figure 3: the spatial:whithin query in action.

This kind of queries can be useful to find the density of archaeological evidences in a given area or related to a given object in order to discover relevant phenomena. Other queries can be generated combining archaeological documentation and spatial data to find, for instance, all the urns containing remains of women and located in a given area in order to discover reserved places for the burial extending spatial analysis to social behaviours. The following XQuery fragment is a typical example of this kind of query:

```
declare namespace
gml = "http://www.opengis.net/gml";
for $i in //urns
where $i/sex="Female"
and $i/spatial:whithin
(
  <gml:Polygon>
    <gml:outerBoundaryIs>
      <gml:LinearRing>
        <gml:pos>
          7 85,6 11 96,6 12 96,7 7 95,7
        </gml:pos>
      </gml:LinearRing>
  </gml:Polygon>
)
```

```

    </gml:outerBoundaryIs>
  </gml:Polygon>
)
return $i/gml:Feature

```

7. Conclusions and future work

The MAD application is going to become the main integration platform for all kind of data coming from archaeological activities, including geographic data. One of the main key features of the integration process resides in the richness of the RDF syntax that easy the integration of spatial entities in ontological structures based on classes and relations, like the CIDOC-CRM one. In this paper we presented the new features for geospatial queries; the possibility to extend the already present query framework by creating new functions; the on-the-fly conversion and management of geographical and spatial datasets represented in GML together with non-spatial archaeological data. All the above makes MAD a natural environment for building high performance and scalable integrated applications [DN99].

The advanced web interfaces of MAD give the users a simple way to interact with integrated data and provide the heritage professional and institutions with the possibility of sharing their standard-encoded archives. MAD aims to become an advanced general purpose XML services provider to be used for the serialization of integrated information in different scenarios and for the delivery of on-demand information to remote applications using web oriented protocols. MAD will also act as a web mediation framework for the aggregation and redistribution of information coming from different archives. The long term validity of the system is guaranteed by the use of well known and well accepted standards like the W3C Recommendations, that make our data fully supported by any platform.

Future work will concern the full integration of geospatial features within the CIDOC-CRM framework for example providing tools to extract locational information contained in CIDOC-CRM compliant records (e.g. in `E53.Place` class and subclasses) into geographical references, availing of standardized gazetteers (lists of geo-referenced placenames) that play in this context the same role as thesauri for textual information.

8. Acknowledgements

MAD and its extensions have been developed as part of EPOCH [EPO], project no. IST-2002-507382. We would like to thank Andrea D'Andrea for the brilliant suggestions he gave us during the whole development and testing process, and Franco Niccolucci for his usual patience.

References

[ADO] Adobe SVG Viewer. <http://www.adobe.com/svg/viewer/install/main.html>.

[ADS] Archaeology Data Service. <http://ads.ahds.ac.uk/>.

[AMA] AMA, Archive Mapper for Archaeology. <http://www.epoch.eu/AMA/>.

[BAT] Apache Batik SVG Toolkit. <http://xmlgraphics.apache.org/batik/>.

[BC04] BOUCELMA O., COLONNA F.: GQuery: a Query Language for GML. *Proc. 24th Urban Data Management Symposium, Chioggia-Venice, Italy, October 2004* (2004).

[BL98] BERNERS-LEE T.: Semantic Web Roadmap, an attempt to give a high-level plan of the architecture of the Semantic WWW, September 1998. <http://www.w3.org/DesignIssues/Semantic.html>.

[Bri07] BRIHAYE P.: eXist Developer's Guide to Modularized Indexes, 2007. http://exist.sourceforge.net/devguide_indexes.html.

[CDG03] CROFTS N., DOERR M., GILL T.: The CIDOC-CRM Conceptual Reference Model: A standard for communicating cultural contents. *Cultivate Interactive* 9 (Feb. 2003). <http://www.cultivate-int.org/issue9/chios/>.

[CLPA] COX S., LAKE P. D. R., PORTELE C., A. WHITESIDE: Opengis Geographic Markup Language (GML) Implementation Specification. 29 January 2003.

[COC] The Apache Cocoon Project. <http://cocoon.apache.org/>.

[CRM] The CIDOC Conceptual Reference Model. <http://cidoc.ics.forth.gr/>.

[DN99] D'ANDREA A., NICCOLUCCI F.: A Web-based access to GIS. Integrating geographical databases through the WWW. Proceedings of the 2nd International Congress on 'Science and Technology for the Safeguard of Cultural Heritage in the Mediterranean basin', Paris, July 1999.

[Doe01] DOERR M.: A comparison of the OpenGIS Abstract Specification with the CIDOC CRM 3.2. ICS-FORTH, Heraklion, Crete. Oct. 4, 2001.

[EPO] EPOCH: The European Research Network of Excellence in Open Cultural Heritage. <http://www.epoch.eu>.

[EXI] eXist Open Source Native XML Database. <http://exist-db.org>.

[Fel06] FELICETTI A.: MAD: Managing Archaeological Data. *7th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST2006)* (2006). October 30, November 4, Nicosia, Cyprus.

[GML] GML: the Geography Markup Language. <http://www.opengis.net/gml/>.

[GRA] GRASS GIS. <http://grass.itc.it/>.

- [Hun03] HUNTER J.: Enhancing the semantic interoperability of multimedia through core ontology. <http://www.itee.uq.edu.au/~ereseach/papers/2003/hunter-core-ontology/paper.html>.
- [JAF03] JONES C., ABDELMOTY A., FU G.: Maintaining Ontologies for Geographical Information Retrieval on the Web. *Ontologies, Databases and Applications of Semantics for Large Scale Information Systems (ODBASE03) Proceedings of the OTM Confederated Int. Conf.* (2003), 934–951. R. Meersam, Z. Tari, Schmidt, D.C.Springer Verlag, LNCS 2888.
- [KML] KML documentation. <http://code.google.com/apis/kml/documentation/>.
- [LN07] LINKOVA Z., NEDBAL R.: Ontology approach to integration of geographical data. *WETDAP 2007, Proceedings of the 1st Workshop Evolutionary Techniques in Data-processing* (2007). Technical University of Ostrava, Ostrava, 2007.
- [Mei03] MEIER W.: eXist: An Open Source Native XML Database. In *Web, Web-Services and Database Systems* (2003). NODe 2002 Web- and Database-Related Workshops, Erfurt, Germany, October 7-10, 2002, 169-183. Lecture Notes in Computer Science 2593 Springer, 2003.
- [Mei06] MEIER W.: Index-driven XQuery processing in the eXist XML Database. Paper presented at the XML Prague 2006 Conference. <http://exist-db.org/xmlprague06.html>.
- [ND06] NICCOLUCCI F., D'ANDREA A.: An Ontology for 3D Cultural Objects. *VAST: International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage 2006* (2006), 203–210.
- [OGI] OpenGIS Implementation Specification: Coordinate Transformation Services, OpenGIS Consortium. <http://www.opengis.org/techno/specs/01-009.rtf>.
- [OWL] W3C Semantic Web Activity, Web Ontology Language. <http://www.w3.org/2004/OWL/>.
- [RDF] W3C Semantic Web Activity, Resource Description Framework. <http://www.w3.org/RDF/>.
- [RQL] RQL: A Declarative Query Language for RDF. www.ics.forth.gr/isl/publications/paperlink/www2002.pdf/.
- [SPA] SPARQL Query Language for RDF. <http://www.w3.org/TR/rdf-sparql-query/>.
- [SVG] Scalable Vector Graphics (SVG). <http://www.w3.org/Graphics/SVG/>.
- [SW] W3C Semantic Web Activity. <http://www.w3.org/2001/sw/>.
- [TAB] The Tabulator, Generic Data Browser.
- [WFS] OpenGIS Implementation Specification: Web Features Services (WFS), OpenGIS Consortium. <http://www.opengeospatial.org/standards/wfs>.
- [XLK] XML Linking Language (XLink) Version 1.0. www.w3.org/TR/xlink/.
- [XLS] XSL Transformations (XSLT) Version 1.0. W3C Recommendation 16 November 1999. <http://www.w3.org/TR/xslt>.
- [XPA] XML Path Language (XPath) Version 1.0. W3C Recommendation 16 November 1999. <http://www.w3.org/TR/xpath>.
- [XPO] XML Pointer Language (XPointer). <http://www.w3.org/TR/xptr/>.
- [XQF] XQuery Functions Documentation. <http://demo.exist-db.org/xquery/functions.xq>.
- [XQR] XQuery 1.0: An XML Query Language W3C Candidate Recommendation 8 June 2006. <http://www.w3.org/TR/xquery/>.
- [XSD] W3C XML Schema. <http://www.w3.org/XML/Schema>.