

Advantages of XML as a data model for a CRIS

Patrick Lay, Stefan Bärtsch
GESIS-IZ, Bonn, Germany

Summary

In this paper, we present advantages of using a hierarchical, XML¹-based data model as the basis for a CRIS, as opposed to the relational model, which is currently recommended by EuroCRIS. The main benefits lie in the relatively easy way to model data, since there is no need to create and maintain mappings between different models. Another advantage is the possibility to create a website for a CRIS easily out of the underlying model.

1 Comparing the relational with the hierarchical approach

CERIF (Common European Research Information Format) is the EC-recommended data standard for modeling CRIS (Current Research Information Systems) data. The CERIF2006 1.1-FDM specification² is based on the entity-relationship (ER) model (Chen, 1976) and the relational database model, on which many popular database management systems like Oracle or MySQL are based. The CERIF2006XML 1.1-FDM Data Exchange Format Specification³ defines the CERIF XML-based interchange format.

Currently, EuroCRIS suggests using the CERIF2006FDM as the data model for a CRIS and CERIF2006XML only for data exchange between applications. In the following, we will discuss the advantages of using a XML-based data model as the basis for a CRIS. As we will show, the main benefit lies in the ease of modeling hierarchical or semistructured data which, among other things, makes it easy to generate websites and input masks out of the model without the need to elaborate mappings into other data models.

1.1 Characteristics of CRIS data

As can be seen in the CERIF specification, a CERIF-based CRIS stores information about four entities: persons, organisations, projects and publications. Each entity has a number of attributes and relations with other entities. Particularly with regard to websites that visualize CRIS data, this data can be viewed as hierarchical (tree-like); for example, a project consists of different persons who write different publications while working for different organizations. Alternatively, an organization consists of different persons who work in different projects.

1 <http://www.w3.org/TR/2006/REC-xml11-20060816/>

2 http://www.dfki.de/~brigitte/CERIF/CERIF2006_1.1FDM/CERIF2006_FDM_1.1.pdf

3 http://www.dfki.de/~brigitte/CERIF/CERIF2006_1.1FDM/CERIF2006XML_1.1.pdf

1.2 The relational (ER) approach

While the ER approach allows simple modeling of entities it gets quite complex when modeling the various relations between these entities; this can be seen in the CERIF specification where the majority of the text deals with these relations (“link entities”) while the four core entities are dealt with shortly. This aspect complicates the modeling process. Moreover, queries tend to get complex as multiple SQL joins have to be carried out to reconstruct the inherent structures.

According to (Abiteboul, 1997), relational data, which is typed, unordered and grouped in semantic entities with the same attributes, can be called structured data.

1.3 The semistructured (XML) approach

In contrast to the structured relational data, semistructured data (SSD) is data that is ordered, not strictly typed and where entities of one group can have different attributes (Abiteboul, 1997). SSD is often called self-contained or self-describing since there is no explicit need for a schema and type information can be omitted or placed directly into the data. Websites and hierarchical data can be described well by SSD; XML is a markup language to specify SSD.

In the XML approach, much of the above mentioned linking overhead can be done implicitly through the self-contained hierarchy. Another point is that this structure is much more human-readable than the relational table view.

Section 2 presents the advantages in more detail.

2 Benefits of XML-based CRIS

2.1 Simpler model

As stated above, CRIS software based on CERIF-XML is easier to create because of the reduced database programming needed in the document-oriented XML model. In the XML approach, much less resources have to be put in linking the entities via complex relations (i.e. SQL joins), making it also easier to read without generating reports out of relational tables.

It is imaginable that the primary structure of the XML tree hierarchy (e.g. the above example “a project consists of different persons who write different publications while working for different organizations”) is not adequate for a special use case (for example if the user wants to see a list of all organizations). In this case, special queries have to be carried out against the database, similar to the relational approach. Nevertheless, the underlying data model (i.e. XML) stays the same.

2.2 Website generation

One of the main benefits of a XML-based CRIS is that websites can easily be generated out of the XML data by using proven technologies like XSLT. The idea is to transform the XML into valid XHTML which then can be displayed in the web browser. Additional layouting

can be done by cascading style sheets (CSS)⁴. This approach separates the content (data) from design (CSS) consistently.

The inherent tree structure induces a navigational structure for the website which allows the user to navigate through the tree; for example, a user can “move” from project to project and navigate down to the involved persons and their publications. In the relational approach, the developer has to build this navigational structure (more or less) manually.

When generating web pages out of the model, the “granularity” of the page can be adapted; on one side, the complete tree could be displayed on one web page and on the other side, every node in the tree hierarchy (i.e. an XML element) could be displayed as a separate webpage. In real-world examples, it is reasonable to strike a balance between both extremes, i.e. usually it is a good idea to display medium-sized tree fragments on one page with the possibility to navigate between the fragments.

Furthermore, the use of XML eases the output in different output formats since there exists a large number of XML-based formats like WML⁵ (Wireless Markup Language for mobile devices), SVG⁶ (Scalable Vector Graphics for images) or the just mentioned XHTML⁷ which follows the “single source, multiple output” principle that is known and proven from the context of content management systems.

2.3 Automatic generation of input masks

When implementing a CRIS, a developer usually has to accomplish two tasks: on the one hand, he has to define and maintain the database schema and on the other hand, he has to implement a graphical user interface (GUI) for entering and changing the data.

In addition to the above mentioned generation of web sites, a XML-based data model allows the generation of input masks out of the underlying XML schema. The idea, which is presented in (Lay, 2007), is to generate Java classes directly out of the XML schema. By executing these Java classes, a user can then enter and change data. Another advantage is that the entered data can automatically be validated against the XML Schema, thus allowing only valid input.

This approach allows rapid prototyping during the process of implementing a CRIS or a website based on the CERIF specification since input masks for entering data can be generated automatically from the schema without the need of additional programming.

It should be mentioned that this holds only for simple data manipulation GUIs since special business logic has to be dealt with separately by implementing specialized software.

2.4 Metadata Harvesting

The XML approach also proves useful for metadata harvesting and (federated) search; data harvested in CERIF-XML format could be directly stored in its native form rather than converted to a relational model. For example, the OAI-PMH protocol⁸ is XML-based, thus the data could be easily transformed from CERIF-XML to OAI-PMH. Furthermore, RSS⁹

4 <http://www.w3.org/TR/REC-CSS2>

5 <http://www.wapforum.org/what/technical.htm>

6 <http://www.w3.org/Graphics/SVG/>

7 <http://www.w3.org/MarkUp/>

8 <http://www.openarchives.org/OAI/openarchivesprotocol.html>

9 <http://www.rssboard.org/rss-specification>

(Rich Site Summary), and its possible successor ASF¹⁰ (Atom Syndication Format) for specifying news feeds are also XML-based thus allowing generation out of the data model.

3 Technical Aspects

From the technical view, one benefit of an “all XML system” is that there is no need at any point to map or convert data, like from relational tables to HTML (for output) or to XML (for export). Since XML is used at all stages in the process, at the most some XSL transformations have to be carried out.

Unlike some years ago, today many proven database technologies and systems for XML exist: X-Hive, eXist, Tamino to name a few. These systems supply efficient storage and query mechanisms. For an overview on these systems, see (Bärisch, 2008). An in-depth comparison between native XML and XML-enabled database systems can be found in (Chaudri, 2003).

4 Conclusion

While the relational approach is good for modeling tabular data, it is often inadequate for modeling hierarchical and semistructured data (Abiteboul, 2000). Data of CRISs and especially of websites often have inherent semistructured and hierarchical characteristics. Moreover, relational databases have to make use of multiple tables to represent relations which can make the modeling process circumstantial. Using XML, semistructured data can be easily modeled in a tree structure, without the overhead of utilizing such „helper relations“. Another problematic issue in relational approaches is schema evolution; while the schema in semistructured databases can be changed relatively easy, it can become quite a complex task in relational databases.

The semistructured approach offers many advantages when it comes to visualize the data in different forms/formats. A theoretical background and some more examples can be found in (Lay, 2007).

On the other hand, an advantage of the relational approach, transaction security, is not that important in the CRIS context because CRISs (and their websites) are accessed mostly reading.

5 References

- Abiteboul, S. (1997). Querying Semi-Structured Data. In: Afrati, F and Kolaitis, P (Ed.): Database Theory-ICDT'97, Delphi, Greece, Springer.
- Abiteboul, S.; Buneman, P.; Suci, D. (2000). Data on the Web: From Relations to Semistructured Data and XML. Morgan Kaufmann.
- Bärisch, S. (2008): Technologies for CERIF XML based CRIS. Workshop EuroCRIS, Maribor, Slovenia, 2008.
- Chen, P. (1976): The Entity-Relationship Model – Toward a Unified View of Data. ACM Trans. Database Systems, 1 (1): 9-36.

¹⁰ <http://www.atompub.org/rfc4287.html>

- Chaudri, A.B.; Rashid, A.; Zicari, Z. (2003). XML Data Management. Native XML and XML-Enabled Database Systems. Addison-Wesley.
- Lay, P.; Lüttringhaus-Kappel, S. (2004): Transforming XML Schemas into Java Swing GUIs. In: Dadam, P.; Reichert, P. (Ed.): GI Jahrestagung (1), INFORMATIK 2004 - Informatik verbindet, Band 1, Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), Band P-50 der Reihe LNI, S. 271-276. GI, 2004.
- Lay, P. (2007): Entwurf eines Objektmodells für semistrukturierte Daten im Kontext von XML Content Management Systemen. Ph.D. thesis, Bonn, Germany.

Contact Information

Dr. Patrick Lay
Lennéstr. 30
53113 Bonn
Germany
patrick.lay@geis.org
<http://www.geis.org/staff/play>

Stefan Bärisch
Lennéstr. 30
53113 Bonn
Germany
stefan.baerisch@geis.org
<http://www.geis.org/staff/sbaerisch>