a methodology for metadata modelling - depth for a flat world

Andreas Aschenbrenner
ERPANET

Abstract: The recognition of the value of metadata continues to rise, and accordingly metadata frameworks are ever more widespread, they grow more comprehensive and they become increasingly complex. This rise in quantity, size, and complexity calls for a methodology that supports metadata design. Information modelling techniques as they are routinely employed in information system design and other domains are well suited for this task offering both visualisation techniques and an entire design methodology. Above all, information models help to manage complexity and leverage communication, thereby promoting reuse and interoperability. Already some metadata initiatives successfully employed modelling techniques building on the large body of existing experience in this area, yet these techniques remain to be widely adopted in the metadata world.

Information modelling techniques allow visualisation for intuitive interpretation and clear communication, facilitate a structured approach to design, and create new perspectives on existing metadata models. This paper describes the application of information modelling to metadata. It also provides orientation where the metadata community can further extend their modelling skills to create quality metadata models with a robust design.

Keywords: conceptual metadata model; metadata modeling; metadata visualisation; graphic modelling technique; information model; entity-relationship analysis.

1 introduction

Abstraction is probably one of the most powerful of human capabilities. Plans and models have assisted so many before us to coordinate hunting, to find the way, to develop novel tools, and sheer uncountable other activities. Today, modelling also manifests in a myriad of formalised methodologies and takes place in a variety of fields and situations. Conceptual models can take many forms: they can be text-based or employ graphics for visualisation; high-level or highly detailed; plain or hierarchically structured.

When composing a metadata set a host of requirements, influences and scenarios need to be considered. After all, metadata needs to be tuned to a specific business environment with all its activities, roles, and possible interdependencies. Such a complex undertaking calls for the use of formal modelling techniques that support the design process.

Metadata design starts from an initial

requirements analysis that explores the relevant business processes by establishing use cases and functional models (1). The actual metadata is then represented by a data or - as it is also called - information model, and its development comes as a natural succession to precursory analyses. Such a comprehensive approach ensures that all external requirements are accounted for, supports implementation at a later stage, and essentially raises the quality of the final product.

Taking a look at the available experience in the metadata community, this design process can be followed nicely is the "Functional Design for a digital depot" (2), where the actual metadata model is based on functional models of a comprehensive process analysis. In a similar vein, a core standards activity in metadata modelling, IFLA's Functional Requirements for Bibliographic Records (3), based their metadata model on requirements analyses and use cases. Another authoritative initiative that employs various modelling techniques and indeed is working on a data model with regard to digital preservation is the InterPARES project (4). The conceptual models of this huge, international project are part of their core deliverables, and surely they are also a great means for communication and discussion.

Before metadata-related initiatives picked up modelling techniques, these techniques were widely used in information system design. It is the experience accumulated in that domain that the metadata world can still benefit from. This paper refers to these experiences. It focuses on information models to illustrate the value of an actual metadata model. The description of the modelling technique in Chapters 3 and 4 is practically oriented and illustrates how a graphic metadata model can transport more information in a far clearer way compared to a flat text listing of a metadata set. Chapters 5 and 6 then explore the new perspectives such a model allows and reflect some extensions that may further enhance the power of this technique.

2 background

Information models take a prominent role in engineering and computer science. The success of a development project depends on the models created in the design phase. Respective techniques were first developed already with the advent of data processing systems in the 1950's. A proliferation of methodologies and tools followed, and modelling techniques are widely applied today. Graphic modelling techniques are employed to facilitate human interpretation as part of requirements analysis and conceptual design. These visual models are then translated into detailed lists suitable for implementation.

An early methodologies was the Structured Analysis and Design Technique (SADT) developed in the 1970s as a "language for communicating ideas" (5).

The family of Integrated Definition Languages, short IDEF, were first developed in the 1970s by the US Air

Force, and they are standard modelling techniques today. They cover a range of applications from functional to information modelling, simulation, object-oriented analysis and design and knowledge acquisition. Specifically, IDEF0 (6, 7) is a functional modelling language that builds on SADT, and IDEF1X (8, 9) provides for information modelling.

The development of the Unified Modeling Language (UML) (10) started in the 1990s building on the experience gained in a range of existing object-oriented analysis and design methods. UML is geared at combining a range of modelling techniques and provides a set of twelve model types including functional as well as information models. The Object Management Group (OMG), a non-profit consortium, coordinates the development of UML to create a rigorous, open standard for software modelling and system design.

The Extended Entity-Relationship (EER) model is a widely used conceptual information model, and has a long-lasting history in system engineering (12, 11). The EER found many proponents, some of whom introduced individual styles or extensions for specialised applications, but the basic concepts have hardly been obscured.

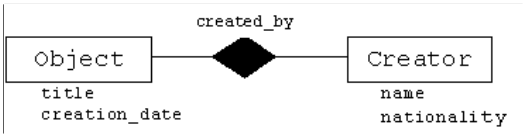
In the following we will use the well established and widely used EER to model metadata. Also the two metadata initiatives (2) and (3) introduced earlier employed entity-relationship models. However, the concepts of information models and the conclusions presented in the following are of a more general nature and it does not really matter with which modelling language they are notated. The abstraction mechanisms presented below can hence be applied with IDEF1X or any other information model as well.

3 the methodology

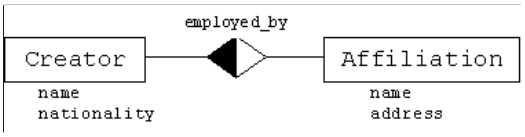
In order to understand the working of an EER and its application for metadata, we construct in the following a small learning example in a step-by-step process. We start from a metadata set for any sort of digital object comprehending four elements:

```
object
title
creation_date
creator
nationality
```

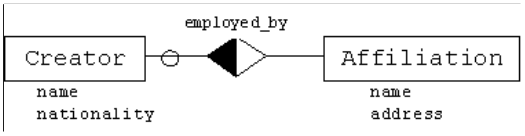
Clearly this metadata set is actually composed of two different entities: there is the actual 'object' described by the *title* and the *creation_date*, and then there is the 'creator' of the object of whom we know the nationality. These two entities are in a certain relation to each other: an object is created by one or more creators, and each creator may have created one or more objects. In EER notation this is called a many-to-many relation, and is notated like this:



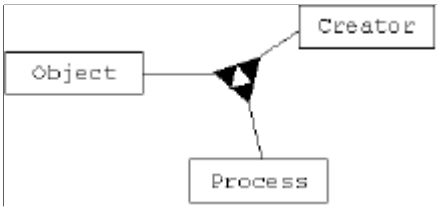
Other connectivities of relations are a one-to-one relation or a one-to-many relation. To demonstrate a one-to-many relation, let us assume that each creator may be working for exactly one organisation, but each organisation may contract many creators:



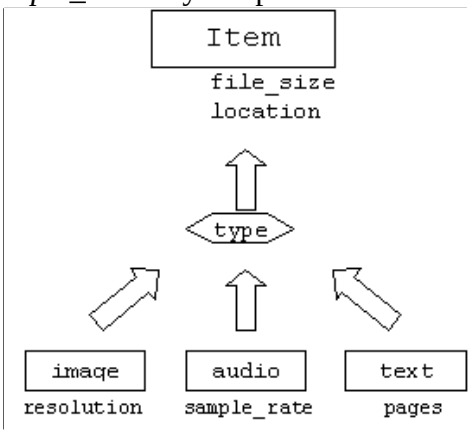
One step further we define that each affiliation may contract zero, one, or any number of creators, whereas a creator must be assigned to exactly one organisation. For this reason we introduce a circle signifying 'optional' on the side of the 'creator' entity, whereas we leave the relation on the side of the 'affiliation' unchanged.



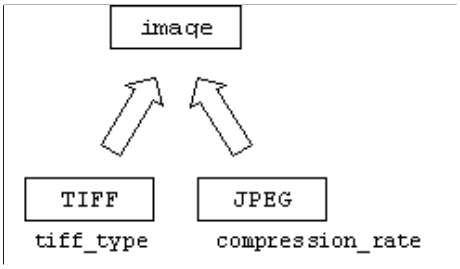
Lastly, a more sophisticated kind of relation is the ternary relation. The above relations were largely binary ones between two different entities. Unary relations are between one instance of an object and another instance of the same object. Ternary relations, consequently, involve three different objects. For the following example we assume that each object is created through a number of processes conducted by each creator, and also that the creator may actually perform the same processes on various objects. This calls for a many-to-many-to-many relationship.



Turning our focus to the entity 'object', we find that each object may consist of any number of items, also just one at least. Each item has certain attributes such as *file_size* and *location*. Furthermore, each item is of a certain *type*, for example an 'image', 'audio', or 'text'. These item types are specialisations and may have attributes of their own; for example, an image has a *resolution*, whereas for an audio the *sample_rate* may be specified.



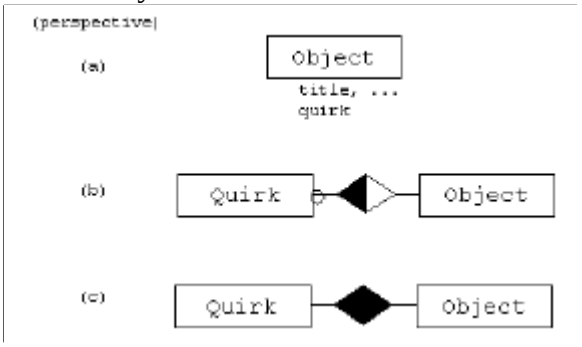
Each of these items can be stored in a certain file format. An image, for instance, may be stored as a TIFF or as a JPEG. These specific formats may again have attributes of their own, similar to the generalisation/specialisation above. The following notation called subset, however, allows overlapping entities. In other words an image may be stored as a TIFF, as a JPEG, or both formats at the same time.



This basic toolset of the EER model can be applied for modelling any conceptual metadata framework. The above is a synopsis of the original discussion in (12) and it is translated to modelling metadata. Relevant in this context is mainly Step 1 described in Chapters 1 and 2 in that paper. For a more extensive description please refer to that paper, or to any of the numerous tutorials and information sources available online and via other channels.

4 application

Information models do not provide a mathematically sound technique which allows only a single possible solution for a specific metadata model. They rather help to understand and shape a task; they help to ask the right questions and provide a typology for communicating possible answers. Indeed, distinct perceptions of a single metadata set may manifest in small but eye-catching differences within the model. As an example to this we assume that in the tentative metadata model we started to create above, we want to describe possible 'quirks' of 'objects', i.e. deficiencies in the original objects. This time, however, we flip the procedure: first we look at possible descriptions of this requirement in EER notation followed by a discussion.



In the first description the quirk is an attribute of the entity 'object'. In other words, quirks are described in a short text, which is attached to the very object. Approach (b), however, assumes that each object may have any number of distinct quirks including none. Yet another perspective is (c), where various objects can have the same quirk or any number of quirks. If in a specific environment a set of recurring quirks can be identified, predefined quirk descriptions could be quickly assigned to newly incoming objects, which may raise efficiency considerably. As illustrated by the Figure above, differences between the models are quite evident, whereas in a flat textual description of the metadata set these differences may go unnoticed and may cause misunderstandings and wasted efforts.

Visualisation is one key feature of a conceptual model. A visual model further facilitates organising and structuring the data at hand through multiple abstraction levels and modularisation. Ultimately this is conducive to efficiency in the model's practical application. To illustrate the value of modularisation and

aggregation, we specify a specific quirk closer by describing the technology environment (i.e. the hardware and software platform) it occurred in. In another part of the model, the description of the process an object was created in, we plan to include this technology description as well. By making 'technology' an entity of its own, we can establish a relation between it and the 'process' as well as the 'quirk' object. As technology descriptions can be expected to be rather repetitive, this bears huge time and cost savings in practice. Creators are in this approach not obliged to produce a technology description themselves, but they just select given descriptions, which are reused by the people creating the descriptions of possible quirks. (Note also that this approach could not be realised if perspective (a) in the Figure above was chosen.)

Information models can be taken down from rather sketchy high-level models to detailed descriptions of future functionality. In a further step after the model has been established, the types of each attribute should be specified. Consistency requirements for attributes should be established as well. For instance, the attribute 'CreationDate' from the entity 'object' should be a date somewhere between the birth and the death dates of the creator. Exact type definitions and consistency rules will further enhance the quality of metadata.

5 implications

Groupings in the metadata model offer further opportunities for optimising system implementation and streamlining user tasks. For instance, instead of requesting that all attributes of an entity need to be entered again and again, the system could automatically complete the empty attributes once it uniquely identified the corresponding dataset. Looking at the entity 'creator', once the creator's name has been given, and there is only a single creator with that name known to the system, the birth and death dates as well as the nationality could be filled in automatically. As obvious such a semi-automatic approach may appear, it requires the modularisation of data to make it possible in the first place. More than that, modularisation in conceptual modelling is the basis for allowing multiple creators of an object. Again this is obvious to humans, yet these concepts and relations can only be created with adequate data structures at their basis. Picking up the example above, the system would just not know which nationality belongs to which author, or who died if merely two authors and one date of death were given in the form of a plain metadata list.

Modularisation in metadata design may also reflect the organisational roles and responsibilities for metadata creation. Let's take the 'creator' entity again: somebody needs to enter a creator's data first. Should the person who enters the metadata be the creator herself, or should this be taken care of by the organisation? Going one step further, the task for populating the metadata of a specific module within a metadata framework

could actually be outsourced to a dedicated service. A whole sector could decide to join forces and create such a service that all organisations in the sector can make use of. In the case of the 'creator' entity, this step has been taken in the cultural heritage sector by a European project called LEAF (13). The LEAF system contains authority files that describe specific persons. LEAF can indeed be used as a service responsible for one particular module in the metadata framework of a library, as it allows external resources to link to its authority files.

Another example for exploiting possible synergies is a File Format Registry that holds a comprehensive catalogue of file format documentation. Instead of providing all the object type information itself, an organisation may rather entrust this task to an external service. The Global File Format Registry (14) is setting up exactly such a service. One of the open challenges is with the unique identification of entries in this database, so that specific organisations can actually reference to and establish a relation between their metadata model and the registry.

What other such services are conceivable, on a corporate, sectoral, or even a global level? Information models for metadata may help to answer this question and thereby highlight opportunities for cooperation, which essentially saves costs and raises quality.

6 extensions

Several extensions to existing conceptual modelling techniques are conceivable that could tune them specifically for modelling metadata. One convenient extension might be a **colour coding** for attributes to convey how rigidly their types are defined: on the one end are just textual fields that allow any conceivable input, on the other are clearly specified data types such as the ISO 8601 date format (15). Tightly defined data types, of course, allow a certain level of machine comprehension, which is conducive to their automatic handling. One may distinguish three levels of typing: (1) machine comprehensible; (2) strongly typed; and (3) human understandable. Each of these levels could be assigned a colour, so that the viewer is able to discern at first sight, which attributes can be interpreted by the system (1); those which can to some degree be automatically handled and manipulated, but are essentially meaningless to the system (2); and those which are unstructured strings and unfit for automatic handling (3).

Another extension may be required on a higher conceptual level: increasingly complex relationships between different metadata sets are being established, such as certain attributes that are part of various metadata sets at the same time, or virtual attributes that are created automatically and on the fly from other metadata. Technologies such as Application Profiles (16) foster these pioneering developments. Modelling such complex relationships may, however, require extended modelling techniques.

Same applies for including a **temporal component** in metadata modelling, the importance of which was underlined in (17). Temporal modelling techniques have already been analysed and introduced for various conceptual modelling techniques, including EER (18) and also for the IDEF family (14) – no need to reinvent the wheel if these existing techniques are applicable.

Previous research, albeit in the field of system engineering, has also focused on **quality** related issues. The different background notwithstanding, the findings of this research are transferable. For example, Daniel Moody (19) identified model quality factors and established a quality management framework for both the model as such and the process of modelling. The framework takes different roles in the modelling process into account, and should lead to a model that can be implemented and is complete, simple, flexible, integrated, and understandable.

An important design goal for metadata models obviously is their **long-term stability**. While time will inevitably make certain adaptations necessary, the model can be designed such that it is robust despite necessary modifications. Lex Wedemeijer (20) researched the long-term evolution of conceptual schema. His findings leverage the simplicity, extensibility, and essentially the stability over time of conceptual schema. The transfer of these findings will be particularly valuable, for quality and long-term stability are central concerns in metadata modelling.

With more and more initiatives embarking on metadata modelling, techniques are needed to **compare**, combine and reuse models. As (21) in the scope of the "Guidelines of Modeling GoM" project pointed out, however, modelling is an essentially creative and subjective act. The exactly same requirements may therefore be captured into entirely different models by different designers. Guidelines such as those introduced by GoM (21) and by the quality frameworks referenced above potentially reduce subjectivism in the design process to such a degree that model comparison is possible. However, any more automatic approach to comparison of a large body of models such as suggested in (22) is probably still out of reach for metadata models, due to the inherent heterogeneity of backgrounds and the often incompatible terms and perspectives of metadata initiatives.

7 conclusions

Information modelling techniques applied to metadata allow quick comprehension, and help keeping track of large and complex models. Even newcomers will find the interpretation of such a metadata model straight-forward, and will appreciate it as a means for communicating the foundation concepts of a specific model. Some experience is necessary in order to best exploit the features and opportunities modelling offers in the design phase. However, the wealth of experience and resources available in other fields supports a steep

learning curve. Tutorials and auxiliary software tools are easily found.

Information modelling is one chain link in the design process. Before the metadata model can be created, a requirements analysis needs to explore the real-life environment and the tasks and responsibilities the model has to support. After the model is finalised, it needs to be taken forward to implementation. The close connection between information models and software engineering helps bridging the gap between conceptual metadata models and system implementation. Methodologies for translating models into database schema are widely available. Desirable for metadata is a methodology for translating a conceptual metadata model into an XML/RDF framework. In fact, an information model was designed for RDF (23), which is tailored to the specific use and very concrete. However, ways to translate conceptual models into the RDF data model remain to be explored.

Another area that calls for research is the comparison of metadata models, which is particularly needed for tools and services such as metadata registries. Metadata registries (24) continue to be a hot topic for their promise to enhance the discovery and reuse of existing metadata definitions. Information modelling techniques may support metadata registries by reducing idiosyncratic model features through the application of modelling quality frameworks. This promotes more canonical models, in order to avoid different models being created for essentially the same requirements. At the same time metadata models facilitate the communication between different communities and thereby foster the reconciliation of the variety of concepts and notions inherent in the diverse backgrounds of metadata initiatives.

Over all, there is little doubt that this support in conceptualisation and communication, as well as their fostering quality, efficacy, and robustness suggest the adoption of modelling techniques and make them more consistently used in the metadata world.

References

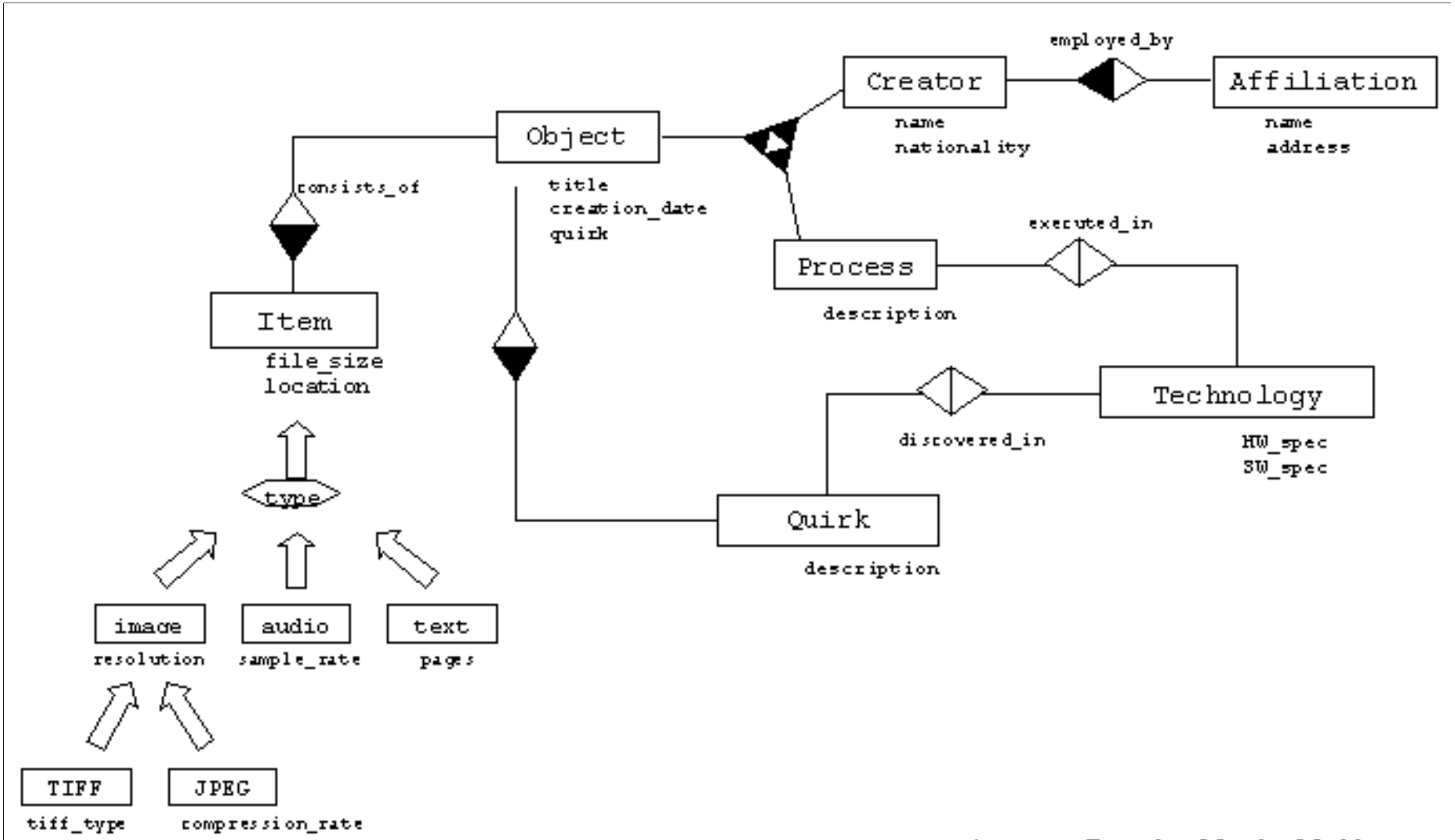
1. J.García Molina, M.José Ortín, Begoña Moros, Joaquín Nicolás, and Ambrosio Toval. Towards Use Case and Conceptual Models through Business Modeling. In: A.H.F. Laender, S.W. Liddle, V.C. Storey (Eds.). ER2000 Conference, LNCS 1920, pp. 281-294, 2000.
2. Nico van Egmond, Hans Hofman, Jacqueline Slats, Tamara van Zwol. Depot 2000 - Functional design for a digital depot. Rijksarchiefdienst, Den Haag, 2000.
3. IFLA Study Group. Functional Requirements for Bibliographic Records. UBCIM Publications – New Series Vol 19; September 1997. ISBN 3-598-11382-X.
4. Bill Underwood. The InterPARES Preservation Model: A Framework for the Long-Term Preservation of Authentic Electronic Records. In: Toblach/Dobbiaco. Choices and

Strategies for Preservation of the Collective Memory. Italy, 25-29 June 2002.
Also see the website of the InterPARES 2 project (International Research on Permanent Authentic Records in Electronic Systems). http://www.interpares.org

5. D.Ross. Structured analysis: A language for communicating ideas. In: IEEE Transactions on Software Engineering 3(1). Special Issue on Requirements Analysis. (1977), pp16-34.
6. Keith McConnelly (US Department of Defense). Introduction to IDEF Modeling: Function and Information Modeling. http://www.defenselink.mil/nii/bpr/bprcd/0066.htm
7. National Institute of Standards and Technology (NIST). Integration Definition for Function Modeling (IDEF0). Federal Information Processing Standards Publication 183 (FIPS PUBS). December 1993. http://www.itl.nist.gov/fipspubs/idef02.doc
8. National Institute of Standards and Technology (NIST). Integration Definition For Information Modeling (IDEF1X). Federal Information Processing Standards Publication 184 (FIPS PUBS). December 1993. http://www.itl.nist.gov/fipspubs/idef1x.doc
9. IEEE 1320.2-1998 - IEEE Standard for Conceptual Modeling Language-Syntax and Semantics for IDEF1X97 (IDEFobject) - Description. http://standards.ieee.org/reading/ieee/std_public/description/se/1320.2-1998_desc.html
10. Object Management Group (OMG). Unified Modeling Language. http://www.omg.org/um
11. P.P.Chen. The Entity-Relationship Model: towards a unified view of data. ACM Transactions on Database Systems; v1, n1; (March 1976). pp 9-36
12. Toby J.Teory, Dongoing Yang, James P.Fry. A logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model. ACM Computing Surveys, v18, n2; (1986). pp 197-222
13. Max Kaiser, Hans-Jörg Lieder, Kurt Majcen, Heribert Vallant. New Ways of Sharing and Using Authority Information - The LEAF Project. In: D-Lib Magazine (ISSN 1082-9873), November 2003; Volume 9, Number 11. http://www.dlib.org/dlib/november03/lieder/11lieder.html
14. Global Digital Format Registry (GDFR). http://hul.harvard.edu/gdfr/
15. ISO 8601. Data elements and interchange formats - Information interchange - Representation of dates and times. ISO TC 154 (International Organization for Standardization, Technical Committee).
16. Thomas Baker, Makx Dekkers, Rachel Heery, Manjula Patel, and Gauri Salokhe. What terms does your metadata use? Application profiles as machine-understandable narratives. In: Journal of Digital Information

2(2), November 2003. http://jodi.ecs.soton.ac.uk/Articles/v02/i02/Baker/

17. Results of the Harmony Project. http://www.metadata.net/harmony/. Of particular interest in this context: Carl Lagoze, Jane Hunter. The ABC Ontology and Model. In: Journal of Digital Information 2(2), November 2003.
18. Heidi Gregersen, Christian S. Jensen. Temporal Entity-Relationship Models - A Survey. In: IEEE Transactions on Knowledge and Data Engineering, May 1999; v.11, n.3, pp 464-497.
19. Daniel L.Moody, Graeme G.Shanks. Improving the quality of data models: empirical validation of a quality management framework. In: Information Systems 28(6), 2003; pp 619-650; ISSN 0306-4379.
20. Lex Wedemeijer. Long-term evolution of a conceptual schema at a life insurance company, Annals of cases on information technology, Idea Group Publishing, Hershey, PA, 2002.
21. Reinhard Schuette, Thomas Rotthowe. The Guidelines of Modeling - An Approach to Enhance the Quality in Information Models. In: T.W. Ling, S. Ram, and M.L. Lee (Eds.). ER'98, LNCS 1507, pp. 240-254, 1998. Springer-Verlag Berlin Heidelberg 1998.
22. S.Castano, V. de Antonellis, M.G.Fugini, B.Pernici. Conceptual schema analysis: techniques and applications. In: ACM Trans. Database Systems 23(3); 1998; pp 286--333; ISSN 0362-5915.
23. Eric Miller. An Introduction to the Resource Description Framework. In: D-Lib Magazine (ISSN 1082-9873), May 1998.
24. The Dublin Core Metadata Registry. http://dublincore.org/dcregistry/



APPENDIX - Exemplary Metadata Model