Communications of the IIMA

Volume 7 | Issue 2 Article 6

2007

Enterprise Documentation: A Formal-Model Approach

Xin James He
Dolan School of Business Fairfield University

Myron Sheu
Department of Information Systems & Operations Management California State University Dominguez Hills

Follow this and additional works at: https://scholarworks.lib.csusb.edu/ciima

Recommended Citation
He, Xin James and Sheu, Myron (2007) "Enterprise Documentation: A Formal-Model Approach," *Communications of the IIMA*: Vol. 7: Iss. 2, Article 6.
DOI: https://doi.org/10.58729/1941-6687.1034
Available at: https://scholarworks.lib.csusb.edu/ciima/vol7/iss2/6

This Article is brought to you for free and open access by CSUSB ScholarWorks. It has been accepted for inclusion in Communications of the IIMA by an authorized editor of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.

Enterprise Documentation: A Formal-Model Approach

Xin James He
Dolan School of Business
Fairfield University
xhe@mail.fairfield.edu

Myron Sheu
Department of Information Systems & Operations Management
California State University Dominguez Hills
msheu@csudh.edu

ABSTRACT

Most digital documents within an enterprise reside outside traditional databases. Search engines have been used to retrieve these digital documents but the results are often unsatisfactory. Digital libraries yield search results with greater precision than general search engines and are capable of compartmentalizing a wide variety of digital documents. However, digital libraries cannot subtly control application-specific document retrievals because they lack a formal model for recognizing the intrinsic relationships among digital documents. This research proposes a formal model, the **Entity-Oriented Enterprise Documentation Model** (EOEDM), to facilitate document retrievals for enterprise applications. This EOEDM is a meta-level modeling schema, exploiting the entity-relationship model in database design as well as the object model in application design. Since enterprise documents usually result from business activities, the proposed model targets the enterprise documentation needs and relies on the patterns of business activities to associate and locate digital documents with higher precision and better recall.

INTRODUCTION

Information assets of an enterprise are contained in traditional databases as well as in documents of various formats. While much progress has been made to use the information in traditional databases for enterprise applications (Fulker et al., 2001), most information assets are still contained in documents outside the databases and are often underutilized. These documents are mostly unstructured and can be found in file servers, tapes, workstations, and even file cabinets. Attempts have been made to develop documentation systems to locate such documents and to discover them in groups for business applications.

Search engines, such as those on the Internet, provide the abilities to locate information on the web. However, natural language is highly ambiguous and therefore, more often than not, users of Web search engines are overwhelmed by the enormous number of returned results. In comparison to search engines, digital libraries yield higher precision and better recalls. Since the recalls measure how well a search system finds what one wants and the precision measures how well it weeds out what one does not want, the recalls and precision are more critical for enterprise applications than for Web search engines. However, digital libraries, in general, lack a theoretical information model (McGray & Gallagher, 2001; Cooper, Coden, & Brown, 2002). Searches within digital libraries are typically on titles, keywords, and abstracts, but they do not take advantage of the relationships that exist among documents. Browsing in digital libraries is generally following subject categories and thus can utilize only limited relationships among the documents (Geffner et al., 1999). A noteworthy trend in digital libraries is to incorporate multiple forms of digital information objects (both structured and unstructured data) into a single digital library. An example of such a digital library is the NCSTRL+ (Nelson et al., 1999), which implements a concept called *buckets*. Buckets construct object-oriented containers, which can collect, store, search, and transport a logical group of digital

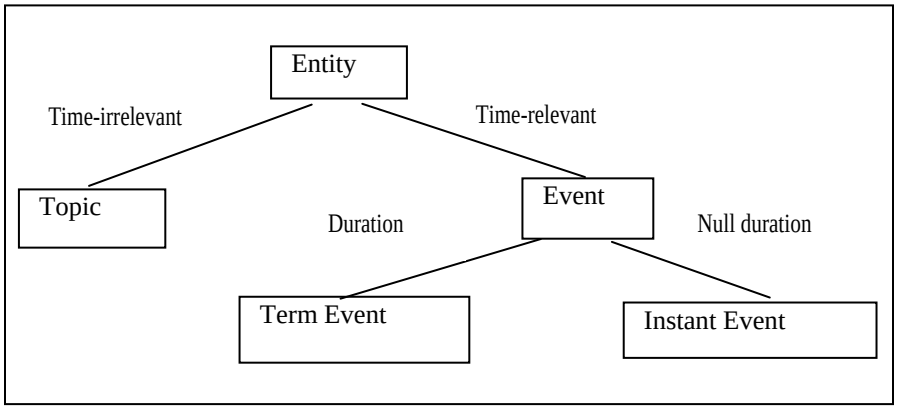
documents as a single unit. Nevertheless, buckets are commonly created by the authors of the related digital documents. Consequently, they are constructed for the original authors, not necessarily for information inquirers.

Additional search methods may also associate digital documents in terms of content, rather than key words and/or categories. Examples of such search methods include content-based queries (Florescu, Levy, & Mendelzon, 1998), semantic matching in a hierarchical search (Janée, Frew, & Valentine, 2003), and content-based indexes on a variation of the primary theme (Medina, Smith, & Wagner, 2003). Other attempts to model the relationships among digital documents include Relationship Management Model (Isakowitz, Stohr, & Balasubramanian, 1995), Relationship-Based Analysis (Yoo and Bieber, 2000), Content Manager by IBM to manage heterogeneous and unstructured data (Somani, Choy, & Kleewain, 2002), and XML Topic Maps (XTP) for retrieving the knowledge contained in unstructured data (TopicMaps, 2004). All these methods, however, also emphasize on syntax-oriented modeling and general-purpose relationships without considering meta-level business specifics. Therefore, they all lack the flexibility to discover and manage application relationships in a business context that exists among digital documents. As these systems deal with large scale digital documents, they could either restrict a very narrow range of digital documents due to imposed syntax and/or semantic matches or could not control the relevance of clustered retrievals due to an unspecified business environment. Having perceived the viability of pattern-oriented search, Grossmann, Hudec, & Kurzawa (2004) call for creating a data warehouse to describe the contents of documents accessible over the Internet. Similarly, Guelfi & Pruski (2006) use the ontology for optimal representation and exploration of the web, but the complexity of the resulting model would increase exponentially should it be applied to a real-world application.

THE PROPOSED ENTERPRISE DOCUMENTATION MODEL

In light of enterprise needs for retrieving digital documents, we have developed a formal model, the **Entity-Oriented Enterprise Documentation Model (**EOEDM**)**, to address the weaknesses of the existing search methods amid enterprise documentation needs. By dynamically circumscribing the application-specific nature of the current enterprise documents, the EOEDM takes a formal-model approach to locating the associated documents, which leverages both the entity-relationship model (Chen, 1976) for database design and the object model (Minsky, 1975) for application design. It is called entity-oriented because each *Entity* in the model belongs to a class that refers to a specific type of digital documents while entities within the model do not necessarily inherit and instead they may bond through a variety of relationships, such as a sibling relationship. For simplicity, such a model can have only two entity classes as shown in Figure 1. The first class, *Topic***,** includes the entities with which temporal information is irrelevant. The second class, *Event***,** includes the entities for lifetime. In an extreme situation, it is possible for certain event entities to have a null duration. In other words, the starting and ending times of these events may not be meaningfully distinguishable, but the occurrence time of such an event could be essential to some applications. Such a differentiation improves precision and recall of a search within an enterprise since it effectively separates instantaneous events from production activities as companies become progressively project orientated. Instant events become immediately historical and thus are appropriate to be stored in archives while documents related to ongoing projects are usually stored in operational systems.

Figure 1: Classification of document entities.



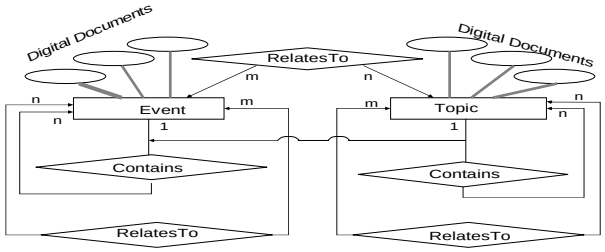
Because a traditional entity relationship (ER) model can only accommodate flat relationships, we incorporate object orientation in the EOEDM to handle ancestral relationships that should readily exist among enterprise digital documents. Oftentimes, a set of digital documents could be taxonomic and offer more value when reviewed in conjunction with other documents in closely or remotely related fields. For example, we had an event instance such as "02/27/2007 Stock Market Crash." Many documents must have been generated by the event. The digital documents that were considered subsequent to this event would likely result in additional digital documents. In such a case, a traditional ER model that does not support a progressive navigation would result in a large collection of documents such that targeted documents may easily be submerged. The EOEDM, in contrast, can navigate along a chain of clusters through which only digital documents of interest will be selected by incorporating the object orientation. Specifically, if an impacted enterprise, such as a brokerage firm, uses the EOEDM to describe the relationships between the event and many internal events that occurred thereafter, the digital documents resulting from subsequent events can be located by considering their impact on such topics as organization, finance, and product line. Some or all of these topics can be bound together in an order determined by customized criteria through an information acquirer or by default criteria in terms of, for example, relevance, size and immediacy.

With the EOEDM, enterprise document collections are represented by entities rather than by digital documents themselves as in digital libraries. Moreover, to cluster digital documents across multiple business domains within the enterprise, the EOEDM can take into consideration that enterprise document collections likely reflect internal business patterns. As an example, a concept-exploration project in an aerospace company may render various kinds of digital documents such as calls for proposals, bidding proposals, offers, progress reports, prototype results, manuscripts, and datasets. These closely related documents may be scattered across multiple departments and stored in different formats. However, their locations should coincide with their corresponding business patterns such as where and when these business processes have occurred. If properly circumscribed (Genesereth & Nilsson, 1987, p. 134) according to relevant business patterns, the documents can be clustered to facilitate retrievals.

In the EOEDM, entities are searched for information retrievals. Since entities can in turn contain other entities, the relationships among the entities lead to granulating primitive entities in terms of their scope, duration, and profundity. Once a search reaches a desired level of abstraction, all digital documents at and below that level can be examined. Note that an entity could represent a specific aspect of a business activity, that its sibling entities could represent the same aspects of related business activities, and that they could together compose a business subject or a business event. Additionally, some entities may be more stable than others. For example, a project may have to follow a set of business processes, but the stakeholders of each process may remain stable and thus can be leveraged to link these related documents. By collecting documents through various links that are tied to business patterns of an enterprise, the EOEDM could better support both recalls and precision in business applications due to the fact that it does not rely on statistical hits or syntactic keywords to retrieve documents, but rather on certain shared aspects of business activities.

Figure 2 illustrates an underlying schema that EOEDM supports, consisting of only two primary relationships, *Contains* and *RelatesTo*, with which we can create a hierarchy of entities if necessary. The *Contains* is unidirectional and has a one-to-many relationship for *Events* and a many-to-many relationship for *Topics*. The one-to-many restriction on *Event* entities would help enforce temporary boundaries whereas the many-to-many relaxation on *Topic* entities would help collect digital documents across multiple disciplines. Syntactically, in rare cases the EOEDM would allow *Topic* entities to be restricted to a one-to-many relationship and *Event* entities to be relaxed to a many-to-many relationship. But the EOEDM would not allow both entities to have the same one-to-many or many-to-many relationships simultaneously, due mainly to the concern that the search results would otherwise be either too isolated or too redundant. As to the next level of association the *RelatesTo* is defined to connect entities in terms of many-to-many relationships. Since an *Event* is time relevant but a *Topic* is time irrelevant or simply infinite in duration, only a *Topic* can contain *Events* but not the other way around, even though they may be related to each other. This design is significant in terms of preventing circular relevance. In EOEDM the *RelatesTo* relationship features a degree of relatedness that can be determined by a domain expert as a default setting and to be overridden dynamically by an information acquirer.

Figure 2. The entity-oriented documentation model.



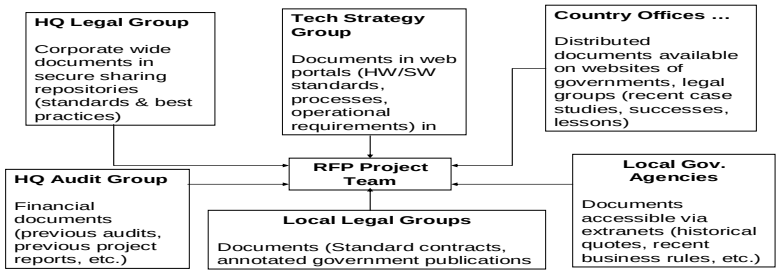
To show how the EOEDM is capable of facilitating information retrievals for enterprise applications, let us consider the distribution and applications of digital documents with an example at a Far East regional division of a USA consulting firm. The IT personnel at the division have a high turnover rate and experts from the headquarters in USA have served on short but frequent assignments to help. This regional division faces a constant challenge in building an effective documentation system in order to manage, search, and retrieve widely dispersed, unstructured data that are generated over time by a dynamic workforce. While such surrounding knowledge repositories could be catalytic for daily operations and project undertakings, potentially useful documents are unfortunately scattered across isolated digital repositories and the task of tagging these documents for collective retrievals is either totally overlooked or miscellaneously remedied for individual needs.

As one of the major service responsibilities, the division often has to prepare a request for proposals (RFP) on behalf of one of its clients, soliciting for bids to implement a customized ERP system. Due to intensive information collection with relevant enterprise documents from various sources throughout the entire company as well as the surrounding business environment, the whole bidding process usually requires a small project team to work intensively for a period of six months.

As shown in Figure 3, the RFP process typically requires access to documents from various sources: Documents from the *Tech Strategy Group* in the headquarters contain the latest technical standards, architectures and processes for hardware, software and business models; documents from the *Country Offices* contain recent case studies of similar implementations provided by various vendors in other parts of the world for comparison and analysis; and documents on new system enhancement requests from various user groups over the past three years must be compiled. Other inputs from a variety of local organizations (finance, sales, and customer service groups) should also be collected as valuable references. An RFP project team must take into consideration all these documents in order to produce a quality request. For instance, conflicts among legal/audit requirements from the headquarters and

local offices often exist and must be brought up in the RFP document. Therefore, the real challenge is that the involved parties store and organize their documents in various digital stores, such as workgroup file shares, Lotus Notes databases, or even departmental web portals. In the past, the lack of an effective search and retrieval system for required information has resulted in a lengthy and costly RFP process, which has been a serious hindrance for such a time-sensitive business process and incurred many opportunity costs.

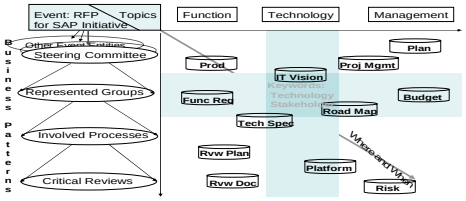
Figure 3: A typical collection of digital documents required for a RFP project.



Given the challenges to the regional division of the consulting firm, let us see how the proposed EOEDM may come into play. With the EOEDM, an organizational structure and/or a project development structure could be leveraged to identify and associate relevant digital documents. At the division level, this organization is quite project-oriented and uses a matrix structure to organize its personnel rather than a traditional hierarchy. However, since it is a global firm, at the headquarters level, it remains hierarchical and maintains that its regional divisions can choose its own organizational structures. By the same token, a project development cycle may follow a traditional waterfall pattern, a spiral pattern, or a combination of both. In the case to which we apply the EOEDM, the *RFP Project Team* creates an *Event* entity called *RFP* for the SAP initiatives and gather digital documents under the *RFP*. The root entity may contain more than one *Event* entities, such as a *Steering Committee* entity that affiliates all stakeholders of the RFP project and a *Project Cycle* entity that brings together all the phases of the system development process. Then we can dynamically instantiate the relationships of relevant entities and thus enable the search and retrieval of these documents in clusters by using these entities to describe relevant metadata, such as participants and development phases. Next, the *Event* entities can be attached with more *Topic* entities. For example, given a predefined company policy for the role of each involved stakeholder in a proposal process, the file repository for the technology strategy group is linked through a *Topic* entity that covers all the hardware standards for a SAP implementation. Such a *Topic* entity is linked to a hierarchy of *Event* entities that include the *Steering Committee* entity. By the same token, if someone from the headquarters sits on the steering committee and speaks on behalf of the corporate legal department, the location for storing the current company-wide legal contract standards applicable to such a project could also be linked to the *Steering Committee* entity. The facilitation of the EOEDM could become effective also when the relationship between different domain entities is captured by constructing both the *Topic* and *Event* types of entities from the stakeholder aspect of an RFP process, such as critical reviews as an *Event* and review criteria as a *Topic*. If the history of implementation is of interest in a search, another *Event* entity, named *System Cycle*, as opposed to *Project Cycle*, can be created so that all project cycles that the SAP system went through can be connected.

Figure 4 shows that additional entity instances can also be created to narrow the search scope in terms of such decisions as department vs. corporate or function vs. technology. With such a hierarchy of entities as search guidelines, the EOEDM supports drill-down retrievals of documents, ranging from certain related business domains to certain business activities.

Figure 4: An *Event* entity, *RFP* for SAP Initiatives, may include other *Event* and *Topic* entities with enterprise-specific business patterns.



Selecting and creating appropriate entities to bind digital documents can be determined by individual domain experts since the EOEDM is merely to provide a meta-level model. In reality, a domain expert may connect documents by creating different *Topic* and *Event* entity instances to reflect other business activities. In the case of this consulting firm, its product lines could be stretched to cover several physical facilities that may be thousands of miles away from each other or may simply have no tangible facilities. In choosing what entities to be included in the documentation model and in what order to bind them, a domain expert may want to consider the stability of the underpinning business patterns. For example, an enterprise's business model and organizational structure should remain quite stable and can, therefore, serve as durable backbones in collecting documents for long-term enterprise applications. Please note that the separation between *Event* and *Topic* entities plays a crucial role in support of the applicability of the EOEDM. Events could pass into history whereas topics should stay for continuity. Since much more digital documents are generated by events than by topics, such a separation in the EOEDM keeps the amount of information to be navigated manageable. Finally, the use of keywords is not excluded by the EOEDM but is helpful only when digital documents are collected based upon a chosen cluster of *Event* and *Topic* entities.

In contrast to the approach taken by the Defense Advanced Research Projects Agency (2004) in developing an automatic topic detection and tracking system, our approach is much business relative in a sense that it specifically targets organizations that have most of their document collections stored according to their business patterns, including but not limited to application domains, operation processes, organizational structures, and project sponsorships. In most organizations, the document entities and their relationships correlate with some business patterns because they are the results of business activities, which are organized according to business norms.

FROM A MODEL TO A SYSTEM

The EOEDM can be implemented based upon various levels of sophistication. It is important to understand that the effectiveness of the EOEDM relies heavily on the protocol of interactions among the involved parties. To show how a minimal set of functions and roles work together to materialize the usefulness of the EOEDM, below we have developed a prototype system for the EOEDM, **Entity-Oriented Enterprise Document Information System** (EOEDIS).

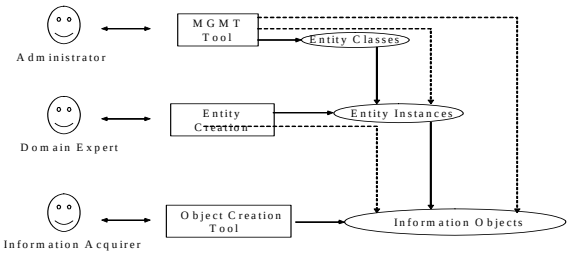
The Architecture of EOEDIS

Our prototype incorporates the basic digital library technology, expands the functions of the interactive tools (Bainbridge, Thompson, and Witten, 2003), and brings about additional utilities for building and maintaining entity-oriented indexes for better precision and recalls. We have deliberately designed the tool set to differentiate the roles involved in the development and deployment of the EOEDIS for optimal cooperation.

The viability of the EOEDM lies heavily in its operational protocol through which the primary players construct and maintain an enterprise documentation model. Figure 5 illustrates the high-level functions and their roles in the

system. The system management role, for example, uses the *Management Tool* to set defaults and to control all the contents of the system, including authorization and security. The *Domain Expert* role, which may be played by a corporate librarian familiar with internal business processes, uses the *Entity Creation* update tool to create, structure, and revise entities. Collection acquirers or data object authors use the digital information *Object Creation Tool* to submit digital information objects for publishing to the EOEDIS.

Figure 5: The operation structure of EOEDIS.



Once implemented, the data model for the EOEDIS should include the typical digital library metadata with a library engine that generates entity-oriented indexes for efficient search, navigation, and access to digital information objects. Navigations using specific business patterns could be made controllable by creating additional layers of connectivity. Such a tiered navigation strikes a balance between accuracy and performance while it renders the same degree of recalls due to the use of a formal model.

Implementation Issues of the EOEDIS

The prototyped EOEDIS is intended to be quite general and, in a real application, should be further customized to leverage specific business patterns of an organization. In the prototype, behind a set of tools in support of various roles is a user interface generator that converts a general purpose EOEDIS to a customized one with specifications furnished by a domain expert. Each custom system should not only offer a consistent look and feel but also be able to reconstruct business patterns and prioritize them as default settings. However, neither a system administrator nor an end user should be given the privilege to customize the EOEDIS for a particular organization.

Our initial experience in applying the EOEDIS to a medium-sized organization indicates that the navigation interface must be user friendly in order to gain a quick endorsement for the key participants, especially for information acquirers. Several utilities can be integrated into the search interface to improve its user friendliness, one of which is the prioritized business pattern list that reflects the previous patterns used by an information inquirer. Another such utility is to construct the pattern list based on the role of the information inquirer. The interface should take into consideration that the collection of documents for a senior manager does not have the same pattern as that for a system engineer. In essence, while the syntactic entity classes are simple, the custom entity-oriented model should be constructed by a domain expert to best represent the business characteristics (patterns) of a specific organization, which plays a vital role in realizing the potential of the EOEDM.

In addition to traditional search capabilities of the entities, it is important to support browsing via relationships among the entities in a multi-tier cluster as demonstrated in Figure 6. The browsing function that navigates through multi-tier clusters via either *Contains* or *RelatesTo* relationship should follow multiple channels to reach related digital documents. Such a drill-down capability would bring about significant efficacy in pruning irrelevant and collecting factual documents without having to manually adjust the filter to avoid an immense collection of digital documents.



Figure 6. In addition to conventional searches, EOEDIS supports multi-tier clustered retrievals

CONCLUSION

The EOEDM presented in this paper, unlike the existing methods in the current literature that largely overlook business patterns associated with digital documents within an enterprise, takes a formal model approach to enterprise documentation by clustering digital documents in light of intrinsic aspects of business activities described in the documents. While this formal model is adhered to a theoretical framework, it remains flexible enough to impose little restrictions to constructing most business patterns. Consequently, it helps facilitate the retrieval of digital documents for various enterprise applications. The EOEDM locates digital documents by following business relationships among the documents, instead of relying on such traditional methods as statistics or keywords. Because business patterns are relatively stable, fairly predictable, and largely unique to an organization, the EOEDM would assist enterprise users to find documents with higher precision and better recalls. An EOEDIS prototype is developed to demonstrate the feasibility of EOEDM. Although it is not intended to support a general-purpose documentation system due to the limitations of the formal-model approach, the EOEDM introduced in the paper is going to work particularly well in extracting specific collections of digital documents for enterprise applications.

REFERENCES

Bainbridge, D., Thompson, J., & Witten, I. H. (2003). "Assembling and enriching digital library collections," *Proceedings of the third ACM/IEEE-CS joint conference on Digital libraries*, Houston, Texas.

Chen, P. (1976). "The Entity-Relationship Model: Toward a Unified view of Data," *ACM Transactions on Database Systems, 1*(1).

Cooper, J. W., Coden, A. R., & Brown, E. W. (2002). "Detecting similar documents using salient terms," *Proceedings of the eleventh international conference on Information and knowledge management*, McLean, Virginia.

DARPA Program (2004). http://www.nist.gov/speech/tests/tdt/index.htm, accessed in June, 2004.

Florescu, D., Levy, A., & Mendelzon, A. (1998). "Database techniques for the World-Wide Web: a survey," *ACM SIGMOD, 27*(3), pp. 59-74.

Fulker, D., Dawes, S., Kalinichenko, L., Sumner, T., Thanos, & Ushakov, A. (2001), "Digital library collaborations in a world community," *Proceedings of the first ACM/IEEE-CS joint conference on Digital libraries*. Roanoke, Virginia.

Geffner, S., Agrawal, D., El Abbadi, A., & Smith, T. R. (1999). "Browsing large digital library collections using classification hierarchies," *Proceedings of the eighth international conference on Information and knowledge management*, Kansas City, Missouri.

Genesereth, M. R., & Nilsson, N. J. (1987). "Logical Foundations of Artificial Intelligence," *Morgan Kaufmann*, Polo Alto, CA, p. 134.

Grossmann, W., Hudec, M., & Kurzawa, R. (2004). "Web usage mining in e-commerce," International Journal of Electronic Business, 2(5), pp. 480-492.

Guelfi, N., & Pruski, C. (2006). "On the use of ontologies for an optimal representation and exploration of the web," Journal of Digital Information Management, 4(3), pp. 159 – 168.

Isakowitz, T., Stohr, E., & Balasubramanian, P. (1995). "RMM: a methodology for structured hypermedia design," *Communications of the ACM, 36* (8), 34 - 44.

Janée, G., Frew, J., & Valentine, D. (2003), "Content access characterization in digital libraries," *Proceedings of the third ACM/IEEE-CS joint conference on Digital Libraries*, Houston, Texas, 261-262.

McGray, A., & Gallagher, M. (2001), "Principles for digital library development," *Communications of the ACM, 44*(5), 48 – 54.

Medina, R., Smith, L., & Wagner, D. (2003). "Content-based indexing of musical scores," *Proceedings of the third ACM/IEEE-CS joint conference on Digital Libraries*, Houston, Texas, 18-26.

Minsky, M. (1975). "A framework for representing knowledge," in P.H. Winston (Ed.), The Psychology of Computer Vision, McGraw-Hill Publishing, New York, 211-217.

Nelson, M. L., Maly, K., Shen, S., & Zubair, M. (1999). "Buckets: Aggregative, Intelligent, Agents for Publishing in Digital Libraries," *WebNet Journal, Internet Technologies, Applications & Issues, 1*(1), 58-66.

Somani, A., Choy, D., & Kleewein, J. C. (2002). "Bring together content and data management systems: Challenges and opportunities," *IBM Systems Journal, 41*(4), 686-696.

TopicMap (2004). http://www.topicmaps.net/, accessed in July 2004.

Yoo, J., & Bieber, M. (2000). "Finding linking opportunities through relationship-based analysis," *Proceedings of the Eleventh ACM on Hypertext and Hypermedia*, 181-190.