

LiCoRMS – Towards a Resource Management System Based on Lifecycle and Content Information

Axel Schulz^{1,2}, Benedikt Schmidt², and Heiko Paulheim³

¹ Technische Universität Darmstadt, Telecooperation Group,
Hochschulstraße 10, 64289 Darmstadt, Germany

² SAP Research, Bleichstrasse 8, 64283 Darmstadt
{axel.schulz, benedikt.schmidt}@sap.com

³ Technische Universität Darmstadt, Knowledge Engineering Group,
Hochschulstraße 10, 64289 Darmstadt, Germany
paulheim@ke.tu-darmstadt.de

Abstract. Knowledge intensive work is characterized by dealing with an increasing amount of resources, like documents or e-mails. To increase efficiency, users frequently reuse existing resources, e.g., create new documents by using existing ones as a template. This paper introduces *LiCoRMS*¹, a light-weight system for supporting the reuse of resources by capturing and managing relationships between resources using lifecycle *and* content information in resource-centered metadata as well as in an ontology.

Keywords: Metadata, Lifecycle, Resource Management, Ontology

1 Introduction

In modern knowledge-intensive working environments, the knowledge worker performs a lot of tasks to acquire, to distribute and to create new knowledge. Especially linking resources is one of the main objectives of a knowledge worker [1]. Such links can be created implicitly, e.g., by reusing existing resources (which relates the original reused resource to the newly created one), or by explicitly creating links between information sources.

A study of Ravasio et al. [2] shows that, while users have to deal with an increasing amount of resources, they generally have problems to find related resources, since links between resources are often not explicit, e.g., between a document attached to an e-mail and the detached document stored in a local folder.

Fig. 1 shows an example of collaborative work on a document, where authors exchange different versions of the document via e-mail. In the example, user 1 creates a new document, edits it, and sends it to users 2 and 3 via mail. Users 2 and 3 edit the document and resend the edited versions to user 1. User 1 saves the attachments on his desktop. During these external editing processes, user 1 has also changed her local version.

¹ Lifecycle and Content-based Resource Management System.

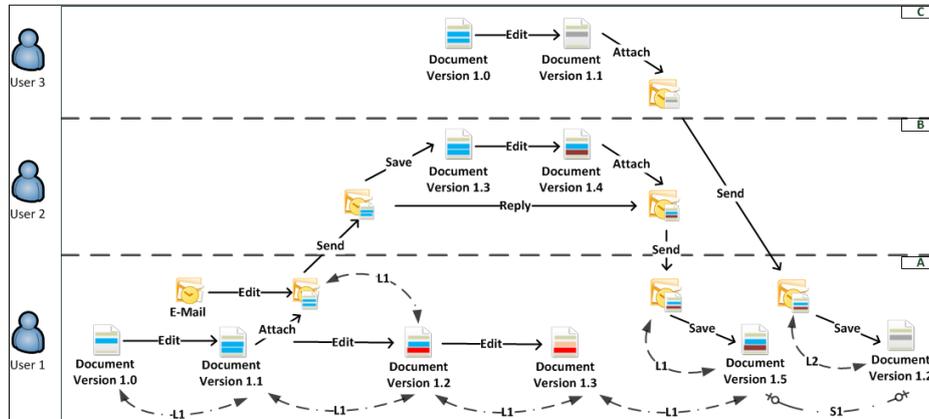


Fig. 1 Example lifecycle for the collaborative creation of documents, involving several related versions of the same document

This collaborative scenario shows that highly complex, distributed activities on resources can lead to an increased number of implicitly related resources on the local desktop. To address these local resource management problems shown in the example, the reuse of resources can be supported by capturing, managing and explicating such implicit relationships between resources.

In order to help user 1 to identify and manage the different versions of the document, it is necessary to capture lifecycle information (L1,L2), which may be distributed across different systems. Relationships between resources may also exist because of the semantic relationship of resources (S1). For example, a document is related to an e-mail it is attached to because of the contents of the document and the e-mail. Furthermore, there may be relationships to additional documents on the user's system.

This paper introduces the *LiCoRMS* system, which captures the user's actions on his local system to describe resources by using *both* lifecycle and content metadata (as opposed to existing systems, which typically only use one of the two) and to provide related resources based on this metadata. The system is robust against loss of information when collaborating with users who do not use LiCoRMS.

2 Management of Lifecycle and Content Metadata

In LiCoRMS, we store lifecycle and content metadata in a resource-centered manner as well as in an ontology-based repository. Thus, we combine the advantages of a decentralized storage, which supports distributed processing of documents, with the power of ontology-based reasoning on resources; their contents, and their usage. For resource-centered metadata, there are formats commonly used, like IEEE LOM²,

² <http://ltsc.ieee.org/wg12/>.

Dublin Core³ and CAM [3]. We decided to use an adaptation of the CAM format, because it allows an efficient representation of metadata and enables the description of lifecycle information.

Besides general information about a document (such as title, type, creation date, etc.), we store lifecycle information as a series of events (including pointers to other resources that were included in the event, e.g., when a document is attached to an e-mail), as well as content information about the document (stored as a set of keywords with their frequency of occurrence).

In the context of a resource management system, the identifiers of documents and events are of special importance. First, references to resources must be uniquely identifiable. Second, for a decentralized management of resources, these values are stored as part of the resources in a way that they are also preserved when processing a document on external systems. Thus, the direct reference between the metadata and the resource remains existent. This enables the bridging of systems without context monitors. Each version of a document has its own ID, which makes it possible to capture and to differentiate lifecycle metadata for every version of a document.

3 Central Concepts of the Resource Lifecycle Ontology

The metadata format is represented using a Resource Lifecycle Ontology (RLO), which is built upon the widely adopted DOLCE top level ontology [4]. Using an ontology allows for providing metadata information in a standardized format, such as RDF.

In the RLO, we have defined a number of central concepts for formalizing content and lifecycle information of documents. A resource, which is present as a file in the file system, is represented by the `CWO:File` concept, using the corresponding concepts from the Computer Work Ontology (CWO) [5].

The lifecycle information through the changing of resources by user activities are represented in the `RLO:InformationTransformationActivity` concept. Every `RLO:InformationTransformationActivity` is related to a `CWO:File`, because the activity changes the file. For representing this relationship, we have defined the `RLO:transforms` association. Thus, a direct relationship between an activity and the corresponding resource can be described. For describing the different activities on resources, several concepts have been introduced. For example, a `RLO:Mail-send-activity` describes the sending of e-mails. Transformation activities may also `DnS:use` (for concepts and relations of the Descriptions & Situations (DnS) Ontology and the Ontology of Plans (OOP) see [4]) other files, e.g., the file attached to an e-mail. The executing user can be defined using the `OoP:Activity` and the related `DnS:Agent`.

A `RLO:File` can be changed by several transformations. Therefore relationships between the transformations have to be described. These are defined using the `DnS:successor` and `DnS:predecessor` relationships between the transformations. This allows for tracking the lifecycle information of a document.

³ <http://dublincore.org/>.

In addition to the lifecycle information described in the transformation activities, the ontology allows for defining content information to allow for finding related resources based on the content information for resources without lifecycle information. Those connections are established using text-based similarity, using the `RLO:DescriptiveStatistic` concept, which is an `OIO:InformationObject` [6] and describes another `OIO:InformationObject`. The `RLO:DescriptiveStatistic` references a set of `OIO:Word`.

4 Conclusion

In this paper, we have introduced the LiCoRMS system for capturing and managing lifecycle and content metadata for resources. While most existing systems concentrate either on lifecycle or on content metadata, our system combines both variants and preserves metadata when processing resources on systems without LiCoRMS. We have implemented a prototype of LiCoRMS, consisting of a context monitor, a centralized ontology-based storage, a visualization module for resource lifecycle information, and slots for several usage modules.

The simultaneous capturing of lifecycle and content information as resource metadata and in an ontology-based repository allows for a number of possible applications, such as proactive user assistance or decentralized collaborative document editing. While the resource-centered storing enables the exchange of resources with external systems, the ontology-based storage enables the fast discovery of related resources through lifecycle information, which would be difficult using only a resource-centered metadata format.

References

1. Davenport, T.H.: Thinking for a living. How to get better performance and results from knowledge workers Harvard Business School Press, Boston, Mass., USA (2005)
2. Ravasio, P., Schär, S.G., Krueger, H.: In pursuit of desktop evolution: User problems and practices with modern desktop systems. *ACM Transactions on Computer-Human Interaction* 11, 156-180 (2004)
3. Wolpers, M., Martin, G., Najjar, J., Duval, E.: Attention metadata in knowledge and learning management. In: Tochtermann, K., Maurer, H. (eds.) *Proceedings of I-Know 2006: 6th International Conference on Knowledge Management*, pp. 1-8. Graz, Austria (2006)
4. Gangemi, A., Borgo, S., Catenacci, C., Lehmann, J.: *Task Taxonomies for Knowledge Content D07* (2004)
5. Schmidt, B., Paulheim, H., Stoitsev, T., Mühlhäuser, M.: Towards a Formalization of Individual Work Execution at Computer Workplaces. To appear in: *Proceedings of the 19th International Conference on Conceptual Structures, Lecture Notes in Artificial Intelligence* (2011)
6. Oberle, D., Lamparter, S., Grimm, S., Vrandečić, D., Staab, S., Gangemi, A.: Towards ontologies for formalizing modularization and communication in large software systems. *Applied Ontology* 1, 163-202 (2006)