Caravela: Semantic Content Management with Automatic Information Integration and Categorization
(System Description)

David Aumueller and Erhard Rahm

University of Leipzig
{david,rahm}@informatik.uni-leipzig.de

Abstract. Semantic web content management poses much manual work onto the community. To reduce this labour we have devised Caravela[1], a generic approach to dynamic content integration and automatic categorization. Content and documents of different types can be integrated from diverse semi-structured sources and categorized along multiple dimensions. Automatic linking provides dynamic categorizations at no user cost. We illustrate our approach by an online bibliography categorizing scientific research publications.

1 Introduction

With the emergence of semantic technologies, methods for semantically annotating information are also used in web content management systems and collaborative environments such as wiki systems. A recent survey of systems for semantic annotation [17] lists several requirements of such systems, including support for user collaboration, ontologies, heterogeneous document formats, document evolution, persistent annotation storage, and automation. Current implementations have problems to fully meet these requirements. In particular, they incur *too much manual work* to add content, categorize content, and deal with the evolution of the content schema and ontologies.

To better meet the requirement of automation we have devised a new approach to semantic content management and created *Caravela*, a generic system for dynamic content integration and automatic categorization. In addition to functions of a web content management system it supports multiple taxonomies to semantically categorize different types of content (e.g. documents, movies, products). It provides functions to automatically integrate, transform, structure, and categorize content. Powerful automatic linking creates dynamic categorization without any user effort. Content can be enriched and periodically refreshed automatically from external web data sources. Furthermore, content type schema and taxonomy evolution are supported to meet changed user requirements. Caravela is fully operational and has e.g. already been used in several instances of a collaborative publication categorizer and a movie navigator.

[1] A *caravela* is a small, highly manoeuvrable ship, used for exploration.

E. Franconi, M. Kifer, and W. May (Eds.): ESWC 2007, LNCS 4519, pp. 729–738, 2007.

The remainder of this paper is structured as follows. We first illustrate the use of the generic platform by briefly introducing the publication categorizer application. We illustrate the underlying architecture of our system in section 3, describing the content repository supporting evolution. In section 4 we present the main contributions towards semi-automatic content management, in particular integration, transformation, and categorization. After presenting related work in section 5 we conclude with an outlook on further work.

2 Sample Application

Caravela is being used for the development and maintenance of web-based bibliographies to collect, structure, and classify scientific publications in a particular domain. While there exist many bibliographic utilities (comprehensive list e.g. on dmoz.org) most of them focus on the generation of references to include in own publications. However, there is little tool support for maintaining open, web-accessible bibliographies to collect relevant publications in dynamic areas, e.g. semantic web technologies. Such bibliographies should ideally categorize publications in semantically rich ways and require little manual work to add and update bibliographic entries. While wiki technology can help to spread the manual work among many users, automatic content integration and enrichment is very important to improve the utility of a web-based bibliography. The Caravela platform can be used to support such requirements.

Fig. 1 shows part of a screenshot of the publication categorizer for papers on schema evolution [13]. Publications are classified along multiple taxonomies, shown on the left. In the example domain we use separate taxonomies for research areas, publication venue, year, citation counts, etc. The mappings between publications and taxonomies may be many-to-many, e.g. for research areas. Categories exhibit an occurrence count indicating the number of corresponding publications.

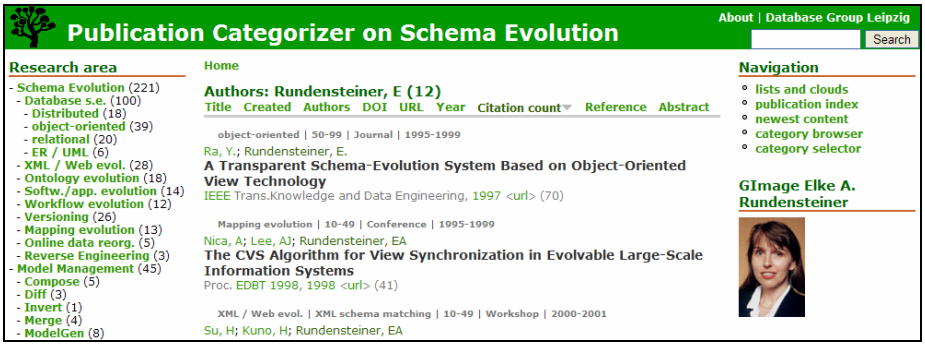

Fig. 1. Publication categorizer, showing publications sorted by citation count

All information to a single publication is presented on one web page, e.g. authors, title, venue, abstract, fulltext or reviews. To reduce manual work we can automatically integrate bibliographic data from external sources, e.g. from data sources like Google

Scholar, publisher web sites or bibliographic reference files. To enrich existing entries relevant data sources can dynamically be queried, e.g. to retrieve current citation counts. We also allow wiki-like manual insertion and editing of instances by interested users. Some taxonomies and some mappings are automatically generated from existing attribute values, e.g. for publication year, publication venue or citation count. Key terms like author names or conference names are automatically extracted and categorized ("automatic linking") and offered for navigational access. Standard features like navigation along the taxonomies, fulltext search, etc. are also supported. Lists of publications can be sorted by their attributes, e.g. their citation count or year.

3 Generic Content Representation

Fig. 2 shows the architecture of our generic approach to semi-automatic semantic web content management, Caravela. It consists of three layers providing data storage (repository), data handling, and data presentation. To limit the implementation effort and to focus on the new aspects for reducing manual work and semantic categorization, Caravela uses some functionality of an existing web content management system (Drupal, drupal.org). In this section we describe the model for generic content representation and its suitability to evolving content. The methods for content integration and categorization are described in section 4.

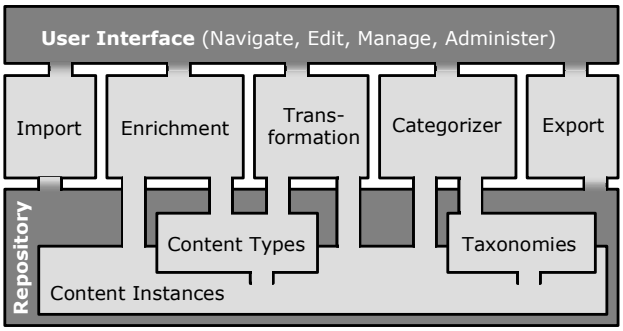

Fig. 2. Architecture of the Caravela platform

We use a relational database to persistently and generically store content and metadata, such as taxonomies. Fig. 3 below illustrates the generic repository structure. Each content item, e.g. publication or movie, is associated to a particular content type and described by several attributes. For example, the publication content type typically has attributes like title, authors, year, and venue. Attributes either store data in simple data types or comprise whole lists of constituents, e.g. a list of authors. We intentionally do not store the constituents separately but extract them dynamically when needed (see 4.3).

As user requirements change in managing content, the underlying content type needs to be adapted. Content types can be altered (or new ones added) by changing,

extending or reducing their attribute lists. Operators for content transformation provide the means to establish the changes on the content instances. Thus, attribute values can be atomized by extracting parts of the values into another attribute, e.g. to extract special bits of data (consider e.g. dates, locations) from verbose text into its own attribute.

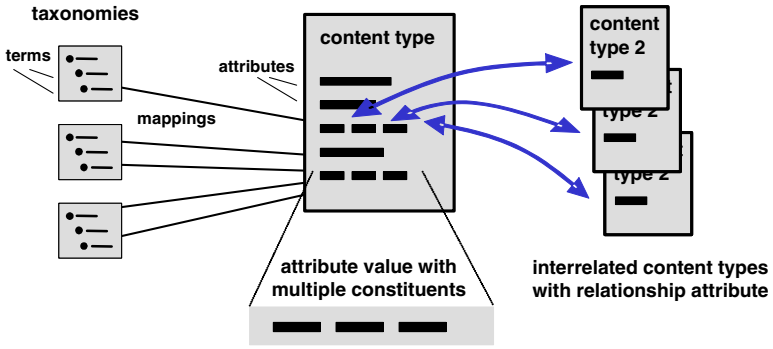

Fig. 3. Content types with various types of attributes and mappings to taxonomies

There may be multiple content types with interrelationships, e.g. a publication content type relating to a detailed conference content type. We use bidirectional relationships between content types so that the content instances are accessible from both directions. For instance, a publication may relate to a conference via a 'published in' relationship, creating a bi-directional navigation path. Complex content evolution is attributed by being able to promote attributes to their own content type. To establish a new content type that is to hold a more detailed description of content formerly residing within a single attribute, we provide data transformations across content types. This includes the creation of associations between the emerging instances of the new content type and the instance of the originating content type. For example, a 'publication venue' attribute can thus be promoted into its own 'conference' content type, or a list of movie actors into their own 'person' content type.

Each content type may have mappings to multiple taxonomies and each content instance may have multiple correspondences to one or more category terms per taxonomy. These mappings are stored in the database with links to the taxonomy terms (Fig. 3).

This simple and generic content representation model eases acquisition and storage of most types of content. Furthermore, content types and taxonomies can easily be added or changed thereby supporting schema evolution. The functionalities of Caravela can be generically extended by providing user-supplied scripts, e.g. to provide further integration (web scraping) and transformation capabilities. The current implementation offers content export to other semantic applications. Furthermore, multiple RSS-feeds provide dynamic content access for other applications.

4 Content Integration and Categorization

Semantic wiki applications (e.g. [2], [4], [11], [18]) can be used to collect any kind of content along with attribute value pairs and interrelating typed links. Such systems though impose the need to master specific syntax and/or lack automatisms to work with the data. With the approaches presented e.g. in [8] and [16], at least the initial population of information can be automated. The full potential of the Caravela platform though lies in automating tasks to save effort and quickly add and transform content. This involves similar data import and transformation tasks as for ETL processing (extract, transform, load) in data warehousing, i.e. information extraction (e.g. [1], [10]), data cleaning (e.g. [15]), and information integration (e.g. [5], [14], [16]), however for documents and web content, where often screen scraping has to take place (e.g. [7], [9]).

Caravela provides functionalities for automating content integration from the Web, content transformation, and automatic categorization. To help maintain high quality content and categorizations all information-editing tasks can also be performed manually. In the next section we outline the methods to integrate and adjust content from files and web sources. We then discuss how content instances are categorized along the taxonomies and how automatic linking is implemented and used, with special regards on the dynamic extraction of attribute constituents.

4.1 Generic Integration of Semi-structured Information

File import. Caravela provides a comprehensive import facility to integrate sets of items available in external data sources. To import data from external files, e.g. XML, RDF, RSS, or CSV files, we adopt a declarative mapping between these document variants to the attributes of a content type. These mappings are specified via a list of XPath expressions, denoting for each target content type attributes its source XML elements. A typical mapping to generate publication instances follows. Here, for each `RDF/item` construct in the source a publication content instance will be created and its attributes filled accordingly:

```
  rdf:RDF/item  -->  publication
./dc:title    -->  publication: title
./dc:creator  -->  publication: authors
```

Web data integration. On a more dynamic level Caravela supports the integration of external data by querying web services, search engines, and applying web/screen scraping. Prior to integration each content instance gathered by such services will be structured internally as list of attribute value pairs. User provided keywords are used to query a service and retrieve values for all attributes of a content type.

An ad-hoc (light-weight) schema matching takes place to map the available attributes of the external data source (web service) to the internal attributes of the according content type. As the schemas of content types usually consist only of few attributes, we use simple name-based matching using string matchers such as edit distance, stemming, n-grams, and phonetic matchers such as soundex. Especially the edit distance measure seems to provide good enough results in our context, as the threshold can be set to match plural and singular forms, e.g. ‘authors’ and ‘author’, or

whether to accept ‘actors’ and ‘authors’ as match or not. To supply mappings that cannot be determined automatically the user can provide synonyms manually that get merged with the auto-generated mapping.

Querying a search engine or a web service with user supplied keywords usually yields multiple results. Thus, before the intended content instances are added into the system the user chooses the instances of interest from the list of returned result set. The user can also adjust and add details much like in a wiki page by editing and adding attribute value pairs directly in the result view (Fig. 4).

Generally, before content instances get actually integrated into the system the user may decide whether to overwrite or skip already existing target instances, or merely append missing attribute values. Generic web/screen scraping is supported by providing scripts that take care of the web data extraction and transform the web data into attribute value pairs, e.g. via regular expressions. Any web service can be attached that offers querying the provided instances by keyword.

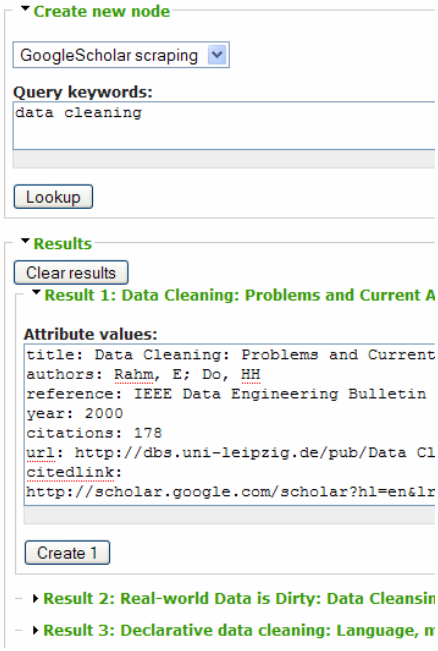

Fig. 4. Editable web data integration

Content enrichment. Instead of creating new content instances by manually supplying keywords, attribute values from existing content instances can be used to automatically query the services to enrich and complete or update the content by incorporating related pieces of information from external data sources. To enrich existing content instances it is necessary to map an existing content instance to an external instance representing the same real world entity. Using multiple available attributes of a content instance as query keywords ensures a better object matching. In Caravela this content enrichment is available as bulk operation for a selection of existing content instances. As the parameters of such a query operation can be stored, it can also be made accessible as single-click action that triggers an update of attribute values of the currently displayed content instance only.

Content mashup. The external content object matching can be used to create a mashup, i.e. including external content live into a view instead of integrating the content into the repository. For instance, we present the results of a Google Scholar author search and author photographs or other images related to the author name in its own blocks on the right hand side of an author page. Other embeddable content of interest include maps and calendars depicting e.g. relevant conference locations and dates.

Data transformations. Integrating information from external data sources may yield inconsistencies such as differently coded values, legacy values, or free form values that may need to be transformed or aligned and mapped into consistent terms. Caravela provides operations to transform attribute values of content instances to achieve such data cleaning. Operations take the specified attribute values of a selection of content instances and replace them by the transformed values and/or fill other attributes with it. Operators for transformation are assembled by regular expressions for search and replace within or across attributes, e.g. replacing instances of unwanted values by defined ones (e.g. '1' and 'F' into female, or 'Very Large Data Bases' into 'VLDB'). As such search and replace rules may sometimes not suffice, conversion tables can be incorporated for look-up. More expressive content manipulation operators can be made available by user-supplied, pluggable scripts that supply functions to derive calculated values, e.g. to normalize author names into a consistent representation.

4.2 Categorization Along Multiple Taxonomies

Content instances in Caravela may be categorized along multiple taxonomies. Offering multiple hierarchies instead of merely one, each taxonomy may be more clearly defined and thus smaller in size, i.e. more easily to understand and maintain. Categorizing content results in a faceted classification available for navigation. This kind of navigation is getting more and more adopted in web applications, e.g. to narrow down product categories or to browse for images and other media (e.g. see [3], [11], [12], [19]). Often, the content in these applications is purely read-only from an end-user perspective or the navigation scheme or categorization is fixed, whereas in Caravela we offer category adjustments that instantly update navigation paths.

The various taxonomies are displayed each in a block on the left hand side. Along each category term the occurrence count indicates the number of correspondences belonging to the term and its descendants, thus adding up document instances assigned to more specific category terms (see Fig. 1). To avoid manual categorization work we provide several automatisms for the categorization along taxonomies. These include the categorization of content instances along given taxonomies, the creation of taxonomies from given content attribute values, and the extension of taxonomies by generating more general terms. We use regular expression and query patterns or incorporate user-supplied scripts to match and create terms.

One approach for automatic categorization is achieved by **deriving taxonomy correspondences** from given attribute values or parts thereof as specified via a regular expression pattern. Consider finding the corresponding decade for a given year. A substring comparison or numerical range containment fulfils this task, as e.g. a substring of the attribute value '1968' matches with the given category term '1960s' or the exact number matches the interval 1960—1969 using a 'between'-query. Existing attribute values can be used to **create a new taxonomy** from scratch, establishing according correspondences to the content instances. This is useful when there are many terms in attributes that are to form a category and/or no other sources available. Consider a taxonomy of all publication venues/conferences mentioned in the system. This approach first yields a flat taxonomy of terms, i.e. simply a 'controlled' vocabulary list, which may be extended by **generating more general**

terms to increase the expressiveness of the taxonomy. To derive the hypernym terms syntactic approaches as aforementioned can be used. From a flat taxonomy of single years e.g., we can derive a first level of more general terms by constructing the decades. Another level on top of that would be presented by the according centuries. This could e.g. be devised by taking the first two digits from the year, incrementing it by 1, and appending '[st|nd|rd|th] century' as suffix to name it appropriately. These functionalities can be extended in Caravela by providing scripts that contain functions to return a more general term to a given term. By querying thesauruses like WordNet such a script may come up with hypernyms not derivable by syntactical patterns. Another strategy for creating a rich taxonomy including more general terms would be to take the number of occurrences belonging to one term into account. Less frequent terms, e.g. years that only carry few correspondences to content instances, could be grouped with others under one more general term. Summarizing, Caravela offers the means to create whole category trees from available attribute values.

As categorization often underlies subtle semantic decisions, e.g. the assignations of publications to research areas, categorization of content cannot be fully automated. To ease manual categorization we devised the drag'n'drop category browser (using AJAX). It offers two modes: In the view displaying all available categories at once, documents can be moved freely around to adjust categorization. A second view presents each taxonomy individually along the list of documents not yet categorized into the current taxonomy. This further entices complete categorization of all content instances.

4.3 Dynamic Categorization by Automatic Linking and Weighting

Apart from taxonomical categorization Caravela offers a powerful dynamic categorization based on attributes. The key idea is that values from certain attributes are automatically and dynamically extracted and cross-linked to all content instances with the corresponding value. These attribute values can be offered for navigation, e.g. at the very spot where they appear in the content instances, or separately as (weighted) lists of grouped/aggregated attribute values (Fig. 5 and 6).

As attributes may contain lists of values, the distinct values (constituents) are available via dynamic extraction as specified in the attribute definition. The default separator for constituents is a semi-colon, but any other pattern may be defined. It can be chosen to define a split pattern as separator or a match pattern to identify the constituents or interesting parts of an attribute value. Any regular expression is allowed; this can be simply a comma or slash for a split pattern or more complex expressions for a match pattern. To display the according occurrence count behind each term (i.e. the number of content instances that contain the same term), the count is gathered using an SQL-query as in `select count(*) from content_type where attribute like '%term%'`. Each term carries

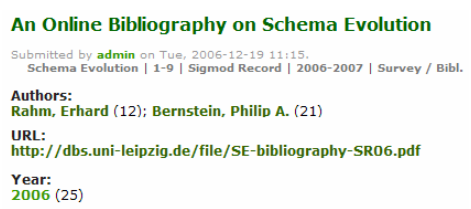

Fig. 5. Occurrence counts in attributes/constituents

an automatically created dynamic link to browse for the content instances containing that term, e.g. for publications by the same author or movies with the same actor. The creation of these dynamic cross links poses no effort to the user who after adding new content immediately benefits from the additional navigation path and the updated occurrence counts.

Aggregated (or grouped) lists of all distinct values or constituents of an attribute within the collection form another categorization scheme to distinguish more prominent attribute values or constituents. Instead of merely presenting the occurrence count or aggregated group count as number the constituents can be visually weighted to produce so called tag clouds. Here the occurrence counts get represented by font size or shades of gray. Thus, more frequent terms get represented larger or in a darker/deeper colour. Instead of representing the frequency (occurrence count) of the terms in the document collection, the weights can also be determined taking other attribute values into account. Regarding the publication categorizer a useful representation consists of author names weighted by their average or maximum citation count of their aggregated publications (Fig. 6). This highlights the more influential authors in the document collection. Such tag clouds are great means to start browsing a collection, as each attribute value or constituent links to appropriate overview pages. Again, there is no user effort in creating them.

Fig. 6. Author cloud as weighted by occurrences (shade) and citations (size)

5 Conclusion and Outlook

With the presented approach towards automatic semantic content management we keep the amount of manual work low. The proposed content repository model is applicable to a large variety of content, easy to maintain and to extend. By being able to integrate information from disparate and unstructured sources Caravela can be used to turn unstructured data into structured data of multiple formats. We provide automatisms for integrating, transforming, and categorizing content of varying type along multiple taxonomies, offering further automatically created dynamic links. By releasing the user from tedious manual work the community can collaboratively lay their strength on maintaining a high quality of the content. Periodically updating content by integrating information from external data sources helps to keep the managed data up to date, e.g. citation counts. Changed user requirements are attributed by schema and taxonomy evolution techniques. Caravela has been successfully applied for different applications, in particular for a powerful publication categorizer, which is well accepted by a growing user base. In further work we plan to provide workflow capabilities to support the repetitive execution of more complex information acquisition and content transformation tasks. The generic approach will be applied to more domains.

References

[1] A. Arasu, H. Garcia-Molina. Extracting structured data from Web pages. In *SIGMOD,* 2003

[2] D. Aumueller. Semantic Authoring and Retrieval within a Wiki. In *ESWC*, 2005

[3] V. Broughton. Faceted classification as a basis for knowledge organization in a digital environment: the bliss bibliographic classification as a model for vocabulary management and the creation of multidimensional knowledge structures. In *New Rev. Hypermedia Multimedia 7(1)*, 2002

[4] M. Buffa, F. Gandon. SweetWiki: semantic web enabled technologies in Wiki. In *Symposium on Wikis*, 2006

[5] W. Cohen. Some practical observations on integration of Web information. In *WebDB*, 1999

[6] A. Doan et al. Community Information Management. In *IEEE Bull. on Data Engineering*.

[7] G. Gottlob et al. The Lixto data extraction project: back and forth between theory and practice. In *PODS*, 2004

[8] A. Di Iorio et al. Automatic Deployment of Semantic Wikis: a Prototype. In *1st Workshop on Semantic Wikis*, 2006

[9] U. Irmak, T. Suel. Interactive wrapper generation with minimal user effort. In *WWW*, 2006

[10] A. Laender et al. A brief survey of Web data extraction tools. In *SIGMOD Record*, 31(2), 2002

[11] E. Oren etal. Annotation and Navigation in Semantic Wikis. In *SemWiki WS at ESWC*, 2006

[12] Schraefel, M.C. et al. The evolving mSpace platform: leveraging the Semantic Web on the Trail of the Memex. In *Hypertext*, 2005

[13] E. Rahm, P.A. Bernstein. An Online Bibliography on Schema Evolution. *ACM SIGMOD Record*, Dec. 2006

[14] E. Rahm et al. iFuice – Information Fusion utilizing Instance Correspondences and Peer Mappings. In *WebDB*, 2005

[15] E. Rahm, H.H. Do. Data Cleaning: Problems and Current Approaches. In *IEEE Data Eng. Bull. 23(4)*, 2000

[16] A. Sheth et al. Managing Semantic Content for the Web. In *IEEE Internet Computing 6(4)*, 2002

[17] V. Uren et al. Semantic annotation for knowledge management: Requirements and a survey of the state of the art. In *Journal of Web Semantics 4(1)*, 2005

[18] M. Völkel et al. Semantic Wikipedia. In *WWW*, 2006

[19] P. Yee et al. Faceted Metadata for Image Search and Browsing. In *ACM CHI*, 2003