Structuring and Presenting Annotated Media Repositories

Lloyd Rutledge, Jacco van Ossenbruggen and Lynda Hardman

CWI (Centrum voor Wiskunde en Informatica)
P.O. Box 94079
NL-1090 GB Amsterdam, The Netherlands
Tel: +31 20 592 40 93, +31 20 592 41 41
E-mail: FirstName.{van.}LastName@cwi.nl

ABSTRACT

We generate hypermedia presentations from annotated media repositories using simple document structure as an intermediate phase. This poster applies Web style technologies to this process. Results include style specification for accessing semantically annotated media repositories, for determining document structure from semantic structure and for applying this document structure to the final presentation.

Categories and Subject Descriptors

H.5.4, H.5.1 [**Information Interfaces and Presentation (e.g., HCI)**]: Hypertext/Hypermedia – *architectures, navigation;* Multimedia Information Systems – *Hypertext navigation and maps*, *Evaluation/methodology;* I.7.2 [**Document and Text Processing**]: Document Preparation – *Hypertext/hypermedia*, *Markup languages*, *Multi/mixed media*, *standards*.

General Terms

Design, Documentation, Languages, Standardization.

Keywords

Document Structure, RDF, Semantics, Style, XHTML+SMIL, XSLT.

1. INTRODUCTION

The Semantic Web envisions a Web that is both human readable and machine processible. In practice, however, there is a large technical, and conceptual, gap between the representation of information at the semantic level and its final presentation. On one side, we find the semantic-oriented technology deployed to build annotated content repositories. This includes RDF and RDF Schema. On the other side is the syntax-oriented technology deployed to build Websites, including XSLT, HTML and SMIL.

Previous work explores transforming semantics to presentation using these technologies and a simple document model we call *structured progression* [2]. We generate structured progressions by finding clusters around resources selected from a set of annotated media items. XSLT transforms this XML-defined general document structure into a Web presentation. More recently, we describe new technical potential for accessing RDF-encoded information from XSLT style sheets, thus applying style to semantics [1]. This poster applies this new technology to our presentation architecture, thus exploring how style applied to semantics affects its relation with general document structure. We present the impact of the new XSLT and RDF technology to three phases, in turn, of our architecture: retrieving document components from the repository, building the general document structure and generating the final presentation.

WWW 2004, May 17-22, 2004, New York, New York, USA.
ACM 1-58113-912-8/04/0005.

2. RETRIEVING CONTENT

Our architecture starts by accepting a user's request for generating a presentation from the media repository [2]. It then retrieves resources from an RDF-encoded repository matching this request to incorporate into the presentation. This returns matching resources from the RDF store that become topics of screen displays in the final presentation. The clustering process then finds additional resources that relate the first resources to each other. These also form topics of screen displays. We call both these types of resources *concept resources* because they represent the conceptual content of the presentation. The transform process accesses these resources with SeRQL queries to find media for presenting the concepts they represent. For these SeRQL queries, each concept resource in the RDF has closely-related *media resources* for conveying it. The media items they locate become the media content of the presentation.

Concept Selection. The user request becomes a query on the RDF returning the leaf node content resources. Our implementation accepts a text string and finds text resources that contain it, but, of course, one could use more complex media search techniques. The concept resources linked to these become the leaf nodes. The clustering treats the RDF encoding like a node-edge graph to derive property tables for applying concept lattice clustering to [2]. However, inferencing encoded by Semantic Web technology can provide additional properties for these tables. Our system, for example, arranges genres of artwork in a subclass hierarchy, allowing subsumption to infer the whole ancestry of genres for each concept node referring directly to any genre. This gives the concept lattice a larger property table to handle, enriching the possibilities for the resulting structured progression.

Inferencing. XSLT transformations can benefit from domain-specific knowledge that does not appear in the XSLT transformation sheet or XML source file. When inference rules derive new information from the repository, the RDF queries embedded in the XSLT sheets automatically benefit from this new information. In addition to providing generation information, an RDF repository can also have inference rules for concluding what media resources fulfill certain template roles, or classes of template roles. In our system, queries from XSLT could refer to such inferred knowledge to determine, for example, the "main" image for a node display about an artist.

Media Selection. The XSLT encoding our transform process has several types of RDF encoding to exploit in its SeRQL queries for media. The simplest is finding directly related resources that are media objects of a given type. More specific is querying for Dublin

Core metadata on the resource. Finally, RDF repositories for specific domains can define their own inference rules for finding media for given roles, thus removing the burden of constructing longer queries from the XSLT.

3. BUILDING STRUCTURE

The returned resources from the user's query are clustered into a structured progression. Structured progressions are built of the following components: *nodes*, *hierarchy*, *sequence* and *recurrence*, which together form essentially an ordered directed acyclic graph (DAG) [2]. We describe the generation of each of these from clustering in turn.

Nodes. A node is a unit of information, all of which is typically displayed simultaneously and viewable in one display. Each node in the structured progression is associated with a concept (or, more precisely, an instance of an RDF class) in the repository. The XSLT processes the XML to get information on the structured progression as a whole and where each node sits within it. The XSLT also queries the RDF directly to get media information about each individual resource. Thus, the XML only encodes the hierarchy. Furthermore, each node in the XML has only a single reference to that node's representation in the original RDF, enabling the XSLT to choose the media conveying the node's topic.

Hierarchy. Perhaps the most important and most used aspect of document structure is hierarchy, which is the nested containment of document content. XML directly offers hierarchical structure, with elements containing other elements as well as direct content. In our architecture, hierarchy of the XML-encoding of the structured progression directly represents the hierarchy of the document. XML is the primary source for XSLT to derive structured progression. As with generating the main displays, the XSLT acquires from the XML a reference to the RDF for each node, which it uses to acquire media items for presentation. It requests Dublin Core title tags to put in the outline display and in overviews. The XSLT also uses the same queries for main images to get images for the summary thumbnail displays.

Sequence. RDF does not specify sequence over its resources. RDF explicitly assumes no significance to the order in which it encode items and, in the absence of the controversial exception of the `<rdfs:seq>` construct, has no concept of significant order. However, it must provide the information necessary for other processes to determine sequence over the resources RDF returns. Our demo uses the date of creation as the RDF-encoded basis for sorting.

Recurrence. A recurrence is the appearance of the same node at multiple locations in the tree. Recurrences find their place in traditional discourse as recurring themes in stories or as topics that different parts of a textbook discuss. The clustering technique produces recurrences along with the hierarchy. If an artifact has separate characteristics in common with separate groups of other artifacts, then this artifact falls in separate groupings and thus in separate places in the hierarchy. The XSLT code detects recurrences as multiple nodes with identical RDF resource references.

4. FINAL PRESENTATION

Given a structured progression for the query returns, style processing generates many possible presentations for it. The variety of styles enable users and designers to specify different styles for conveying the same underlying discourse. Here we present style processing issues for each of the components of structured progression.

Nodes. The selected semantic resources passed through as nodes in the structured progression are the presentation's conceptual content. Each resource node appears as the main current topic at some point in the presentation. Our system makes presentations on both the concepts the query returns and the *composite node* concepts that relate them to one another

Hierarchy. In addition to the nodes selected for presentation, there is the presentation structure built around them. Our system accesses the hierarchal encoding to provide a hierarchical-based navigation into the presentation. The primary component of the presentation providing this is the outline display.

Sequence. Sequence means that sibling nodes with a composite node in the hierarchy have a clear and significant order. Textbooks, and text in general, have a meaningful sequence in which their contents appear. This poster applies the structured progression tree to the generation of a single linear thread. The basis of this procedure is using depth-first traversal to flatten a tree into a sequence of nodes. Some Web presentation format constructs, such as those providing text flow space and time, are inherently sequential, thus conveying sequence readily. Though not as directly sequential as space and time, interaction choices form a chain, making them also sequential. Visual transitions apply to sequence not by conveying them but by smoothing the impact of the traversal between them on the user.

Recurrence. Because our structured progression is a directed acyclic graph instead of a tree, artifacts and entire sections can occur repeatedly throughout a generated presentation. Proper communication of recurrences conveys an additional dimension beyond hierarchy along which document components unite. That is, it is another way to convey relationships between concepts, thus giving the user access to more relationships. Communicating how each node recurs through the presentation helps users understand better how all the document weaves its topics together into a coherent whole. However, conveying recurrence is challenging since it crosses the grain of the presentation hierarchy.

5. SUMMARY AND CONCLUSION

Having XSLT send queries on RDF and process the results simplifies and strengthens architectures for clustering semantic net subsets into document structure for transformation to final presentation. As a result, the XML encoding is simpler and enables increasing flexible and powerful style processing of it. XSLT-defined style thus applies to RDF-defined information repositories as well as to XML derived from it. The demo and other resources for this poster are accessible at `http://homepages.cwi.nl/~media/conferences/WWW2004/`. An extended description of this poster's topic is available [3].

ACKNOWLEDGMENTS

This work was funded by the Topia project of the Telematica Instituut, sponsored in part by IBM, and by the NWO NASH project. Ivan Herman provided valuable insights into the comparison of XML, RDF and XSLT.

REFERENCES

[1] van Ossenbruggen, J., Hardman, L., Rutledge, L., Towards Smart Style: Combining RDF Semantics with XML Document Transformations, CWI Technical Report INS-E0303, October 2003.

[2] Rutledge, L. Alberink, M., Brussee, R., Pokraev, S., van Dieten W., and Veenstra, M. Finding the Story — Broader Applicability of Semantics and Discourse for Hypermedia Generation. In Proceedings of the 14th ACM conference on Hypertext and Hypermedia (HT03) (Nottingham, UK, August 26-30, 2003), 67-76.

[3] Rutledge, L., van Ossenbruggen, J., Hardman, L. Structuring and Presenting Annotated Media Repositories, CWI Technical Report INS-E0402, February 2004