Finding the Story – Broader Applicability of Semantics and Discourse for Hypermedia Generation

Lloyd Rutledge*, Martin Alberink, Rogier Brussee, Stanislav Pokraev, William van Dieten and Mettina Veenstra

*CWI (Centrum voor Wiskunde en Informatica)
P.O. Box 94079
NL-1090 GB Amsterdam, The Netherlands
+31 20 592 41 27
Lloyd.Rutledge@cwi.nl

Telematica Instituut
Postbus 589
NL-7500 AN Enschede, The Netherlands
+31 53 485 04 85
{firstname.lastname}@telin.nl

ABSTRACT

Generating hypermedia presentations requires processing constituent material into coherent, unified presentations. One large challenge is creating a generic process for producing hypermedia presentations from the semantics of potentially unfamiliar domains. The resulting presentations must both respect the underlying semantics and appear as coherent, plausible and, if possible, pleasant to the user. Among the related unsolved problems is the inclusion of discourse knowledge in the generation process. One potential approach is generating a discourse structure derived from generic processing of the underlying domain semantics, transforming this to a structured progression and then using this to steer the choice of hypermedia communicative devices used to convey the actual information in the resulting presentation.

This paper presents the results of the first phase of the Topia project, which explored this approach. These results include an architecture for this more domain-independent processing of semantics and discourse into hypermedia presentations. We demonstrate this architecture with an implementation using Web standards and freely available technologies.

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

Documentation, Design, Experimentation, Standardization.

Keywords

Discourse, Narrative, Hypermedia, Semantics, Clustering, Concept Lattices, RDF, SMIL.

1. INTRODUCTION

While the Web allows users to find existing documents that closely match their needs, it does not yet enable generation of rich and engaging presentations on topics of their choosing. Rather than hoping a good document on a desired topic is already on the Web, and searching through long lists of query returns to find it, a more efficient use of a user's time would be to construct a relevant, coherent and engaging hypermedia presentation on request. While this goal remains elusive, technologies are emerging that can form different steps in a process chain that could generate such hypermedia presentations on demand.

The currently emerging Semantic Web technologies [11] are changing the documenter's task from that of making final presentations to that of placing on the Web nuggets of raw knowledge, annotations of media items and hyperlinks among them that can later apply to generating presentations on the fly. With the large network of media components and semantic relations between them that result, ever-improving link clustering techniques help find the most relevant relations among a given segment of such a net [22]. Recent advances in discourse generation have led to meaningful stories derived from semantic relations [2][14][19]. Finally, with tighter coupling of document processing technologies and better performance of interactive multimedia browsers, engaging hypermedia presentations can convey these stories on-request to their users.

However, both the processing of semantics and the generation of discourse have a strong reliance on human intelligence. Fully automating these two phases along with the rest of the process chain from digital archive to interactive multimedia is a formidable task. Developers typically achieve this by tightly focusing the domains involved: semantic networks usually cover specific topic areas, while computable discourse consists most often of fixed templates for specific narrative genres [14][19]. In this manner, human intelligence guides the automatic processing by restricting the combinatorial possibilities to those the human author can be sure would make sense. This ensures the richness of human involvement in the result, but only by sacrificing automation's flexibility and wide applicability. This domain-specificity also inhibits applying semantics and discourse to how users currently typically experience the Web, which they can query in and have presented as a unified global information repository.

In the Topia (Topic-based Interaction with Archives) project, we explore incorporating semantic and discourse into Web technology and the Web experience, providing a complete process chain from request to presentation that applies to diverse knowledge resources. Our approach is to make the opposite trade-off of much current research by focusing on those aspects of semantics and discourse that are relatively domain-independent and computable. While these aspects may typically seem overly trivial in the humanist fields of semantics and discourse, we propose that processing them can achieve enough to be useful in the absence of human-crafted presentations on the user's topic. We feel this approach can provide general Web search returns with not just appropriate media objects communicating the desired knowledge, but also a presentation structure around them that itself conveys additional knowledge on the desired topic. Since one of the primary benefits of the Web is how it unites different document communities into one global system, we feel this should hold, as much as possible, for different semantic communities as well as the Web extends into semantic processing. This paper’s approach helps the Web move toward this goal.

This approach will never process semantics into discourse-based presentations as well as human-crafted domain-specific ones do. However, it can enable relatively quick access to information, and to better understanding of complex relations underlying a topic, when no alternative is available. Similarly, the resulting presentations may serve as first drafts that authors can request and them improve with their human understanding of richer semantics and discourse.

In the next section, we present related work that processes semantics and discourse in automatically generating hypermedia presentations. We then discuss the four phases in our process architecture, illustrated in Figure 1, which are, in turn: creating the semantic network, determining clusters from semantic relations, deriving discourse from these clusters and, finally, generating the hypermedia presentations based on the discourse structure. The paper enforces this discussion with the Topia project’s implementation, which provides a query interface to a semantic network and generates a presentation from it. The sample semantic network we used annotates the collection of the Rijksmuseum Amsterdam [25]. At the end, we wrap up this paper with a discussion of the implementation itself, followed by a summary and conclusion.

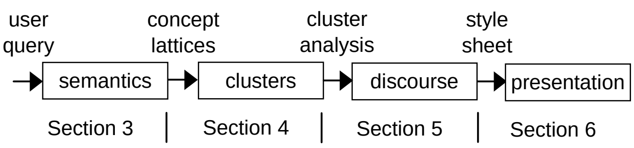

Figure 1. Simple version of our four step architecture

2. RELATED WORK

Since our architecture has phases from four different fields, much ongoing research applies. A lot of it involves more than one of these phases and has similar overall goals to this work. We discuss this related work in roughly the order of the four steps of our architecture, shown in Figure 1.

2.1 Semantic Processing

Semantic markup promises to further improve the quality of search engine results with semantic search systems such as SHOE [15]. These search systems still focus on the quality of the selection for returned items to place in an unstructured list. In this work, we focus on using the semantic relations between the items returned to build a discourse structure around these items in the final presentation. Current research also uses clustering to improve the quality of search results [22]. As with semantics, our application of clustering techniques is for improving the presentation structure around the items returned rather than to help select these items.

The ScholOnto project defines semantic encoding of scholarly argumentation [7]. This work includes modifying and presenting the resulting semantic network. The domain of the semantic data store and the mechanism for presenting it are both strongly focused on the domain of scholarly argumentation. The ScholOnto project also emphasizes the social context of its use, enabling a community of users to readily act as both archivist and audience. It provides easy addition in its semantic network and allows the audience to easily browse and perceive the structure it defines.

2.2 Transforming Semantics to Presentation

The Artequakt project is implementing the automatic generation of text biographies for persons the users request [2]. Topia and Artequakt both process Semantic Web formats into an intermediate discourse structure and then into a document. The three key differences between the systems are the breadth of topic domains they apply to, the semantic richness of the intermediate discourse and the medium of the final presentation.

We describe one key difference between the Artequakt and Topia approaches as top-down and bottom-up. Artequakt uses narrative templates. That is, it has created narrative composites with variables the system can fill in. This is bottom-up because the higher components of the compositional structure are set, and then the system fills in the lower components.

Topia, on the other hand, works from the top down, generating all levels of discourse structure composition. This results in Topia’s approach generating a wider variety of discourse structures because it does not rely on genre-specific templates fixing the broader structure. However, these discourses would probably be less rich than those Artequakt’s human-crafted narrative templates provide. Another distinction is that Artequakt’s presentation medium is a text, whereas Topia generates hypermedia presentations.

Our CWI colleagues have recently combined these approaches by exploring bottom-up discourse generation in a Topia-like environment [14]. They describe converting the same Rijksmuseum semantic data set Topia uses into the structured progression discourse this paper presents, and then into multimedia presentations. Their bottom-up approach shows how rules can be written for a specific discourse genre, such as biography, that convert semantic markup for a particular domain, such as museums, into multimedia presentations. Unlike Topia, they have no clustering phase, since their system generates discourse components directly from the RDF structure using domain- and genre-specific rules encoded and processed with Semantic Web technologies.

Little and her colleagues have proposed a technique for processing semantic annotations in Dublin Core into multimedia presentations [19]. This approaches domain independence of semantics because Dublin Core standardized a small, basic set of properties consider widely applicable across semantic domains. However, its transformation of semantics to multimedia does not account for discourse structure but instead has a direct conversion from particular patterns of semantics to spatial-temporal structure.

2.3 Presentation Generation

The WIP and PPP systems generate hypermedia technical instructions from user queries [3]. The presentation genre is specific to that of how-to manuals. WIP and PPP generate media components tailored for incorporation in particular presentations. They also track the goals of the presentation and how the presentation meets them as it progresses. As with the Topia .project, rather than using templates, WIP and PPP use a simple discourse model, which derives from goal planning, that enables flexibility in how the discourse is constructed. The goal-process also allows dynamism in how the discourse plan adapts to different user navigations. This goal-based discourse generation is less of a trade-off than that of Topia: WIP and PPP are quite flexible in how they achieve the goals, but the approach applies best for instructional presentations in which goal achievement is a key aspect of the discourse.

Kamps has developed a system that generates layout from relations in the artistic domain [17]. Its focus is the intricate special placement and graphics that communicate complex relational structure. In other work, he and his colleagues look at layout generation [4]. Topia's communicative devices are a similar approach, but they focus less on layout and graphics to include the all aspects of hypermedia presentation structure.

3. STORING THE FACTS – SEMANTICS FOR DISCOURSE

The term “topiary hypertext”[1] derives from “calligraphic” and “sculptural” hypertext [28]. Whereas calligraphic hypertext is the creation of new links and sculptural hypertext is the selective removal of existing links from an overly large set, topiary hypertext is, as we interpret it, the planting of linkable objects from which presented links to the existing objects automatically grow. Archivists can add new objects to Topia's topiary hypermedia with semantic links, and resulting links will automatically sprout in later presentations, along with other corresponding changes in presentation content and structure.

3.1 Domain-independence

A key requirement of semantic processing in our approach is that the later generation of clusters, discourse and presentation can readily accept the semantic markup of any domain as input. That is, discourse must be readily derivable from semantics of domains in general supported in a given environment. Our approach for enabling this is two-fold. First, we use the foundational constructs common to all semantic domains. Second, we provide users and documenters with very quick means of adding, potentially during the final presentation request, heuristics for processing specific encountered semantic domains into hypermedia.

The first part of this approach causes a strong reliance on the basic commonalities shared by semantic encoding of different domains. We assume that each semantic network consists of labeled nodes and directed, labeled hyperlinks between them. These components are, by themselves, empty of the inferencing that distinguishes semantics. However, this paper shows that they do provide helpful input for generating discourse. Furthermore, we propose that semantic inferencing can apply when available for an encountered domain. In such cases, the domain-specific detailed results could then fill in lower-level components of the broader discourse generated from the processing of multiple semantic domains.

3.2 Findable vs. Relatable

In the system we present here, the user requests information in a now familiar manner: he or she enters a text string, and a text-based search returns a collection of objects. Our focus here is not on the search itself but on processing the semantically defined relations between the returned items after retrieval. Rather than just line these items up in a list, this semantic relational structure can derive a more informative presentation structure around these items – ideally, a “story” that both unites them and lines them up in a coherent narrative.

Our system distinguishes nodes that the search returns, the *items*, from nodes describing shared properties of these items, the *relations*. The primary semantics format RDF [18], which we use to encode Topia’s semantic metadata store, make a clear distinction between objects and relations. Relations in RDF are first-class objects that can apply to multiple domains, appear in hierarchies and enable querying. This means that rather than relying on a standard set of relations in the semantic network, an implementation can discover which relations occur and permit the user quick specification of how they are processed. Thus, our approach can use the underlying technology common to many semantic networks to distinguish nodes from relations, thus contributing to universality across these networks. For the purposes of later clustering, relations derived from RDF foundational, and thus domain-independent, constructs can be shared property values and references to common RDF constructs.

3.3 Media vs. Concepts

The final presentation consists, of course, entire of displayable media. This media comes from the data store. The data store also has semantically encoded concepts that relate to the media and to other concepts. Processing these concepts helps put discourse structure around the media items in the presentation, but, of course, only the media items appear directly in it.

The presentation needs to display the media items that communicate the selected concepts. The determination of media conveying a selected item is relatively easy. Usually, one can locate a large media item, such as multiple sentences of text, or an object from the bulkier media such as an image, that is part of the item or immediately linked to it. Determining media for the relations that cluster nodes is more complex. Here, large media close to relations still serve as good candidates.

A more difficult media requirement that arises during presentation generation is the need for determining short captions or thumbnails. One resolution is to take the media for the full descriptions and shorten it. Thumbnails of images can simply be a scaled down version of the original. Generating meaningful captions from larger text is more complex, but technologies exist for doing so reasonably well [20].

A more practical solution for these less universal requirements is the adoption across multiple semantic networks of a small set of simple properties that apply to any domain yet specify what

[1] This term emerged during session discussions at Hypertext 2002.

presentation generation needs. For example, properties such as "description" and "caption" are relatively universal and would make the generation of presentation across multiple semantic networks more dependable. Dublin Core [12] provides such constructs, including the concept "title", which we use in our implementation.

3.4 Grouping

As we will see in the upcoming sections on clustering and discourse, the ability to determine groups that the items returned fall into is important in generating discourse. The clustering phase needs to be able to recognize common properties in its group building. RDF, and our use of it here, provides this ability with RDF properties. Our implementation also uses RDF relations by recognizing relations of the same type to the same object as a common property between multiple objects that share it.

4. FINDING THE RELATIONS – CLUSTERING SEMANTICS

When given a collection of items and the relations that join them in a graph, the next step is to look for patterns in the graph that act as landmarks for important locations that guide our traversal through it. A frequently used type of such landmarks is the *cluster*, which is a node with close proximity to a relatively large number of the originally selected nodes in the graph. A cluster node is not always among the originally selected nodes. In such cases, normal navigation may serve to guide traversal to the nodes selected. Though based on domain-specific properties, presentation of items as groups gives items a position in the final presentation that users understand and which can add to their insight in the entire set of retrieved items.

4.1 Concept Lattices

A wide variety of clustering techniques for hypertext exists. For our purposes, it does not matter which one to use, as long as it finds a good group of clusters to pass on to discourse processing. For our implementation, we chose concept lattices because of their relative simplicity and their universal applicability.

A *concept lattice* [13] is the set of pairs of items and the set of properties these item share. The term for such a pair is *concept*. Shared properties determine *clusters*, which we view as particularly important concepts. A single item can appear in more than one cluster. Clusters have directed hyperlinks to other clusters that contain either a superset or a subset of its objects. If more than one cluster has a superset or a subset of the objects in a cluster, then it only connects to the clusters with, respectively, its smallest superset or largest subset. The top concept in the concept lattice consists of the properties all objects share. Many nodes can appear in the concept lattice, depending on the number of clusters formed based on commonalities in the metadata. Figure 2 shows an example of a concept lattice generated by a query for "water" performed on Topia's semantic network annotating the Rijksmuseum Amsterdam collection. Figure 2 shows the cluster graph of this concept lattice.

4.2 Basis of Clustering

The cluster component takes all pairs of property types and corresponding property values which occur in the retrieved items as concepts. This allows all subsets of items in the retrieval result with a common property type and property value to appear as clusters in the resulting concept lattice. In the case of the Rijksmuseum semantic net in our demo, the nodes selected by the original query over are all artifacts. The cluster nodes that unite artifacts into groups are usually descriptive, such as "genre" or "land of creation". The common relations to the descriptive nodes that end up as clusters form the basis for an informative structure around the items. Note that the top node in the lattice consists of the result of the query together with the properties that specify it.

The common property values associated with each cluster characterize the group of items in the corresponding cluster. Conveying items which appear in clusters as groups in the eventual presentation not only helps users viewing items, but it also helps users associate items based on the common property value. This results in increased insight in relations among the items. To enable users to be aware of such associations, some of the clusters form building blocks for the resulting presentation. The next section explains the selection of items in clusters.

Table 1. Artifacts mapped against properties in a concept lattice for query on "water"

(C1) "Water"	(C2) Genre: Water ice and snow	(C3) Genre: Buildings in landscapes	(C4) Genre: Field meadows	(C5) Genre: Dutch Landscapes	(C6) Artist: Jacob van Ruisdael	(C7) Genre: Tree forests	(C8) Genre: Riverscapes	(C9) Artist: Paul Joseph Constantin Gabriel
"A watercourse at Abcoude" (A1)	X		X	X				X
"Watercourse near 's-Graveland" (A2)	X			X				
"Mountainous landscape with waterfall" (A3)	X				X	X	X	
"A water mill" (A4)	X					X	X	
"Landscape with waterfall" (A5)	X				X	X	X	
"Water mill" (A6)		X			X			
"Windmill on a polder waterway, known as 'In the month of July'" (A7)		X	X	X				X
"A waterside ruin in Italy" (A8)		X						
"The battle of Waterloo, 18 june 1815" (A9)			X					
Concept Size	**5**	**3**	**3**	**3**	**3**	**3**	**3**	**2**

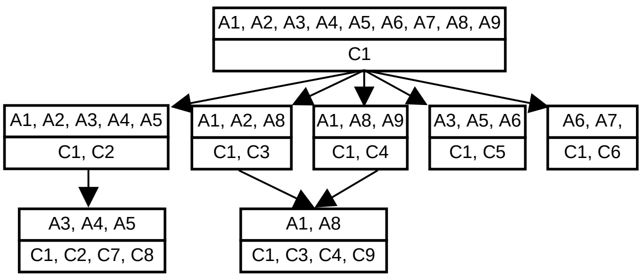

Figure 2. Cluster graph of concept lattice from Table 1

Clustering in Topia can take place directly on top of the RDF database contents. Topia's metadata store encodes properties of items as RDF triples. Subject, predicate and object of these RDF triples are item identifier, property type and property value respectively. The subjects are the items in the concept lattice, and the concepts are the combination of predicate and object.

The current version of the Topia implementation clusters only items with an exact match of the property value. A number of property values subsumed in RDF classes would allow generation of clusters with values in a certain class, thus extending the types of clusters which can appear to items with a property value in a certain class. This can help achieve semantic interoperability when databases are integrated, while the same clustering principle can be maintained, this time based on the appearance of property types and property values in classes. Basing clustering on classes can also help achieve broader and potentially more significant clusters than by exact match of property types and property values. In addition, our selection of main metadata types is not necessary: in a purely universal approach, all relations are equal in the concept lattice. It is possible to extend the selection criteria with other properties, or aggregated or "inverse" properties.

4.3 Significance of Clusters

Topia's clustering component generates a cluster structure of a set of retrieved items. It is a preprocessing step for the discourse generation, which we explain in the next section. In response to a query, the set of items in the repository matching the query comes back, along with property assignments each item has. The set of items and these property values are the input for the clustering algorithm.

Items with a specific commonality appear together as nodes in the discourse hierarchy of the final presentation. Including all clusters in the structured progression can result in presentations with an overwhelming number of nodes. It can also cause repetition of many items during presentation. To reduce this complexity in the presentation while still showing all items, the Topia implementation selects only clusters at a user-selected depth below the top node. The selection is in decreasing order of the number of items contained. This process stops when all items in the retrieval result are in at least one selected cluster. The selection process results in a hierarchy containing all the originally selected items.

The upcoming phase, discourse generation, relies heavily on the clustering that this phase generates. The primary requirement the discourse phase has for the clusters it gets is that they have some measured rating of importance, since this rating determines what type of discourse construct each cluster becomes and what order components are presented in. Our demonstrator calculates this rating based on the size of a cluster combined with how important the user indicates, using a form interface, the cluster's defining properties.

5. BUILDING THE STORY – DERIVING DISCOURSE

In presented hypertext, users "find the story" by forging their own paths through a link network [27]. Users form their own narratives from the many possibilities the underlying hypertext provides. We aim to form stories, or discourse structures, around search returns automatically to make presentations more informative to the user, and perhaps make it easier for users to further develop these stories through interaction. Such discourse should be derivable from any semantic source to contribute to the vision of a fully connected Web. Ideally, one should be able to get the story they want from the Web as a whole, and from any knowledge and media store within it.

However, because of the human insight needed, the idea of a "story" in automatically generated presentations is more of an analogy or guiding principle than a fully achievable goal. Nonetheless, we feel there is a subset of narrative and discourse concepts that one can automatically derive from semantics. A Topia project's goal is determining this subset, how to best derive it from semantics in general and how to most efficiently present it. In this section, we present this computable subset of narrative and discourse, which we call *structured progressions*.

The generation of concept lattices results in clustering the retrieved items so that the items in each of the clusters have at least one property type with a corresponding property value in common. Communication of these groups of items as a whole together with this property type and property value gives users better guidance when inspecting the retrieval result compared to a bare list of items. This is, of course, because presenting the set of items in a cluster as a group conveys that this group shares an important characteristic, namely the common concept of the group. In contrast to an unsorted list, there is a clear reason for presenting each of the items as part of the group it belongs to in the presentation. Note that items can appear in more than one cluster, and the presentation may show them more than once. However, not only is there a gain because of the fact that users learn about the characteristics of individual items by inspecting the groups to which they belong. In addition, users may be able to spot other commonalities in groups of items presented as clusters, for which the metadata was not available for the computer to find that commonality. However, users may be able to find them based on their knowledge and their ability to inspect images and see their features.

The discourse component of our architecture uses the sets of items in the clusters of the generated concept lattice as building blocks for the generation of a framework for *structured progression* by users through the retrieved set of items. We stick to basic core constructs of discourse. This augments its applicability to a wider domain of semantic networks, since constructs of discourse that are less semantically rich do not require rich domain-specific semantic processing to generate them. Structured progression constructs include the order of presentation, grouping of components into sections and subsections, and recurring themes distributed throughout the presentation.

The assignment of clusters to particular components of structured progression depends on the weight each cluster has. This mapping considers clusters with more *weight* as considered more significant to the discourse, and thus worthy of higher-level representation in the discourse. The simplest, and most semantic domain-independent, weight assignment is simply the cluster size. However, there are quick means of giving some clusters more significance that can have a large impact on how informative the resulting discourse structure is. The primary technique for this is to, when encountering a semantic store of a given domain, give higher weight or, in the extreme case, exclusive consideration to certain properties in that domain. Our demonstration combines size with user assessment of the significance of each property to determine a cluster's weight.

The components of structured progression we chose for this work are *hierarchical structure*, meaningful *order*, *recurring themes* and *tangents*. We chose these because they are domain-independent and readily computable. They make up a small subset of narrative, which in turn is a subset of discourse. We describe each of these components next.

5.1 Hierarchical Structure

Documents typically have a hierarchical structure that helps the user keep track of where he or she is while moving forward through it. Texts have sections and sub-sections. Performances have acts and scenes. Because of their importance and, as we discuss next, their derivability from concept lattices, we include hierarchical structured in structured progression.

As Figure 2 illustrates, concept lattices have associated nested containment, or hierarchies. The Topia project's architecture maps this directly to the hierarchy of the structured progression. We consider hierarchical grouping to be the most significant structure progression construct. Therefore, only clusters that are significant enough, as measured by their cluster weight, become part of the discourse hierarchy. These lighter clusters transform instead to other components of structured progressions.

5.2 Meaningful Order

The order in which individual items and groups of items appear conveys something to the user about how the items relate to one another. We feel many aspects of sequencing from traditional narrative and discourse theory require human intelligence to encode or process and are thus not candidates for structured progression. These include building expectations, suspense or steps in a rhetorical argument.

However, numeric properties in semantic properties provide a domain-independent means of sorting items sharing one such property. Systems can detect a sorting property on a group of items and then assert that they appear in the order in the generated structured progression, whose processing then ensure the presentation keeps them in that order. Similarly, the system can sort groups based on the minimum, maximum or average of their items' values for a sorting property. Our Rijksmuseum demo, for example, uses the date of creation property as the basis for sorting artifacts in a group. In the resulting presentation, the user can more easily perceive patterns in how artifacts change over the years.

5.3 Recurring Themes

When a cluster is not important enough to map to the discourse hierarchy, it can instead sit in the structured progression as a recurring theme. Recurring themes are properties shared by multiple items distributed through the discourse hierarchy. When rendered to a presentation, the user learns when seeing each item that it is part of a particular recurring theme. While a hierarchical group sits above its items in a presentation and binds them together as part of its display, recurring themes present themselves as component of each item's display and allow the items to appear apart from each other.

Since an item can appear in more than one cluster, it can occur at more than one position in the structured progression, and thus be a recurring theme. Occurrence of an item in several clusters reveals that all properties of the clusters it appears in apply to it. Awareness of a combination of properties to items allows users to infer other properties of such items. Furthermore, frequent occurrence in items of a certain combination of properties lets users learn about the whole group of items sharing them. Users of the generated presentations should know when they arrive at an item they have seen before or will see again. This allows them to derive meaning from the occurrence of possibly noteworthy combinations of properties.

5.4 Tangents

While clusters with high significance qualify as hierarchical groups and clusters with middle significance qualify as recurring themes, the remaining least significant clusters are not important enough to appear in the primary flow of the resulting presentation. However, the user may choose to wander off the presentation's main path and apply previously undetected significance to these lightest clusters. Providing access to them from points in the presentation that relate most strongly to them allows the user to best understand how these serendipitous discoveries relate to the overall presentation and the knowledge it conveys. To give users this extra freedom, we provide the *tangent* as a component of structured progressions. In the resulting architecture, all concepts map to one of three components of structured progression based on their significance. Since even solitary items can form their own single-member clusters, every property of every item returned has a place as a concept, or topic, in the structured progression and in presentations generated from it

6. TELLING THE STORY – HYPERMEDIA PRESENTATION

A structured progression is not directly presentable because it represents the abstractions of how a presentation would progress without the details of exactly how it should appear. To present the generated document to the user, the final step in the Topia architecture is converting the structured progression into a presentation. Since there can be many different presentations of a given structured progression, these conversion programs act as "discourse style sheets", allowing designers and users to specify preferred means of having structured progression presented.

A presentation not only shows the media it contains, it also reflects the discourse structure joining the media together. Topia's structured progression subset of discourse has simple constructs that presentations can efficiently convey in bulk. We call this larger-scale communication of the underlying story *discourse perceptualization*. Just as data visualization conveys large and complex scientific structures, our discourse perceptualization allows the user to perceive, at every point in the presentation, the overall structured progression and the context of the current point within it. Furthermore, while data visualization uses graphic and visual techniques, discourse perceptualization uses all aspects of hypermedia presentation behavior to communicate to communicate large-scale discourse structure.

The aspects of hypermedia presentation we use for discourse perceptualization fall into four categories, which we take from earlier work [26]: *media*, *layout*, *timing* and *interaction*. Much understanding already exists for creating individual media to communicate discourse, not the least of which comes from thousands of years of literary theory for discourse in text, written and spoken. The Artequakt project handles the automatic generation of text for conveying narrative [2]. Bateman and Kamps explore the construction of text and graphic media and the use of spatial layout for communicating underlying document structure [4][17]. In earlier work, we explored the use of all four hypermedia categories for conveying rhetorical structure in particular [26]. Here, we apply similar techniques to structured progressions instead of rhetoric. By accounting for the generation of media components as well as the structuring of all aspects of hypermedia integration of these components, we feel one maximizes the potential for conveying, or allowing the user to perceive, discourse in a presentation.

In this section, we discuss discourse perceptualization in terms of *hypermedia communicative devices*, introduced in earlier work [26]. Each device is a mapping between a pattern of discourse (or other higher-level conceptual) structure and a pattern of hypermedia presentation structure that conveys it. Communicative devices act as guidelines helping programmers of style sheets define transforms from the first structure to the second. Here, we take some familiar patterns of hypermedia presentation and present them as examples of hypermedia communicative devices for structured progressions. Since structured progressions do not provide rich details of discourse, we focus on devices that more efficiently convey larger scale structure.

In the following subsections, we discuss hypermedia communicative devices for the five structured progression constructs. The devices presented do not comprise a complete list. They are instead examples of what communicative devices can exist for generated discourse. To illustrate our discussion of these devices, Figure 3 provides a screen display from a presentation generated by the Topia project's implementation.

Figure 3. A presentation generated by the Topia demo

6.1 Meaningful Order

Of the structured progression concepts, meaningful order may have the simplest hypermedia communicative devices, so we will present these first. We discussed in earlier work communicative devices for the rhetorical construct *sequence*, which we named, *bookshelf order* (and its variant *manga*), *temporal sequence*, *next and previous buttons* and *page index* [26]. These use inherently perceived linearity in hypermedia presentation structure to convey underlying linearity, or meaningful order. Bookshelf order and manga place items in the (albeit culturally-dependent) *spatial layout* pattern of left-to-right then top-to-bottom. From *timing*, temporal sequences are a clearly universal means for conveying order. *Interactive* structure provides the familiar next and previous buttons and page indices, the later of which is a spatial ordered list of buttons. These devices provide a variety of means, used independently or in combination, of having the user perceive that items in a presentation have an underlying ordering. Figure 3 shows a presentation display with next and previous buttons.

6.2 Hierarchical Structure

One perhaps obvious communicative device for hierarchical structure is *depth-first traversal*. Here, the presentation traverses the structured progression's hierarchy to provide a linear order for presenting all of its components: that of presenting all of a node's children recursively before moving on to the node's next sibling.

A survey of museum Webpages observes the regular use of buttons lined up in a "primary navigation area" bar in the display [9]. We apply this technique by providing a hierarchical navigation menu, or *outline bar*. Figure 3 illustrates the use of outline bars to convey the progression along the hierarchical structure. Outline bars in our demonstrator also *fold*, providing a view of the hierarchy that focuses on the current location. Here, each row of an outline bar shows a level of the hierarchy above (and including) the current node. When the display last child of a parent node finishes, the outline bar row under that parent disappears and in its place appears a row listing the children of the original parent's next sibling.

Outline bars, among other devices, require distinguishing between *full* and *referential* displays of topics. When a topic is the current main topic of the presentation, it, of course, has all of the media items conveying it shown in full. However, outline bar items, for example, need smaller versions of each topic's media. For text outline bars, this may be a title or brief description. Technologies exist for generating such titles from the main text descriptions [20]. If entries listed in the outline bar are images, then the bar could contain thumbnail versions of the images.

Color often communicates certain aspects of a document. Its primary communicative use in the Topia demonstrator is as the varying background colors of the textual components of the outline bar. Different background colors communicate the following three possible states of an outline component: *unvisited*, *current* and *visited*. Figure 3 shows this use of multiple background colors for outline bar entries. Distinguishing visited from non-visited links in particular enjoys in CSS standardization with wide implementation and adoption [5].

As the user progresses through the presentation, the topic just seen will turn from the current color to the visited color. The following topic's background will then switch from unvisited to current. The current color also shows under the node's ancestors in the outline bar to convey the hierarchical context of the current position. These three colors help the user perceive the current position in the hierarchy while traversing through it.

Although structured progressions provide a linear path, allowing the user to choose his or her progressive *navigation* does not necessarily conflict with communicating the document's original progressive order. In fact, navigation hypertext can enhance the user's ability to perceive the underlying structured progression. The means we present here for allowing free navigation of a fixed progression echo those in the AHA! system, which provides many additional devices for free navigation through a primarily linear progression along a primarily hierarchical presentation structure [10]. The primary navigation device our demo provides is the outline bar. The user can click on displayed topics in the outline bar to display the given topic right away, regardless of the presentation's current location in the hierarchy. Because the colors of visited and unvisited nodes in the outline bar are distinct, the user can perceive what he or she missed when straying from the default linear progression, and thus eventually be able to see the entire presentation in their own order, following their own story.

6.3 Recurring Themes

When presenting an item that is part of a cluster that appears as a recurring theme, a presentation should mention that the recurring cluster is associated with the item. We propose that an efficient way to do so is to *introduce* the cluster's theme when presenting an item from it for the first time. This introduction would be a full display of the topic, using all media associated with it. When encountering other items from the theme after its introduction, then it is enough to use a referential display of that theme's topic, letting the user recall the earlier full display. When encountering a recurring theme, the system also typically needs to generate media stating the topic relates to the item and that it is a recurring theme in the presentation. In our demo, we use simple *canned text*, as shown in Figure 3.

There are also several *interactive* communicative devices for conveying recurring themes. These provide navigation to other occurrences of the theme in the presentation. Next and previous buttons for an occurrence of a recurring theme allow traversal to other occurrences of the theme along the order of their appearance in the hierarchy. A page index for each occurrence can provide direct links to each of the theme's other occurrences, displaying these items' referential displays as link starting points. Of course, the outline bar will change with these horizontal traversals by folding, unfolding and updating the topic status colors. These links provide *horizontal* navigation that is orthogonal to the hierarchy, a helpful extra dimension to interaction. They provide another means to let users find their own stories through serendipitously discovering relations between items not automatically detected by the system.

Presentations can convey recurring items with communicative devices similar to those used for recurring clusters. Items do not need introductions in the same way recurring clusters do, since all items are inherent components of the presentation and already have full displays. However, it may not be necessary to repeat full displays of recurring items. When presenting a previously presented item, a referential display should be enough. Thus, as with recurring clusters, first occurrences of recurring items get full displays, while later occurrences get *referential displays*. Recurring items also share with recurring clusters the need for media, such as *canned text*, declaring the item as recurring. Finally, recurring items can use the recurring cluster devices for *interaction* to provide horizontal navigation to other repetitions of the same item. This horizontal navigation also helps users to understand the more complex relational context recurring items have in their structured progression discourse.

6.4 Tangents

While the hypermedia communicative devices for the other structured progression components typically define the main presentation, devices for tangents typically have the user step outside the main presentation. A simple interactive device for tangents is the informative *pop-up*. With pop-ups, the user can access optional information outside that packaged in the primary hierarchy. The user readily perceives that the systems regards this information as relatively far from the requested topic, while giving the user the freedom to discover relations the system missed.

7. IMPLEMENTATION

Figure 4 illustrates the Topia project's implementation of the architecture presented in this paper. This implementation accepts from the user a query for requesting a presentation for a given Topic related to the Rijksmuseum Amsterdam collection. It then delivers the requested presentation.

The query page is in HTML, thus presentable on any conforming browser. It provides a field for allowing the user to enter a query string. The user can also choose between HTML and XHTML+SMIL [23] as the output presentation format.

The user's HTML field entries go via HTTP to our server. The server's first step is converting the search string entered to a query encode in RQL [16], a query language for RDF. The converted RQL query then goes through our installation of the Sesame RDF query engine [1]. Sesame processes this query against our RDF-encoded metadata store.

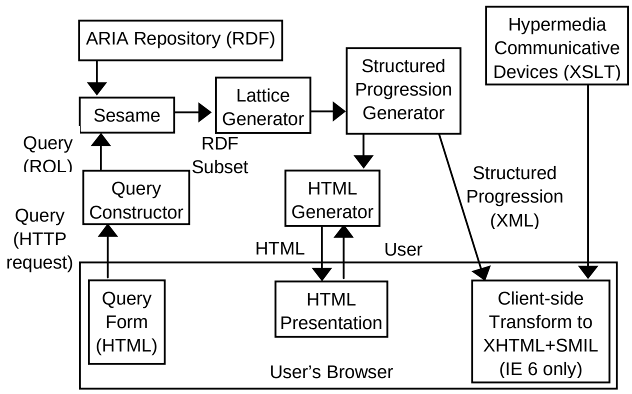

Figure 4. Topia's implementation design

The semantic network our implementation queries against is our conversion of the ARIA database of the Rijksmuseum Amsterdam [24] to RDF [18] and RDF Schema (RDFS) [6]. We applied RDF and RDFS here for such functions as defining sub-properties and a concept class hierarchy. Our RDF-encoded ARIA contains annotations for about 1250 art objects, information about them and relations between them. This query is a simple text-matching query on text fields for the artifact RDF objects. The more complex use of semantics comes in the generation of clusters and the structured progression.

We wrote Java programs for cluster and discourse generation. The subset of the ARIA RDF code matching the query runs through the lattice generator component, which generates the concept lattice. The structured progression generator then processes this lattice for determining the structured progression. We have defined an XML format for encoding this structured progression.

The presentation returned is in the format of the user's choosing: HTML or XHTML+SMIL. For the HTML version, playable on a wider variety of browsers, our architecture processes client-side a Java conversion program and transfers the resulting HTML code. To give the resulting presentation distinct screen displays, our client serves one page in HTML at a time. User clicks on these pages go back to the HTML generator, which then produces and serves the subsequent pages.

To demonstrate multimedia-based devices, style processing and XML-based Web technologies, we put in the demo an XSLT [8] transform style sheet that converts the XML-encode of Topia's structured progressions to hypermedia presentations. This XSLT code acts as a "discourse style sheet". The output presentation format is XHTML+SMIL, an open format developed by the W3C that combines SMIL timing and multimedia constructs with HTML text constructs [23]. It plays on versions 5.5 and higher of Internet Explorer [21]. Because the format handles timing behavior such as sequencing, one XHTML+SMIL file can define all of a presentation's displays.

8. FUTURE WORK

With this first phase of the Topia project over, we are now in the second phase, in which this work continues. One plan for Topia's upcoming phase is to exercise widened applicability of semantic and discourse by applying them to more domains, including medical media, business expertise management and personal image and video collections. We will build RDF annotations for such collections as we did for the Rijksmuseum Amsterdam and then see how the same techniques apply in these other domains.

A goal for the semantics-to-cluster transformation phase is coupling our general semantic processing with semantic processing tailored for an encountered domain. This could include extending the Dublin Core processing proposed by Little [19] by including the conclusions it determines in clustering. At later processing, one could design communicative devices specifically for presenting concepts that these techniques generate.

A task for the cluster-to-discourse transformation phase is Topia's upcoming exploration of minimizing item repetition and recurring themes in the resulting structured progression so that the hierarchical structure holds as much of the relational information as possible. We will also extend the hierarchies complexity by allowing it to have more levels of depth. Another cluster analysis task is accounting for all items in the data store in the generated structured progression. Items that match the original query or occur in larger clusters, which appear in the currently generated presentations, will still have dominant presence in the presentations. However, in upcoming versions of the Topia demonstrator, all items in the data store will be accessible from the presentation interface. This will give the user full exploratory potential in the media store. The user will still clearly perceive as less significant those items that are more distant in the clustering.

For the final discourse-to-presentation phase, we plan the implementation and experimentation with other communicative devices. We also plan to port all of the communicative devices currently demonstrated only with the HTML output to the XHTML+SMIL generation. Then we will use this format exclusively, since it supports all HTML and CSS text formatting along with having SMIL multimedia facilities. With the resulting unified XSLT-based environment, we can then experiment with varying the style of presenting discourse for different users and different circumstances using freely available Web technologies.

A general goal for Topia's second phase is more involvement of the user. To start with, we will provide the user, or author, with more interaction in the earlier phases of the process. This could include a feedback loop in which the user could repeatedly adapt the specifications to incrementally improve the resulting discourse. This would be similar to the user feedback provides by Little's system [19], except that in our work the feedback would alter how the structured progression is built around the selected items rather than what items are selected.

9. SUMMARY AND CONCLUSION

This paper presents the results of the first phase of the Topia project. Developments in four research areas — semantic annotations, clustering, discourse structure and hypermedia generation — form four phases in one document processing chain, generating on-demand engaging hypermedia presentations from media archives. While many approaches focus on topic domains or discourse genres to richer semantic inferences and narrative, we focus here on semantic and discourse techniques that apply more widely. With this approach, the user's request can incorporate material from any domain into the generated presentation. This is essential for generating requested presentations from materials throughout the Web and from many fields instead of from focused repositories.

For increasing domain-independence of semantic processing, the Topia project developed techniques that focus on standardized *semantic foundational constructs* such as object-relation distinctions and standardized object properties. This paper also proposed techniques for rapid human direction of deriving weighted clusters from semantics within encountered domains, such as *quick weight assignments* to semantic object and relation types. We facilitated automatically converting semantics to, and presentations from, discourse by focusing on the subset we call *structured progressions*. Finally, our approach optimizes use of structured progressions with *discourse perceptualization*. Designers define perceptualization using *communicative devices*, which map structured progression to hypermedia behavior.

We developed a system using this approach that processes queries on our semantic annotations of the Rijksmuseum Amsterdam collection, returns collections of matching items and relations between them, builds a structured progression around these items, and then generates hypermedia presentation on the requested topic. This system uses freely available implementations of Web technologies. Thus, this paper's results can readily implemented on the Web, whose wide variety of information sources is what our approach aims to facilitate exploiting.

The demonstrator and other resources for this paper are accessible at http://www.cwi.nl/~media/conferences/HT03/ .

10. ACKNOWLEDGMENTS

Funding for work on this paper came from the Topia project of the Telematica Instituut. Frank Nack of CWI and Marc Davis of the University of California provided useful information on current research in narrative. Theo van der Weide and Franc Grootjen of the Katholieke Universiteit Nijmegen help us with concept lattices. Geert-Jan Houben of the Technische Universiteit

Eindhoven, Lynda Hardman of CWI and Patrick Schmitz provided particularly helpful comments. Sorin Iacob and Frank Aldershoff of the Telematica Instituut helped initiate this work. We thank the Rijksmuseum Amsterdam for their permission to use their Website's database and media content. We also thank IBM for sponsoring the project.

REFERENCES

[1] Aidministrator Nederland B.V., Sesame.

[2] Alani, H., Kim, S., Millard, D.E., Weal, M.J., Hall, W., Lewis, P.H., and Shadbolt, N.R. Automatic Ontology-based Knowledge Extraction from Web Documents, IEEE Intelligent Systems, 18(1) (January-February 2003), 14-21.

[3] André, E., The Generation of Multimedia Documents, in. Dale, R, Moisl, H. and Somers, H. (eds.), A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text, Marcel Dekker Inc., 2000, 305-327.

[4] Bateman, J., Kamps, T., Kleinz, J. and Reichenberger., K. Towards constructive text, diagram and layout generation for information presentation, Computational Linguistics 27(3), 2001, 409-449.

[5] Bos, B., Lie, H.W., Lilley, C., and Jacobs, I. (eds). Cascading Style Sheets, Level 2. World Wide Web Consortium (W3C) Recommendation, May 1998.

[6] Brickley, D., and Guha, R.V. RDF Vocabulary Description Language 1.0: RDF Schema. World Wide Web Consortium (W3C) Working Draft, January 2003. (work in progress)

[7] Buckingham Shum, S., Uren, V., Li, G., Domingue, J. and Motta, E. Visualizing Internetworked Argumentation, In: Kirschner, P.A., Buckingham Shum, S.J. and Carr, C.S. (eds), Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making, Springer-Verlag: London, 2003, 185-204.

[8] Clark, J. (ed). XSL Transformations (XSLT) Version 1.0. World Wide Web Consortium (W3C) Recommendation, November 1999.

[9] Czajka, K. Design of interactive and adaptive interfaces to exploit large media-based knowledge spaces in the domain of museums for the fine arts. Masters Thesis, University of Applied Science Darmstadt, June 2002.

[10] De Bra, P. and Calvi, L. AHA! An open Adaptive Hypermedia Architecture. The New Review of Hypermedia and Multimedia, 1998, 115-139.

[11] Decker, S., Melnik, S., van Harmelen, F., Fensel, D., Klein, M., Broekstra, J. Erdmann, M. and Horrocks, I.,The Semantic Web: The roles of XML and RDF, IEEE Internet Computing, 15(3) (October 2000), 63-74.

[12] Dublin Core Community. Dublin Core Element Set, Version 1.1, 1999.

[13] Ganter, B., and Wille, R., Applied Lattice Theory: Formal Concept Analysis. Preprints http://wwwbib.mathematik.tu-darmstadt.de/Math-Net/Preprints/Listen/pp97.html, 1997.

[14] Geurts, J., Bocconi, S., van Ossenbruggen, J.R., Hardman, L. Towards Ontology-driven Discourse: From Semantic Graphs to Multimedia Presentations, CWI Technical Report INS-R0305, May 2003.

[15] Heflin, J. and Hendler, J., Searching the Web with SHOE, AAAI-2000 Workshop on AI for Web Search, 2000.

[16] Karvounarakis, G., Alexaki, S., Christophides, V., Plexousakis, D. and Scholl, M. RQL: A Declarative Query Language for RDF, In: Proceedings of the Eleventh International World Wide Web Conference (WWW2002), May 2002, Honolulu, Hawaii, USA, ACM Press, 592-603.

[17] Kamps, T., Diagram Design : A Constructive Theory, Springer Verlag, 1999.

[18] Lassila, O. and Swick, R.R. (eds), Resource Description Framework (RDF) Model and Syntax Specification. World Wide Web Consortium (W3C) Recommendation, February 22nd, 1999.

[19] Little, S., Geurts, J. and Hunter, J., Dynamic Generation of Intelligent Multimedia Presentations through Semantic Inferencing. In: Proceedings of the Sixth European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2002), Springer, September 2002, 158-189.

[20] Mani, I. and Maybury, M. T. Advances in Automatic Text Summarization, MIT Press, 1999.

[21] Microsoft, Inc. Internet Explorer 6. 2002.

[22] Modha, D.S. and Spangler, W.S., Clustering Hypertext with Applications to Web Searching, In: Proceedings of the Eleventh ACM Conference on Hypertext and Hypermedia (HT00), San Antonio, USA, June 2000, ACM Press, 143-152.

[23] Newman, D., Schmitz, P., and Patterson, A. (eds). XHTML+SMIL Profile. World Wide Web Consortium (W3C) Note, 31 January 2001.

[24] van Ossenbruggen, J., Geurts, J., Cornelissen, F, Rutledge, L. and Hardman., L Towards Second and Third Generation Web-Based Multimedia, In: Proceedings of the Tenth International World Wide Web Conference (WWW10), May 2001, Hong Kong, ACM Press, 479-488.

[25] Rijksmuseum Amsterdam, Rijksmuseum Amsterdam Website. http://www.rijksmuseum.nl/.

[26] Rutledge, L., Davis, J., van Ossenbruggen, J., and Hardman, L. Inter-dimensional Hypermedia Communicative Devices for Rhetorical Structure. In: Proceedings of the International Conference on Multimedia Modeling 2000 (MMM00), Nagano, Japan, November 2000, World Scientific, 89-105.

[27] Walker, J., Piecing together and tearing apart: finding the story in afternoon, In: Proceedings of the Tenth ACM Conference on Hypertext and Hypermedia (HT99), Darmstadt, Germany, February 1999, ACM Press, 111-117.

[28] Weal, M.J., Bernstein, M. and Millard, D.E., On Writing Sculptural Hypertext. In: Proceedings of the Thirteenth ACM Conference on Hypertext and Hypermedia (HT02), Maryland, USA, June 2002, ACM Press, 65-66.