XML Topic Maps and Semantic Web Mining

Benedicte Le Grand <Benedicte.Le-Grand@lip6.fr>
Michel Soto <Michel.Soto@lip6.fr>
David Dodds <drdodds42@hotmail.com>

Abstract

Navigation and information retrieval on the Web are not easy tasks; the challenge is to extract information from the large amount of data available. Most of this data is unstructured, which makes the application of existing data mining techniques to the Web very difficult. However, new semantic structures that improve the results of Web Mining are currently being developed in the Web. This paper presents how one of these semantic structures - XML topic maps - can be exploited to help users find relevant information on the Web. This paper is organised as follows: first, we introduce XML topic maps in the context of Tim Berners-Lee's Semantic Web vision. Then, we show how topic maps allow to characterise and "clean" Web data through the definition of a profile; this is achieved by the analysis of a lattice generated by a classification algorithm - called Galois algorithm. This profile may be used to evaluate the relevance of a web site with regard to a specific request on a traditional search engine. We finally explain how data on the Web can be clustered, organised and visualised in different ways so as to enhance users' navigation and understanding of these documents. Additional characterization of web data is provided by joint author David Dodds method for computing context and similarity. It uses OpenCYC DAML-OIL and Upper Ontology to leverage perception and development of the characterization. We present the results obtained with topic maps and compare and contrast our similarity measures and clusters, the ones obtained with the Galois lattice and the ones obtained with OpenCYC DAML-OIL.

1. Introduction

Navigation and information retrieval on the Web are not easy tasks; the challenge is to extract information from the large amount of data available. Most of this data is unstructured, which makes the Web application of existing data mining techniques. However, new semantic structures that improve the results of Web Mining are currently being developed in the Web.

This paper presents how one of these semantic structures - XML topic maps - can be exploited to help users find relevant information in the Web. This paper is organised as follows: first, we introduce XML topic maps in the context of Tim Berners-Lee's Semantic Web vision [2]. Then we show how topic maps allow to characterise Web sites through the definition of a profile. This profile may be used to evaluate the relevance of a web site with regard to a specific request on a traditional search engine. We finally explain how data on the Web can be clustered,
organised and visualised in different ways so as to enhance users' navigation and understanding of these documents.

2. XML Topic Maps and the Semantic Web

Finding information on the Web is very difficult. Search engines may return hundreds or more links to users' queries - provided that the proper keywords are used. Choosing the most relevant Web sites to explore is not trivial, because no semantics help evaluate the relevance of each hit. The next step is not easier: once a link is chosen, navigation is not always intuitive. Users can get lost easily: they may not find the information they are looking for even though it does exist. Sometimes they do not manage to go back to a page they have already visited. This is due to the lack of structure of the Web. Therefore it is necessary to add structure and semantics as well as to provide a mechanism which allows a more precise description of data on the Web. According to Tim Berners-Lee from W3C [2]:

"The Semantic Web is a vision: the idea of having data on the web defined and linked in such a way that it can be used by machines - not just for display purposes, but for using it in various applications."

This Semantic Web can be achieved by adding semantic structures to the current Web. Many candidate techniques have been proposed, such as * semantic networks [16], conceptual graphs [5], the W3C Resource Description Framework (RDF) [14] and XML Topic Maps [11].

Semantic networks are basically directed graphs (networks) consisting of vertices linked by edges. Edges express semantic relationships between the vertices.

The conceptual graphs theory developed by Sowa [10] is a language for knowledge representation based on linguistics, psychology and philosophy.

RDF data consists of nodes and attached attribute/value pairs. A node can be any web resource (page, server, basically anything for which you can give a URI), or an other instance of metadata. Attributes are named properties of the nodes, and their values are either atomic (character strings, numbers, etc.), metadata instances or other resources. This mechanism allows us to build labelled directed graphs.

Topic maps, as defined in ISO/IEC 13250 [10], are used to organise information in a way that can be optimised for navigation. Topic maps were designed to solve the problem of managing large quantities of unorganised information. Information is not useful if it cannot be found or linked to. In the paper-publishing world, there are several mechanisms to organise and index the information contained within a book or document. Indexes allow readers to go directly to the portion of the document that is relevant to their information needs. Topic maps can be thought of as the online equivalent of printed indexes. Topic maps are also a powerful way to manage link information, much as glossaries, cross-references, thesauri and catalogs do in the paper world. Topic Maps allow users to create a large quantity of metadata and tightly interconnected data. They constitute a kind of semantic network above the data themselves.
A new specification which aims at applying the topic map paradigm to the Web is currently being written; this initiative is called XTM (XML Topic Maps) [11]. XML Topic Maps allow to structure data on the Web and therefore make Web mining more efficient.

It was recently proven that the RDF and Topic Map models could inter-operate at a fundamental level [9]. Both standards are concerned with defining relationships between entities with identity. Each language can be used to model the other.

All the techniques described previously have the same goals and many of them are compatible. We decided to further investigate XML Topic Maps and study how they could enhance Semantic Web Mining.

We aim at helping users find relevant information and we contribute to this difficult task at three levels:

1. by evaluating Web site relevance to users needs based on semantic criteria;

2. by filtering the topic map; the topic map profile constitutes a reference that can be used to select the most semantically significant objects (called regular objects); this allows to identify the major subjects which the topic map deals with and to discard less relevant topics;

3. by enhancing navigation on the Web through the aggregation of conceptually related topics and through the visualisation with different scales - or levels of details.

The different steps of topic maps - or Web site\textsuperscript{1} - analysis are represented in figure 1:

\textsuperscript{1} In the following, we will use the term "topic map" which is more general than "Web site". Topic maps may apply to any kind of data.
We propose to achieve the first goal by defining the profile of a topic map - and later applying it to Web sites. These profiles characterise topic maps and help evaluate their relevance to users' information needs. The computation of this sort of topic map "DNA" and its interpretation are described in section 4.

The topic map may contain topics that are not semantically significant or not much related to others. We call them singular topics. These may be eliminated from the topic map so as to clean it, as explained in section 4.2. That is our second goal.

Our third contribution consists in enhancing navigation and information retrieval within a Web site. Information retrieval varies according to the needs of the user. If he looks for an answer to a specific question, query languages (like "tolog" [6]) are appropriate. Their strength is to exploit the relationships between objects, which allows to answer questions better. For example, one may seek the Beatles' songs that were not written by John Lennon. This kind of information would be difficult to find with a traditional search engine.

If the subject of interest is clearly identified, it is easy to explore the corresponding topic in the topic map. This topic can be reached through a list of topics, for example an alphabetical list. Tools to navigate in topic maps have been designed so that any topic can be reached in seven mouse clicks at most.
If the user has no precise question nor any clear subject of interest, none of the search techniques described above can apply. This is the case of a beginner user who wishes to have a global understanding of the topic map so as to decide where to start his navigation. Therefore she first needs a simplified view of the topic map, with no detail, then she can decide to see more precise information as his subject of interest gets clearer. Let us compare this to geographical maps: there is no point in displaying very specific data on a map of the world; however, more and more details may be added as the user focuses on some part of the map. We propose to use clustering algorithms to group semantically related topics together at different abstraction levels. Cluster computation and visualisation are described in section 5.

Figure 1 shows that topic maps - or Web sites - characterization, filtering and clustering are deduced from the results of a conceptual classification algorithm based on Formal Concept Analysis and Galois connections. This algorithm is presented in section 3.

3. Conceptual classification algorithm

The starting point of our Web analysis is a conceptual classification algorithm based on Formal Concept Analysis and Galois connections. FCA is a mathematical approach to data analysis which provides information with structure. FCA may be used for conceptual clustering as shown in [4] and [12]. Let us first define a few terms:

- an object is a topic or an association of the topic map,
- the objects have characteristics called properties. We describe how these properties are determined in 3.1.

A profile allows us to characterise a topic map in a structural way. With this footprint, we can tell if the topic map is specific or general. We can also tell if the objects of a topic map are similar or very different. In order to characterise objects, we use a Galois algorithm to classify the objects conceptually. This algorithm groups objects in concepts according to the properties they have in common. It is very powerful because it performs a semantic classification without having to express semantics explicitly. We first describe how the objects and their properties are generated from a topic map. Then we describe Galois lattices and detail the statistical computations made on the objects. We finally explain how the profile is determined.

3.1. Objects and properties generation

The generation of objects and properties is a 2-step process.

- First step

Every time there is an element with an identifier (that is an id attribute), a new object is created. The name of the object is the value of the identifier. As stated in the DTD (Document Type Definition), all topics and associations of the topic map have an identifier, so there will be as many objects as the number of topics and associations.
The properties of an object correspond to the values of this object's attributes (including the value of the id attribute), as well as the values of its children attributes. These properties are weighted (for instance, the weight of the values of instanceOf attributes may be greater than the weight of href attributes).

Generation of object and properties (first step):

Example: consider the following extract of a topic map about music, written by Kal Ahmed as an XML document:

```xml
<topic id="t-the-clash"> 
  <instanceOf> 
    <topicRef xlink:href="tt-band"/>
  </instanceOf>
  <baseName>
    <baseNameString>The Clash</baseNameString>
    <variant>
      <parameters>
        <topicRef xlink:href="http://www.topicmaps.org/xtm/1.0/psi-sort"/>
      </parameters>
      <variantName>
        <resourceData>clash the</resourceData>
      </variantName>
    </variant>
    <variant>
      <parameters>
        <topicRef xlink:href="http://www.topicmaps.org/xtm/1.0/psi-sort"/>
      </parameters>
      <variantName>
        <resourceData>Clash, The</resourceData>
      </variantName>
    </variant>
  </baseName>
</topic>
```

An XML document is made of elements limited by tags and is hierarchically structured. In the above example, topic, instanceOf and baseName are elements. An element may have characteristics called attributes. The attributes of an element are declared inside the opening

---

tag of the element. The element topic has an attribute id with a value tt-clash. The element instanceOf has no attribute.

When parsing the topic map, we find a topic which has an identifier with the value t-the-clash. An object t-the-clash is thus created.

In order to determine the properties of these objects, we look for all the attributes of this element. In this case, the only one is the identifier.

Then, we have a look at the children of this element (that is all the XML elements included in the element) to find their attributes. We repeat this for all the children.

In this example, the analysis of this abstract of the topic map creates an object t-the-clash with the properties t-the-clash (weight e.g. 0.5), tt-band (weight e.g. 2) and http://www.topicmaps.org/xtm/1.0/psi-sort (weight e.g. 0.2). The weights shown here correspond to one possible scenario - in which the type of a topic (weight 2) is more important than its name (weight 0.5), its occurrences (weight 0.2) or the associations it is involved in (weight 1).

In the same way, the analysis of the following abstract:

```xml
<topic id="tt-band"> <instanceOf> <topicRef xlink:href="tt-music"/> </instanceOf> <baseName> <baseNameString>Band</baseNameString> </baseName> </topic>
```

leads to the creation of an object tt-band with the properties tt-band (weight e.g. 0.5) and tt-music (weight e.g. 2).

Finally, the following example describes an association:

```xml
<association id="assoc6"> <instanceOf> <topicRef xlink:href="at-recorded"/> </instanceOf> <member> <instanceOf> <topicRef xlink:href="tt-band"/> </instanceOf> <topicRef xlink:href="t-the-clash"/> </member> <member> <instanceOf> <topicRef xlink:href="tt-track"/> </instanceOf> <topicRef xlink:href="t-i-fought-the-law"/> </member> </association>
```

The object assoc6 is created with the properties assoc6 (weight e.g. 0.5), at-recorded (weight e.g. 2), tt-band, t-the-clash, tt-track and t-i-fought-the-law.

So far, the properties of an object are only intrinsic properties. For example, the object t-the-clash takes a part in the association assoc6, but this does not appear in its properties yet, since the association is not declared inside the topic with the identifier t-the-clash. The second step takes these characteristics into account.

- Second step
The second steps adds non intrinsic properties to the objects by cross-referencing the data. In fact, for an object O with a set of properties P, each property P becomes an object with O (amongst others) as a property. The properties of an object then become its intrinsic properties and all the properties that were added recursively.

In the previous examples, the object assoc6 has the properties assoc6, tt-band and t-the-clash. The property assoc6 is added to the objects tt-band and t-the-clash. So all the objects know the associations they appear in.

Moreover, the object t-the-clash has the property tt-band. The data is cross-referenced by adding t-the clash to the object tt-band. This example illustrates a new type of information, which was not present in the first step: the object tt-band now knows it has an instance of t-the-clash. In the preceeding scenario, t-the-clash was the only one to know its membership.

In the end, tt-band has the properties tt-band (weight e.g. 0.5), tt-music(weight e.g. 2), t-the-clash (weight e.g. 1), assoc1 (weight e.g. 1), assoc2 (weight e.g. 1) and assoc6 (weight e.g. 1). The object t-the-clash has the characteristics http://www.topicmaps.org/xtm/1.0/psi-sort (weight e.g. 0.2), tt-band, t-the-clash (weight e.g. 0.5), assoc1 (weight e.g. 1), assoc2 (weight e.g. 1) and assoc6 (weight e.g. 1).

Note that the properties assoc1 and assoc2 correspond to other associations in which tt-band and t-the-clash appear. These associations are present in the topic map but not in the
extracts we presented.

3.2. Introduction to Galois lattices

The notion of Galois lattice to express a relationship between two sets is the basis of several conceptual classification methods. This notion was introduced by Birkhoff [3] and by Barbut and Monjardet [1]. Galois lattices consist in grouping objects into classes that materialise concepts of the domain under study. Individual objects are discriminated according to the properties they have in common. This algorithm is very powerful as it performs semantic classification. Topic maps are semantic structures themselves, but they may be very large and complex, so this algorithm is interesting to extract more semantics from them. The algorithm we implemented is based on the one that was proposed in [7].

Let us first introduce the basic concepts of Galois lattices.

Let E and E' be two finite sets (E consists of a set of objects and E' is the set of these objects' properties), and \( R \subseteq E \times E' \) a binary relation between these two sets. Figure 2 shows an example of binary relation between two sets. According to Wille's terminology [13], the triple \((E, E', R)\) is a formal context which corresponds to a unique Galois lattice. It represents natural regroupings of \( E \) and \( E' \) elements.

Let \( P(E) \) be the powerset of \( E \) and \( P(E') \) the powerset of \( E' \). Each element of the lattice is a couple, also called concept, noted \((X, X')\). A concept is composed of two sets \( X \in P(E) \) and \( X' \in P(E') \) which satisfy the two following properties:

\[
X' = \mathcal{P}(X) \quad \text{where} \quad \mathcal{P}(X) = \{ z' \in E' \mid \forall z \in X, xRx' \} \\
X = \mathcal{P}(X') \quad \text{where} \quad \mathcal{P}(X') = \{ z \in E \mid \forall z' \in X', xRx' \}
\]

Figure 2.

A partial order on concepts is defined as follows:

Let \( C_1=(X_1, X'_1) \) and \( C_2=(X_2, X'_2) \). Then:

\[
C_1 < C_2 \iff X'_1 \subseteq X'_2 \Rightarrow X_1 \subseteq X_2
\]

Figure 3.

This partial order is used to draw a graph called a Hasse diagram, as shown on figure 2. There is an edge between two concepts \( C_1 \) and \( C_2 \) if \( C_1 < C_2 \) and there is no other element \( C_3 \) in the lattice such as \( C_1 < C_3 < C_2 \). In a Hasse diagram, the edge direction is upwards. This graph can be interpreted as a representation of the generalisation / specialisation relationship between couples, where \( C_1 < C_2 \) means that \( C_1 \) is more general than \( C_2 \) (and \( C_1 \) is above \( C_2 \) in the diagram).
4. Characterisation of topic maps: conceptual profile

4.1. Calculating the statistics for every object.

We calculate statistics for every object of the topic map. We compute a weighted mean of these statistics. Each object has a weight which is assigned according to its importance in the topic map (the number of occurrences of the object in the XML source file).
Consider any object $O$. It is characterised by a vector with 6 components:

- The first component ($A_1$) is the percentage of concepts in the lattice that include the object in its extensions. This value tells if $O$ is present in many concepts of the lattice. A low value for $A_1$ may indicate that $O$ has few common characteristics with other objects. However, the other components allow to increase our knowledge.

- The second component is the maximum number of objects with which $O$ is grouped, divided by the total number of objects. We have to select the concept containing $O$ and with the largest number of objects. We add a constraint on this concept: it must contain at least one property. Indeed, we wish to group objects with common properties. The component $A_2$ shows if $O$ is grouped with many other objects. However, this value is a maximal value. The validity of $A_2$ must be checked using $A_3$.

- $A_3$ is the mean number of objects with which $O$ is grouped divided by the number of objects. This time we can tell if $O$ is linked to a large number of objects and determine the significance of $A_2$. If $A_3$ is high, then there is a concept including $O$ and many other concepts. On the other hand, if $A_3$ is low, $O$ is grouped with very few other objects. That concept is thus an exception and we should not base our analysis on it.

- Let $S$ be the set of objects which are grouped with $O$ in one or more concepts of the lattice; these objects have at least one of $O$'s properties. $A_4$ is the maximum number of properties $O$ shares with the objects contained in $S$, divided by the total number of objects in the lattice. This component is deduced from the concept containing the object $O$ and which includes the greatest number of properties. Again, we add a constraint on this concept: it must contain at least two objects, that is at least one object different from $O$. We want to evaluate the number of shared properties, thus we need at least one object with which $O$ shares them. $A_4$ tells whether the objects which are close to $O$ share many common properties with $O$ or not. Objects are more similar when they share more properties. This similarity can either be structural or conceptual. However, this value is a maximum number which must be validated with $A_5$.

- $A_5$ is the mean number of properties $O$ shares with other objects, divided by the total number of properties. This tells the degree of significance of $A_4$.

- Finally, $A_6$ is about the topic map itself, and not about the lattice. It is the number of occurrences of the object in the topic map divided by the number of occurrences of objects of the same type (either topic or association). $A_6$ is used to compute the topic map's profile. This profile represents the characteristics of a mean object. Each component of this vector is the mean of the components of each object in the topic map, with a weight $A_6$ given to each of these objects. Thus, objects with a high number of occurrences in the topic map will influence the profile much more than objects with few occurrences.

Note that the first five components are deduced from an analysis of the lattice whereas the last component only depends on the XML document itself.
4.2. Topic map profile and selection of objects

When the statistics have been computed for every topic and every association, the profile can be deduced. It is a vector for which each component is a mean of the components of all the objects with the weight $A_6$ of each object. For $N$ objects $O_1, O_2, ..., O_N$, each component $A_i$ of the profile vector $P$ is computed as follows:

$$P.A = \sum_{k=1}^{N} O_k.A_i \cdot O_k.A_i$$  \hspace{1cm} (3)

where $O_j.A_i$ is the component $A_i$ of the $j$-th object.

We wish to keep the most relevant objects, that is the ones which share "many" common properties with "many" other objects. These objects are called regular objects because they are semantically more significant than others. The significance of the words "many" (properties) and "many" (objects) is given by the topic map profile. A regular object is associated to at least as many objects and shares as many properties as the profile.

Among the statistics presented in section 4.1, the values $A_3$ and $A_5$ are more relevant than $A_2$ and $A_4$: maximum values may not give a reliable information because they may correspond to an exception. The comparison between the objects and the profile is thus done using the components $A_3$ and $A_5$.

A regular object $O$ must verify the following conditions:

$$O.A \geq \text{profile}.A_i \hspace{1cm} (4)$$

This should be refined using the standard deviation. The standard deviation for $A_3$ is the mean distance between an object's value of $A_3$ and the profile's value of $A_3$.

$$\text{std.dev}_{A3} = \frac{\sum |O.A - P.A|}{N}$$ \hspace{1cm} (5)

For $A_5$, the standard deviation is computed in the same way:

$$\text{std.dev}_{A5} = \frac{\sum |O.A - P.A|}{N}$$ \hspace{1cm} (6)
Thus, a regular object is defined as follows:

\[ O_{A_{3} > \text{std} \cdot \text{der} \cdot A_{5} \cdot P_{A}} \]
\[ O_{A_{3} > \text{std} \cdot \text{der} \cdot A_{5} \cdot P_{A}} \]

The regularity conditions can be changed (to be more or less restrictive) with a coefficient (C). Thus, a regular object meets the two following requirements:

\[ O_{A_{3} + C \cdot \text{std} \cdot \text{der} \cdot A_{5} \cdot P_{A}} \]
\[ O_{A_{3} + C \cdot \text{std} \cdot \text{der} \cdot A_{5} \cdot P_{A}} \]

A non regular object is called a singular object - it conveys little semantics. When the objects of the topic map are submitted to these conditions, singular objects are eliminated. When C increases, more objects are suppressed since the conditions are harsher.

After this selection, we have a new list of objects which are used as an input for the Galois classification algorithm. A new lattice is generated and the statistics computed on this new panel of objects provide a new profile. We can thus select once again the regular objects for this new footprint of the topic map. The new regular objects are used again as an input for the Galois algorithm, etc. until all the objects become regular. This happens when no object is eliminated. The algorithm stalls and we get a stable list of regular objects which we must group together.

4.3. Results

Several topic maps - of different sizes and subjects - have been analysed. The figure 4 displays the distribution of objects in three topic maps. The coordinates of the center of a disk correspond to the values of A3 and A5 attributes. The diameter of a circle is proportional to the number of objects which have these values for the A3 and A5. All the objects of the simple topic map are very close. This topic map is qualified of "homogeneous", which means that all topics have the same semantic significance. Music and icc are "heterogeneous" structures. The objects in the lower left corner have low values for A3 and A5: they are "singular", i.e. they are not much related to other objects in the topic map. These topic maps can be filtered easily by eliminating these singular objects.
Figure 12. Fig. 4. Topics conceptual distribution
Figure 5 illustrates the filtering of six topic maps. Some topic maps can be simplified a lot; this is the case of discovery. On the other end, after the last iteration, xmle99mp still contains almost 70% of its topics. This means that it is more difficult to filter this topic map: all topics have the same semantic value.

5. Topic map clustering and visualisation

5.1. Clustering algorithm

The Galois lattice which is generated from a topic map contains some concepts which are made up of a set of topics which share common properties. The lattice gives an exhaustive description of the input data and the number of concepts generated may be very high. The concept lattice shown in the figure 3 is quite complex although it was generated from a small topic map (which contains 46 objects). We wish to group topics together into clusters in order to provide different level of detail (or scales) of the topic map. We propose to extract a tree from the Galois lattice. The concepts contained in this tree are the clusters. Thus, we have a hierarchy of clusters. The root of the tree contains all the topics; it is a gross cluster which provides no additional information. The next level groups some topics together, the next level executes a finer grouping of topics, etc. The number of levels of detail is given by the depth of the tree.
Many clustering algorithms exist; we chose to implement a clustering algorithm based on Galois conceptual classification. The clusters we generate are thus conceptually and semantically relevant. This algorithm also allows us to use the generalisation/specialisation relationship inherent to the Galois lattice.

To build the tree of clusters, we start from the representation which provides the greatest level of detail. Every cluster corresponds to an object: the objects are not grouped together. We begin to construct the leaves of the tree: these clusters correspond to the fathers of the upper bound of the lattice (which is represented at the bottom of the lattice on the Hasse diagram). This is the most specific level.

For each leaf, we select one unique father which is a generalisation of the concept. This selection is done according to a hierarchy of criteria which will be developed in the following. One father is selected for each selected node, and so on until the lower bound of the lattice is reached. At the end of this process, a tree is created. Each level of the tree contains clusters which correspond to a level of detail.

We defined a hierarchy of selection criteria when a concept has several fathers in the lattice.

- first, we consider the distance between each father and the lower bound of the lattice (this distance corresponds to the minimum number of edges between them).

- if one of the fathers' distance to the lower bound is smaller than the others, this node is selected. Being at lower distance from the lower bound means that this concept is semantically richer.

- if several nodes are at a minimum distance from the lower bound, we compare the sum of the weights of the properties contained in their intention. The node with the highest value is selected.

- if several fathers meet this requirement, the algorithms chooses the one which minimises the total number of branches in the tree. If this condition is not unique, different scenarios are considered, one for each possible father.

**5.2. Cluster analysis**

Once the tree of clusters is generated, different measures may be computed, e.g. the proportion of concepts of the initial lattice which were not selected to be clusters.

The depth of the tree is interesting because it indicates the number of navigation levels that may be provided to the user. We also study the distribution of clusters at each abstraction level. If a cluster has no father, it means that it cannot be generalised. On the other hand, a cluster with no children corresponds to the most specific level.

We may also compute the distances between clusters. The distance between two clusters may be the average - or minimum, or maximum - distance between two objects (one in each cluster). Let O1 and O2 be two objects. Let P1 be the set of properties of O1 and P2 the set of properties
of O2. Let INTER be the intersection of P1 and P2, and UNION the union of P1 and P2. The similarity between O1 and O2 is defined as:

\[
\text{sim}(O1,O2) = \frac{\sum_{o\in \text{INTER}} w_{ij}}{\sqrt{\sum_{o\in O1} w_{ij}^2} \sqrt{\sum_{o\in O2} w_{ij}^2}}
\]

The distance between O1 and O2 is given by (2):

\[
\delta(O1,O2) = \frac{1}{1+\text{sim}(O1,O2)}
\]

5.3. Cluster representation

The levels of detail are symbolised by different colours. At each abstraction level, clusters are represented by portions of a disk, as shown in figure 6. Each cluster's size is proportional to the number of children this concept has. When the pointer of the mouse is over a cluster, its extension - the set of topics contained in this cluster - or intention - the set of these topics' properties - is displayed. When the user clicks on a part of the disk, this cluster becomes the current context - i.e. the whole disk - and its content is displayed in greater detail. The disk in the upper left corner represents a global view of the topic map before focusing on a specific cluster.

The figure 6 shows the results of this clustering algorithm on our example topic map about music. These representations are SVG (Scalable Vector Graphics) graphics [15]. SVG is a language for describing two-dimensional graphics in XML. SVG drawings can be interactive and dynamic. SVG leverages and integrates with other W3C specifications and standards efforts. By leveraging and conforming to other standards, SVG becomes more powerful and makes it easier for users to learn how to incorporate SVG into their Web sites.
5.4. David Dodds Topic Map Code and SVG Code

Further code and discussion of David Dodds material may be found at openmeta.org "XML 2001 December presentation". (after Dec 4 2001)

```xml
<?xml version="1.0"?>
<!-- notice: at the web site = http://www.openmeta.org, there will be further-->
<!-- code and descriptive text as of Dec 2001 -->
<!DOCTYPE topicMap

PUBLIC "~/TopicMaps.Org//DTD XML Topic Map (XTM) 1.0//EN"
"file://usr/local/home/gromit/xml/xtm/xml.dtd">
<!-- the "file usr.." above was used as an illustration file location for the dtd, that file-->
<!-- instance occurs in the XTM home site example, of an XML program "shell" for a TM -->
<!--The topics, occurrences, and associations below are not from the XTM home site-->
<topicMap xmlns="http://www.topicmaps.org/xtm/1.0/"
xmlns:xlink="http://www.w3.org/1999/xlink"
xml:base="http://www.openmeta.org/People/David/Circuit25v7">
<!-- topics, associations, and merge map directives if any go here next-->
<!-- mytopref ignore a.r.m.s. photographs circles arrows paragraphs-->
<topic id="Gravitic-Anomalizer-circuit">
<baseName>
```
Electric Circuit Ground

ground

20
<resourceRef xlink:href="www.openmeta.org/2001/svgpicture.svg ..."/>
</occurrence>
<instanceOf>
<topicRef xlink:href="#book-content"/>
</instanceOf>
<scope>
<topicRef xlink:href="#text-thing"/>
<topicRef xlink:href="#published-book"/>
</scope>
<resourceRef xlink:href=" URL of this WROX Pro Metadata book ..."/>
</occurrence>
</topic>
<association>
<instanceOf>
<subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#class-instance"/>
</instanceOf>
<scope>
<topicRef xlink:href="battery-ground-connection"/>
</scope>
<member>
<roleSpec>
<subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#class"/>
</roleSpec>
<topicRef xlink:href="#connected-pairs"/>
</member>
<member>
<roleSpec>
<subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#instance"/>
</roleSpec>
<topicRef xlink:href="#battery"/>
<topicRef xlink:href="#ground"/>
</member>
</association>
<association>
<instanceOf>
<subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#class-instance"/>
</instanceOf>
<scope>
<topicRef xlink:href="battery-atright-ground"/>
</scope>
<member>
<roleSpec>
<subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#class"/>
</roleSpec>
<topicRef xlink:href="#atright-pairs"/>
</member>
<member>
<roleSpec>
<subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#instance"/>
</roleSpec>
<topicRef xlink:href="#battery"/>
<topicRef xlink:href="#ground"/>
Electronic Circuit Antenna
<topic id="circuit-design">
  <baseName>
    <baseNameString>Electrical Circuit Design</baseNameString>
  </baseName>
  <scope>
    <topicRef xlink:href="#electronic-circuit"/>
    <topicRef xlink:href="#line-drawing"/>
    <topicRef xlink:href="#picture"/>
  </scope>
  <baseNameString>circuit-design</baseNameString>
</baseName>
</topic id="electronic-circuit">
  <baseName>
    <baseNameString>electronic circuitry</baseNameString>
  </baseName>
  <scope>
    <topicRef xlink:href="#electrical-power-source"/>
    <topicRef xlink:href="#circuit-design"/>
    <topicRef xlink:href="#line-drawing"/>
    <topicRef xlink:href="#picture"/>
  </scope>
  <baseNameString>electronic-circuit</baseNameString>
</baseName>
</topic id="line-drawing">
  <baseName>
    <baseNameString>Line Drawing</baseNameString>
  </baseName>
  <scope>
    <topicRef xlink:href="#electrical-power-source"/>
    <topicRef xlink:href="#circuit-design"/>
    <topicRef xlink:href="#electronic-circuit"/>
    <topicRef xlink:href="#picture"/>
  </scope>
  <baseNameString>line-drawing</baseNameString>
</baseName>
</topic id="picture">
  <baseName>
    <baseNameString>Picture</baseNameString>
  </baseName>
  <scope>
    <topicRef xlink:href="#svg-program-output"/>
    <topicRef xlink:href="#circuit-design"/>
    <topicRef xlink:href="#electronic-circuit"/>
    <topicRef xlink:href="#line-drawing"/>
  </scope>
  <baseNameString>picture</baseNameString>
</baseName>
</topic id="text-thing">
  <baseName>
    <baseNameString>TextThing</baseNameString>
  </baseName>
  <scope>
    <topicRef xlink:href="#rdf-schema"/>
    <topicRef xlink:href="#clips-electronic-circuit-code"/>
    <topicRef xlink:href="#line-drawing"/>
  </scope>
</baseName>
</topic id="text-thing"/>
<!-- this SVG file is the picture/diagram of the electronic schematic -->

<svg width="9in" height="7in" viewBox="0 0 640 344">
  <g id="battery">
    <path d="M 42 214 L 110 214 M 110 200 L 110 226 M 122 206 L 122 220 M 122 214 L 166 214 " style="fill:none; stroke:black; stroke-width:2"/>
  </g>

  <g id="ground">
  </g>
</svg>
Example anim01 - demonstrate deBono diagram animation and use of SVG as visual reification of Lakoff metaphors

rect x="1" y="1" width="253" height="199"
fill="none" stroke="blue" stroke-width="2" />

text id="uplabel" x="230" y="20"
style="font-family:Verdana; font-size:12.333; fill:darkblue">
UP
</text>

text id="downlabel" x="230" y="180"
style="font-family:Verdana; font-size:12.333; fill:darkblue">
DOWN
</text>

text id="goallabel" x="125" y="42"
style="font-family:Verdana; font-size:12.333; fill:darkblue">
GOAL
</text>

g id="leftfunnelside">
<path d="M 99 180 L 99 57"
style="fill:none; stroke:black; stroke-width:10"/>
</g>

g id="rightfunnelside">
<path d="M 153 57 L 153 180 "
style="fill:none; stroke:black; stroke-width:10"/>
</g>

rect id="arrowstreamer" x="108" y="190" width="5" height="5">
<animate attributeName="x" attributeType="XML"
begin="0s" dur="9s" fill="freeze" from="108" to="142" />
<animate attributeName="height" attributeType="XML"
begin="0s" dur="9s" fill="freeze" from="5" to="-157" />
<animateColor attributeName="fill" attributeType="CSS"
from="rgb(0,0,255)" to="rgb(128,0,0)"
begin="0s" dur="9s" fill="freeze" />
</rect>

rect id="particlestreamer" x="147" y="190" width="5" height="5">
<animate attributeName="y" attributeType="XML"
begin="0s" dur="9s" fill="freeze" from="190" to="42" />
<animateColor attributeName="fill" attributeType="CSS"
from="rgb(0,0,255)" to="rgb(128,0,0)"
begin="0s" dur="9s" fill="freeze" />
</rect>
6. Conclusion and further work

This article presented how XML topic maps could be exploited to help users find relevant information on the Web. This contribution is at several levels: first, we characterise Web sites by defining their profile. This may be used to evaluate Web sites relevance with regard to a specific query. Second, our analysis identifies topics that have no interest - semantically speaking - which allows to "clean" the topic map. Finally we showed how we could enhance navigation by clustering Web pages and displaying them with different levels of details.

These results were deduced from the analysis of Galois lattices generated from Web sites with a conceptual classification algorithm. This algorithm is very powerful as it groups topics semantically.

In the future, we will study Web sites clusters in more details. For example, we noticed that some of the clusters are less relevant than others; it may thus be possible to further filter the Web site if it is really too large.

We will also investigate how ontologies may be used to characterise our clusters. Galois algorithm generates clusters which have a semantic value without expressing this semantics explicitly. Ontologies may help us make this information explicit.

Bibliography


**Biography**

Benedicte Le Grand
Benedicte Le Grand was born in 1975. She received her engineer diploma from the Institut National des Telecommunications in 1997 and is currently a PhD student at LIP6 (Laboratoire d'Informatique de Paris 6) in the Network Department. Her research deals with Virtual Reality and its use for complex systems visualization. She is interested in markup technologies and especially XML; she was a speaker at ACM CIKM'99 (Conference on Information and Knowledge Management) in Kansas City and at Markup Technologies'99 in Philadelphia.

Michel Soto
Laboratoire d'Informatique de Paris 6, Universite Pierre et Marie Curie
Paris
France
Email: Michel.Soto@lip6.fr

Michel Soto is an associate professor at the LIP6 Laboratory (Laboratoire d'Informatique de Paris 6), University Pierre et Marie Curie, Paris, France. His main research interests are currently virtual worlds interoperability and complex systems visualization.

David Dodds
Blue Shield
San Francisco
U.S.A.
Email: drdodds42@hotmail.com

David Dodds has worked with computers since 1968, when he wrote Continuous Systems Models at university. Later he developed a program to operate a multichannel analog speech synthesizer, which produced recognizable Coast Salish, an (West Coast) Indian language. He also wrote neural network simulations in 1972. After playing with programs to model human personality types and writing code to simulate a town council in a Forrester "Limits to Growth" type model, and teaching computing science at university for a while, he joined Nortel (Northern Telecom) where he wrote text understanding software, and expert systems in C and Prolog. He has been working the last few years on the various emerging XML technologies, was on the W3C SVG workgroup to develop the specification for SVG 1.0; and on the Idealliance committee to develop XML Topic Map specification. David has published numerous papers in robotics and in fuzzy systems. He works on systems of representation, designing UKL (Unified Knowledge Language), and pursues designs of systems of meta data and meta-programming representation. Most recently he was co-author of the book WROX Professional XML Meta Data.