Topic Maps for Context Management

Ruaidhrí Power
Ruaidhri.Power@cs.tcd.ie

No Institute Given

Abstract. The advent of context-aware computing requires a new paradigm in information management. Context information in a computing environment will take the form of disparate and often heterogenous information objects distributed about the network. This paper contends that topic maps are a useful tool to provide a high-level framework for the management of the interconnection of these information objects. This can be achieved by using a 'dynamic' topic map to overlay a management structure on the information, resulting in an extremely powerful way of representing knowledge in a context-aware environment. Individual topic maps can be combined to provide a global view of the context information space, and this paper explores this idea to provide inter-smart space management. Certain extensions to topic maps are also proposed, to provide useful services for context systems. An example of such a service is an 'aggregation service' which would take information objects from a number of sensors and derive new context information from them.

1 Introduction

One of the realities that must be faced in context management is that there will not be one single globally standard model for representing context information. Context data will come in many different forms, from many different sources. Any attempt to formally structure all potential context information would be difficult at best in a controlled situation, within one organisation for example, but almost impossible in an inter-smart space scenario. What is needed is a way to dynamically combine data from these different sources into context information so that it can be queried at runtime.

These data sources will consist of sensors placed in the smart space environment and user devices such as PDAs, workstations and tablet PCs, which can be combined with more traditional sources of information such as user information stored in databases. However each source will only provide us with part of the picture of context that we are seeking, so we must combine these pieces of information into an understandable whole.

A Context Information System (CIS) needs a way of representing the data that can be retrieved from these individual sources in a way that can be processed and reasoned about by a computer. We will term this representation metadata. We can encode knowledge about the relationship between information sources by adding connections representing the relationships between the metadata objects within the CIS. In a smart space scenario, example sources of context information could include:

- Sensor readings
- Device status readings for PDAs, desktop machines, etc.
- User information repositories
- User feedback information

This knowledge will give the CIS an overall view of context data being provided to the system, and it will then be possible to use this knowledge both internally within the CIS and also to provide views upon this knowledge to interested context-aware clients.

Topic maps are one candidate for the task of information encoding, other candidates include semantic networks and conceptual graphs. These methods are all similar in that they attempt to connect pieces of data into a graph which represents the relationships between them. These methods do not attempt to enforce a rigid structure on the information they describe but rather they provide a lightweight way of navigating the information which exists in separately maintained data sources.

Another potential application of these technologies would be to facilitate translating the knowledge provided by one source into a form that can be used elsewhere. For example, one of the goals of the Semantic Web is to make it possible to exchange information by making the meaning of the information available, as well as the information itself. In the Semantic Web, it should be possible for programs to share and process information even when the programs and their information models have been designed completely independently. This concept is interesting in a smart space scenario, where the form of the information cannot be completely defined a priori, particularly when availing of context information from sources external to one's organisation, for example when roaming.

This work explores the use of topic maps as a possible means to structure context information and access to the data sources. What we attempt to accomplish is to use the topic map as a means to encode knowledge about the smart space system, and overlay this knowledge as a dynamic topic map over the information objects. It may also be possible to use topic maps or other Semantic Web constructs to facilitate interoperability between sources of context information. This interoperability would be useful in a smart space system, both within individual smart spaces and for users roaming between different smart spaces.

2 Background

2.1 Context

The idea of context management is fundamentally to give computers more information about the world in which we live, in order to enhance their participation in our daily lives. By supplying context information to applications, the user experience can be enhanced and new applications can be produced. There are many different interpretations of what context information is, such as:

> "[Context information is] any information that can be used to characterise the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between the user and an application, including the user and application themselves."[2]

> "Context-aware computing is the use of environmental characteristics such as the user's location, time, identity and activity to inform the computing device so that it may provide information to the user that is relevant to the current context."[3]

Modern-day programs make very little use of any context information. Most programs are limited to the explicit input that they receive from the environment, usually in the form of user interaction. This frequently leads to differences between what the user wishes to accomplish and what the computer tries to do, because the computer has not been given sufficient relevant information. This is counterproductive, as computers are very capable of assisting users in performing tasks. The field of context management attempts to extract this required information from the computing environment and supply it to context-aware programs in order to enhance the user experience.

One of the most vital areas in smart space research is in collaboration and management between and within different smart spaces. This involves the gathering, interpretation, storage and dissemination of context informaton dynamically and in real time. The organisation of context information, otherwise called context management, should allow for context-aware services in smart-space environments. This should entail seamless accessibility to context-aware services by context-aware applications as the entity, be it a person or a device, moves between different smart spaces.

The information that can compose context is very broad, and can come from a variety of sources. A user's name, age, address, native language, current location and learning style could compose part of his context. Similarly, the people sharing a room with him or working in his office could be considered to be part of this context information, as could the current temperature and lighting conditions. Because almost any information could be considered context information from the point of view of some entity in a smart space, there is very little information that we can discard as being irrelevant. Perhaps the most important characteristic of context information is that we cannot be entirely certain what information will be relevant in advance of constructing a system to manage this information. Therefore, such a system will have to be generic enough to handle new forms of context that will present themselves during its lifetime, and updating to handle this context information should be as painless as possible.

The challenge is to take all this information from disparate and heterogenous sources and to construct a useful computer representation of it that can be queried by context-aware applications. Once this representation has been constructed, it must also be kept up to date with changes within the environment. If this representation is kept consistent and current, it will be possible to extract information which could be used in a variety of modern-day applications, as well as making new types of applications possible.

Context information will almost certainly be managed at the level of an organisation, such as a business or even a simple home user, rather than being managed globally. This is mainly due to the sheer volume of data that will compose context, and also privacy and security concerns that will prompt people to manage their own context information. The set of information that will compose this context is so dynamic that it will never be standardised, so means will have to be developed to promote interoperability between context systems. This is particularly the case for roaming applications, where context information must be supplied and received for a roaming user or device to avail of services within another smart space.

It must therefore be possible to interchange this context information between different context systems, while preserving its meaning. This task will involve automatically translating between different representations of context, possibly through the use of Semantic Web technologies.

2.2 Semantic Web

The Semantic Web[6][7] attempts to provide information about the semantics of data on the World Wide Web, giving data a well-defined meaning so that it can be parsed and understood by computers. Semantic Web technologies allow data on the Web to be defined and linked in a way so that it can be used for more effective discovery, automation, integration and reuse among applications. Adding this 'semantic layer' to the Web would allow much more intelligent linking between different sites, intelligent searching rather than plain text searches, and generally enable much better cooperation between users and computers. Many new applications could also be produced that would take advantage of this semantic layer and process the information it provides. According to Tim Berners-Lee from the W3C[4]:

> "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."

Today, much information is available via the World Wide Web. We have moved from creating general interest sites to creating sites that let us shop, book flights, and do banking online. These services are presented in the same human-readable way as other Web sites are, so they can be easily understood by a human. However the relevant information for providing the service is presented along with much irrelevant information such as font information, graphics, advertisements and many other artifacts which are indistinguishable to a computer from the content itself. One of the aims of the Semantic Web is to allow sites to expose their information in a form that can be easily parsed by a computer application. There is already much demand for a system such as this, as seen in the rise in 'web scraper' applications that attempt to extract this information from today's websites.

Another aim of the Semantic Web is to allow people to produce files which explain to a machine the relationship between different sets of data. Much like a database schema, it will be possible to link different fields from different data sources with a semantic link that will indicate that they mean the same thing. This will allow machines to follow links to facilitate the integration of data from many different sources. Previously, such integration happened at the design time of the application. Semantic Web technologies will allow this integration to happen dynamically at runtime.

2.3 Topic Maps

Topic maps[5] are a knowledge representation technology which is being used to implement the concept of the Semantic Web. Although topic maps existed before the Web, XML topic maps were designed to apply the topic map paradigm to the Web.

Topic Maps provide a common framework for managing interconnected sets of information objects. These information objects are the 'subjects' which are represented as 'topics' within the topic map. A subject can be pretty much anything that you can think or reason about: a chair, a webpage, a person, the sun, the concept of religion, or your favourite television character. A topic is created within the topic map to 'indicate' (or refer to) this subject. Information is then added to these topics by creating associations between them, thereby encoding additional knowledge. For example, the fact that a person has a webpage could be described by an association 'hasWebpage' between the topics representing the person and their webpage. The fact that a whale is a mammal is represented by an association 'isMammal' between the topics representing whales and mammals. The fact that a whale lives in the sea is described by an association 'livesIn', between the whale and sea topics; note that there is no restriction on the number of associations between topics.

Since topics and associations can represent almost anything we can think of, topic maps are a means of representing a virtually unlimited number of relationship types between a virtually unlimited number of information types. And when we are given these topic maps, we can ask interesting questions of them such as “What animals live in the sea?” or “Who is this webpage about?” and retrieve meaningful answers by following the associations between topics.

Another feature of topic maps are occurrences, which are information resources relevant to a topic. A topic representing a person could have occurrences such as a portrait, a CV or some descriptive text. These resources are linked to from the topic itself, meaning that anyone with a reference to the topic representing that person would also automatically have access to these occurrences which would give more information. Occurrence types specify which sort of information the occurrence links to, giving any processing application the option of filtering based on occurrence type.

Each topic can also be given a base name, which is a natural language name of the topic. Base names give a human-readable representation of a topic, used for display purposes and also in topic map merging. Base names also assist in making topic maps navigable, by providing us with a description of what the topic refers to. Base names can be used by users as an initial starting point for locating topics within the map.

A subject indicator is a resource that is intended by the topic map author to provide a positive, unambiguous indication of the identity of a subject. A published subject is any subject for which a subject indicator has been made available for public use and is accessible via a URI, for the purposes of knowledge interchange and mergeability. Topic maps improve navigation on the Web through a mechanism that uses these shared resources. Topic map authors can use these published subject indicators (PSIs) to refer to topics within other topic maps, thereby leveraging the work of other authors.

For instance, the ISO have standardised a description of the countries of the world based on their two- and three-letter country codes. When a topic map author wishes to refer to a particular country, he can refer to that country’s topic within the ISO’s topic map to unambiguously indicate what he is referring to. There are projects underway to define collections of published subjects for people to refer to in their topic maps, which will enhance their mergeability at a later stage.

Two topic maps can be merged if they both have topics that refer to the same subject. When they are merged, a new topic is created with the union of the characteristics (names, associations and occurrences) of the two originals. The new topic map which is created will still contain all the information that was in these originals. Topic map merging works best when the topic maps generally refer to the same subjects, producing a valuable new set of information.

3 Design

The aim of this approach is to add value to context data through the use of topic maps. We will attempt to use topic maps to:

- Model smart space users, objects and concepts as topics within a topic map, along with the relationships between these topics
- Provide access to raw context data through topic maps, by using them as a means of navigating the unstructured information sources
- Investigate the topic map as a place to store the meaning of context information, rather than having that meaning implicit in the structure of the information objects themselves

– Query the topic map to extract knowledge that has been obtained and refined from lower-level context data, and make this knowledge available to context-aware applications

It is not difficult to see that information from sensors, devices, etc. in a smart space environment could be made available in some lightweight form such as a HTTP-accessible document. This document could describe the state of the device; its battery life, current orientation, time since last maintenance, etc. A sensor could produce a document that would describe its own status; the current temperature, the people present in the room with the sensor, or the state of a lightswitch.

The different factors reported in these documents are fairly static in nature for a single device, and relatively easy for the device to publish to interested parties. The difficult task in context management is to represent this information in a form in which it can be profitably used by a smart space system, and also be exported so that it can be used by other smart spaces. Other challenges include dealing with changes in this information; should the context system poll the information source repeatedly or can information be pushed whenever it changes or on a periodic basis?

What topic maps can provide is an overlying structure to that raw information. For instance, associations can be made with the topics representing the device's owner or maintainer, with the room or other smart space that the device is currently in, available services, etc. As topic map associations are inherently bidirectional, providing the device with a connection to its owner also provides a reciprocal link between the owner and the device. In this way, we can ask questions of the topic map such as "Who is the owner of this device?", "Who is present in this room?" or "Which of my devices have not been serviced in the last six months?"

We must consider the distinction between using the topic map purely as a map to reference existing information in data sources and using it to encode information within the topic map itself. It is an open question as to whether we should keep most information in the topic map, thereby keeping the data sources as lightweight as possible, or store most of the knowledge in the data sources, simply using the topic map as a means of easily referencing that data.

If unstructured context information (such as that provided by a sensor) is made available to topic mapping software, changes in the data reported by the sensor as an input to the context system can be reflected by a change in the topic map. In the example of a person moving from one room to another, the fact that a sensor detects the person in a different room would cause the context system to automatically associate the topics representing the person and the room they moved into, updating the topic map and breaking the old association.

The information objects managed by topic maps can be shared with other topic map producers by exporting topics as published subjects. For example, in designing a topic map to represent my organisation, I will no doubt want to model the concepts of a user and a device. Using topic maps, I can publish references to these concepts for public consumption. If another topic map producer (in a collaborating institution for example) refers to those published subjects as the definitions of a user and a device in the production of his or her topic map, the two maps can be automatically merged at runtime.

The two maps could also be merged if some of their topics have the same base name in the same scope, which will often achieve what is desired. Some manual merging of topic maps may be necessary in particular circumstances, however once this process is complete the topic maps will be fully interoperable.

Consider then the example of a user from the collaborating institution roaming into my organisation's smart space. A reference to the user's topic is given to the smart space, and because our concepts of what users and devices are have been merged, I can query the topic map provided to me and ask the same questions I could ask of my local topic map, such as "What devices is this user carrying?" and be provided with subject indicators representing the user's devices.

3.1 Additions to Topic Maps

Topic maps alone do not have a solution for all the challenges that are present in smart space engineering. We must also address additions to the topic map paradigm that must be made in order for a Context Information System to be able to provide additional features.

One of these features is the aggregation of low level data into higher-level context. For example, low-level data from information sources such as sensors might give the location of a person to an accuracy of a few feet. However, this information might be too fine-grained and quickly changing to be useful to a consumer of context. They might instead prefer to only see a person's location to the level of which room or building they are in. We therefore see a need to create topics within a topic map which are aggregations of lower-level context. In this example there might be a topic created that would represent the user's location to the accuracy of which building they are in. A Context Information System would then form the information that topic represents by aggregating information from the topics representing the lower-level sensors. This is not a facility inherent in topic maps, and so must be added by the CIS. As well as aggregation these associations might perform other computation to produce more useful context.

Security and privacy concerns must also be addressed. It will not be sufficient to simply allow all context information managed by a CIS to be accessible by any consumer of context, be it an application or another smart space. Access control mechanisms must decide what context information to export and to which clients. These access control mechanisms should allow as much information to be passed to the clients as possible for them to accomplish their task without compromising sensitive information. This balance will not be easy to strike, and maintaining these access control rules will be non-trivial. There may however be scope for automated maintenance of these rules based on policy specifications.

The final (and perhaps the most interesting) addition to topic maps that we forsee would be the concept of adding a degree of confidence to associations. This could be used for example to slowly expire context information that isn't being refreshed over time. For example if a temperature sensor stopped giving readings it might be appropriate to decrease the confidence associated with the last reading it gave over time until the confidence approached zero.

Considering all these factors will hopefully lead to a very full-featured Context Information System, that will be able to cope with almost any form of context that it's presented with.

4 Implementation

As an initial attempt at implementing a context information system, we are building an application which will store a model of example context information. Particularly, it will allow topic maps from other providers to be imported into the local topic map so that the other provider's context information can be used locally, and also allow information stored in the local context information system to be exported to other interested parties.

We wish to demonstrate the natural way in which topic maps from different providers can be merged, even though they were developed separately and perhaps for different purposes. However, it should nevertheless be possible to merge them to some degree and to extract knowledge from them. The extent to which the maps can be merged will depend on how closely the subjects they describe match up. Even if the topic maps cannot be fully merged however, it should still be possible to extract information through pointers given by the foreign topic map.

We present parts of the topic map (expressed as an XML topic map) in order to explain the structure of a simple example map. The first thing we present are some of the concepts that this example will model: a person and a webpage.

```
<topic id="person">
  <baseName>
    <baseNameString>Person</baseNameString>
  </baseName>
</topic>

<topic id="webpage">
  <baseName>
    <baseNameString>Web page</baseNameString>
  </baseName>
</topic>
```

These two topics simply identify ideas that we wish to model and refer to within the topic map, and give them human-understandable names. The next topic represents a particluar user within the context system, and says things about that user.

```
<topic id="ruaidhri">
  <instanceOf>
    <topicRef xlink:href="#person"/>
  </instanceOf>
  <baseName>
    <baseNameString>Ruaidhri Power</baseNameString>
  </baseName>
  <occurrence>
    <instanceOf>
      <topicRef xlink:href="#webpage"/>
    </instanceOf>
    <resourceRef xlink:href="http://www.cs.tcd.ie/Ruaidhri.Power/"/>
  </occurrence>
</topic>
```

Converted into english, this topic describes certain facts. Firstly, that this topic is an instance of the person topic. It also give a baseNameString to the topic, for display and searching purposes. Finally it describes a particular occurrence of this topic, in the form of a webpage to which it gives an occurrence URL. Multiple webpages could be specified, as could different occurrence types.

```
<topic id="device">
  <baseName>
    <baseNameString>Device</baseNameString>
  </baseName>
</topic>

<topic id="myPhone">
  <instanceOf>
    <topicRef xlink:href="#device"/>
  </instanceOf>
  <baseName>
    <baseNameString>My Siemens S35i phone</baseNameString>
  </baseName>
</topic>
```

These two topics represent the concept of a device and a particular device respectively. The second topic is an instance of the device topic. Our next task could be to associate a particular person with a particular device (a mobile phone, in this example), saying that that person is carrying the device. In order to do this, we need to represent the concept of carrying something, and make an association between the person in question and their device.

```
<topic id="carrying">
  <baseName>
    <baseNameString>carrying</baseNameString>
  </baseName>
</topic>

<association id="ruaidhri-carrying-myPhone">
  <instanceOf>
    <topicRef xlink:href="#carrying"/>
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#person"/>
    </roleSpec>
    <topicRef xlink:href="#ruaidhri"/>
  </member>
  <member>
    <roleSpec>
      <topicRef xlink:href="#device"/>
    </roleSpec>
    <topicRef xlink:href="#myPhone"/>
  </member>
</association>
```

This association records the fact that a particular person is carrying a particular device. More precisely, it says that there exists an association of type ‘carrying’ between the topic ‘ruaidhri’ who is playing the role of a ‘person’, and the topic ‘myPhone’ which is playing the role of a ‘device’.

Because the structure of topic maps and topic map markup have been standardised, all that is required for a Context Information System to import foreign context information is an agreement on the terms in question. For example, a foreign context system might use the term ‘customer’ where our system refers to a ‘user’. Similarly, the foreign context system might have a different concept of ‘carrying’ something, for example just being close to the device might be good enough for a particular application. These differences can be overcome if the two context systems agree that their different topics represent the same subject. Once this is done, the topic maps can be merged and the resulting context information used from there.

The program developed will take different forms of context information (represented as topic maps) from different providers and merge them. This will demonstrate the usefulness of topic maps for this purpose. Example topic maps will be developed that represent particular smart space scenarios, to form a design for these topic maps that would be useful to a Context Information System. Concepts such as ownership, presence, mobility and responsibility will be modelled within these topics. We will particularly search for scenarios that do not lend themselves to the topic map paradigm, since even though topic maps are a generic knowledge representation mechanism there may be cases for which they aren't suited.

In our example, the topic map presented above could be merged with a mobile phone provider's context system in order to give the provider information about which mobile device phone calls should be dynamically routed to. In this case simply updating one's local topic map with the context information about which phone is being carried would allow the provider to merge that information with its own, and thereby update its actions based on the context information.

This process will also assist in constructing an overall architecture, by helping to address questions about how topic maps should be structured. Should a CIS manage individual topic maps for each information source, or one overall topic map that encodes all its knowledge? Merging these maps will be particularly instructive, as this process is likely to expose any flaws that may exist in our approach.

4.1 TM4J - Topic Maps for Java

The platform being used to manipulate these example topic maps is TM4J, which is a topic map processing engine written in Java. TM4J is quite an active project with much interest, and reasonably good documentation. From the TM4J website[1]:

> “TM4J is an open-source package for creating topic map processing applications. The core of TM4J is a set of Java APIs for parsing, manipulating and saving topic maps.”

Features of the TM4J engine include:

- An object model which supports all of the constructs of the XTM 1.0 specification
- A set of simple command line tools for manipulating and measuring topic maps
- A command-line utility for merging two or more topic maps automatically
- Ability to persist topic map data in an object-oriented or relational database
- Import of XTM and Ontopia's LTM (Linear Topic Map) interchange formats
- Export of XTM interchange format topic maps
- An implementation of the tolog topic map query language
- An implementation of the TMAPI standard topic map programming API
- Open-source under the Apache License

5 Research direction

Our aim is to investigate the approach of using topic maps to model context information for processing by a Context Information System (CIS). We will attempt to address several research questions:

- How suitable are topic maps for modelling context information? Their background as a generic knowledge representation platform leads us to believe they are worthwhile in the field of context, but our experimentation may find cases for which they aren't suitable.
- How large would an overall representation of context become when combined from different sources? Could this way of modelling be infeasible given modern processing capabilities?
- How do updates in remote context get propagated to local context? Are they pushed periodically from the remote end, or are they requested as needed?
- How difficult is it to merge different pieces of context information together? How much of it can be done automatically, and how much must be done by hand?

If this experimentation suggests that topic maps are a suitable mechanism to proceed with, we will attempt to build a generic framework for context information using topic maps that will take data from different information objects and combine them into an overall view of context. It is however quite possible that a hybrid approach between topic maps and some other knowledge representation mechanism will be necessary.

6 Conclusion

Topic maps provide a useful high-level framework for the management of the interconnection of information objects such as those found in a smart space environment. They are particularly useful because they offer a generic and portable method of encoding knowledge within a smart space system, thereby making themselves a candidate for assisting in inter-smart space management.

Additions will need to be made to the topic map paradigm in order to make it applicable to the highly dynamic nature of smart space environments, however topic maps seem to provide a good initial starting point for knowledge representation in these environments.

References

1. Topic Maps for Java website – `http://tm4j.org/`
2. Gregory D. Abowd, Anind K. Dey, Robert Orr and Jason A. Brotherton 1997, *Context-awareness in Wearable and Ubiquitous Computing*, in ISWC, pages 179-180.
3. Jenna Burrell and Geri K. Gay, *Collectively defining context in a mobile, networked computing environment*, Short talk summary in CHI 2001 Extended abstracts, May 2001
4. Tim Berners-Lee, James Hendler and Ora Lassila, *The Semantic Web*, Scientific American, May 2001
5. TopicMap.com website – `http://www.topicmap.com/`
6. Tim Berners-Lee and Eric Miller, *The Semantic Web Lifts Off*, ERCIM News No. 51, October 2002
7. The W3C's Semantic Web activities – `http://www.w3.org/2001/sw/`