Charting of Unknown Territory:
Application of Topic Maps to Chan-Buddhist Chronicles

Christian Wittern

Kyoto University, Japan

Contents

1. Introduction

Over the last few years, a large number of texts and other resources have been encoded using markup according to the TEI guidelines[1] and other document type definitions. While the TEI provides both a comprehensive theoretical framework for text encoding and a practical guide to its appliance to individual texts or even text groups, there is little to find that goes beyond the encoding of individual documents and gives advice at how to proceed with an analytic framework that might help encode not just concrete features of a text, but also aspects of the world as presented through the encoded texts.

Although it could be argued, that the TEI provides for such a facility through its `Feature Structure' mechanism [2], this does not hold through in practice, because it lacks the support other competing solutions to this problems offer.

With the growing number of encoded material, the absence of more abstract ways of encoding what a text talks about becomes all the more felt. There is so much interesting use these texts could be put to, but so little to work with.

2. Topic Maps

One of the attempts of providing a path to a solution to this problem, or maybe more accurate, a way to approach this problem, has been the work that culminated in the ISO standard 13250, SGML Topic Maps[3]. This standard provides a model and architecture for the semantic structuring of information networks. It has the potential to provide a bridge between the information as contained in texts encoded with schemes like the TEI, other information resources and information about the world in general.

The topic maps standard describes a model and interchange format for topic maps. A good general introduction to topic maps is available in Steve Pepper, *Euler, Topic Maps, and Revolution*, in: Proceedings of XML Europe 99 Conference, GCA, Alexandria, VA, 1999; Steve Pepper, *Navigating Haystacks, Discovering Needles*, in: Markup Languages, 1999 and Hans Holger Rath and Steve Pepper, *Topic maps at work*, in: Charles F. Goldfarb and Paul Prescod (eds):XML Handbook, 2nd edition, Prentice Hall 2000, p 618-635..

Since the publication of the ISO standard in January of 2000, work has begun to develop a XML based version of topic maps, which would be more appropriate to be used on the web. This activity, has been coordinated by an ad-hoc organisation Topicmaps.Org, which was created with the explicit purpose of providing a fast-track towards a usable standard. This standard, which was named XTM, or XML Topic Maps, was first published on December 4th, 2000 with some subsequent corrections and changes published on Februrary 17th, 2001. ALthough there has been some infighting recently, the XTM standard can still be considered to be stable and reliable enough in the syntax it describes to base some serious work on it.[4]

A topic map is an abstract notion, which specifies how "things in the world" are encoded in computers. These "things in the world" comprise everything we can think of, existing things as well as things in our fantasy, abstract terms, laws, whatever. In topic map circles, all these "things" are called `subjects'. Topics, the basic building blocks of topic maps, are thought of as `reifying' these subjects, or in other words, topics in topic maps work as proxy for the subjects, the things we want to talk about. Now, not all the things we want to talk about already exist in a computer, therefore, they are not adressable from topic maps. Topic maps can be thought of as an abstract model, that describes a very abstract way of expressing properties of topics, socalled `topic characteristics'. If topic maps are written out (as opposed to being held in the memory of some topic map processing application) with the topic map interchange syntax, they can be written either as a SGML document (the original ISO 13250 Topic Map syntax) or as a XML document following the syntax developed by the Topicmaps.Org working group. In the

following discussion and examples, I will use the syntax of XML Topic Maps, which is more explicit and understandable than the syntax of SGML Topic Maps.

Topics can have any of the following characteristics.

- instanceOf
- names
- occurrences

SGML/XML document in which different element types are used to represent topics, occurrences of topics, and relationships (or `associations') between topics.

3. Application of topic maps to texts of the Chinese Chan-School

In this project, I am trying to use a topic maps mechanism to encode information in a number of 10th to 13th century Chan-chronicles. The texts have already been marked up according to the TEI guidelines. With the TEI markup, the basic structure of the documents and features like names of persons and places, datable events and the like have been marked. Topic maps are now used to encode the following features in the text:

- Topics and occurrences of Chan-masters and other persons, places, datable events
- Quotations and allusions occurring in the text
- Interjections and comments of later Chan-masters relating to a given anecdote
- Information about the lineage of Chan-masters
- Other external information related to these topics
- Links to other resources relevant to topics occurring in the texts

In addition, a typology of anecdotes and their instances is in preparation. This typology will also be exposed as a topic map. This will allow to more easily trace and visualize the development and relationship of *topoi* in the anecdotes.

Although this is not a technical paper, I would like to demonstrate how this could be encoded with the syntax developped by the XTM working group.

4. Some examples

To illustrate how topic maps can be used to encode information such as those listed above, here are some examples for the items mentioned above.

4.1. Topics and occurrences of Chan masters and other persons, places, datable events

Topic maps can make use of any arbitrarily defined mechanism to associate topics to each other. To enhance inoperability, there are some types of association built into the topic map standard, most importantly, the class-instance and the class-subclass relationship. Since this is a so common and useful type, there is even a syntactical shortcut for this relationship, the element <instanceOf>. In the case of assigning topics for individual Chan masters, I dediced to use the following hierarchy:

- Person (human being)
- Chan master
- Individual masters

There are of course many other ways to assign this. One other possibility I considered was to make the individual masters instances within the lineages, but later I rejected this way of assigning relations, since this could be better and more flexible expressed using custom associations, rather than the class-instance relationship.

Using the XTM syntax, this relationship is expressed in the following way:

```
<topic id="person">
  <baseName>
    <scope>
      <topicRef xlink:type="simple" xlink:href="#zh-tw"/>
    </scope>
    <baseNameString>人</baseNameString>
  </baseName>
</topic>
<topic id="chan-master">
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#person"/>
  </instanceOf>
```

```
    <baseName>
      <scope>
        <topicRef xlink:type="simple" xlink:href="#zh-tw"/>
      </scope>
      <baseNameString>禪師</baseNameString>
    </baseName>
</topic>
```

As can be seen, the class-instance relationship between these to topics is established through the <topicRef> element in <instanceOf>. Another noteworthy feature of the topic maps syntax that can be seen here is how the language of the <baseNameString> is established through scoping it with the topic that defines that language (this topic is not shown). Everything within a topic map is defined through topics and relationships, which makes the whole paradigm very flexible and powerful.

The individual masters would then of course be instances of the topic with the id `chan-master', as shown in the following example:

```
<topic id="C-05-01-02">
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#chan-master"/>
  </instanceOf>
  <baseName>
    <scope>
      <topicRef xlink:type="simple" xlink:href="#zh-tw"/>
    </scope>
    <baseNameString>馬祖道一</baseNameString>
  </baseName>

    (1.)
  <occurrence>
      <instanceOf>
        <topicRef xlink:type="simple" xlink:href="#alternate-name"/>
      </instanceOf>
    <resourceData>馬祖</resourceData>
  </occurrence>
```

(2.)

```
<occurrence>
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#dates"/>
  </instanceOf>
  <resourceData>709-789</resourceData>
</occurrence>
```

(3.)

```
<occurrence>
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#pupil-of"/>
  </instanceOf>
  <resourceRef xlink:type="simple" xlink:href="#C-05-01-01"/>
</occurrence>
```

(4.)

```
<occurrence>
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#zhuan"/>
  </instanceOf>
  <resourceRef                                    xlink:type="simple"
xlink:href=http://www.cbeta.org/xml/T51/T51n2076.xml#xpointer(//lb@[n="0245c23]/range-
to(//lb/@[n="0246c07"]))'/>
</occurrence>
```

(5.)

```
<occurrence>
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#mentioned"/>
  </instanceOf>
  <resourceRef                                    xlink:type="simple"
xlink:href=http://www.cbeta.org/xml/T51/T51n2076.xml#xpointer(//lb@[n="0240c19"])'/>
</occurrence>
```

```
</topic>
```

This is a long example. At the beginning, there is the usual <instanceOf> element, which now points to the topic `chan-master' as expected. What follows is some information regarding this topic, which is given within the element <occurrence>. Since there are a number of it, I have inserted some numbers in order to reference them, these numbers do not form part of the original topic map.

1. The first occurrence gives an alternate name for this master. As can be seen, the type is identified by making the occurrence an instance of a topic with the ID `alternate-name'. The content of this occurrence is contained within the occurrence, identified by the element <resourceData>.

2. Next the dates are given, nothing new here.

3. This occurrence identifies the place of this master in the line, by given his master, identified through a reference to the topic `#pupil-of'.

4. Finally, with this occurrence we have a reference to an entry in one of the lamp-histories about this master. Instead of giving the resource data directly or linking to a topic as in earlier examples, the link goes directly to the location in a text. In this case, the xpointer is used to define a range in the text. There are of course other possibilities, for example identifying an element and its content by using an ID code, or a XPath expression. The document itself is here directly identified with a URL.

5. The only difference of this occurrence to the previous one is that in this case, the master is only mentioned, not the main subject of the entry. Also, only a point in the text, not a section is identified.[5]

4.2. Quotations and allusions occurring in the text

Since the basic techniques have been demonstrated, this will be brief.

```
<topic id="T14n0475quot">
  <instanceOf>
    <topicRef xlink:type="simple" xlink:href="#quotations"/>
  </instanceOf>
  <baseName>
    <scope>
      <topicRef xlink:type="simple" xlink:href="#zh-tw"/>
```

```
    </scope>
    <baseNameString>維摩詰所說經</baseNameString>
  </baseName>
  <occurrence>
    <instanceOf>
      <topicRef xlink:type="simple" xlink:href="#direct-quotation"/>
    </instanceOf>
    <resourceRef                                                    xlink:type="simple"
xlink:href='http://www.cbeta.org/xml/T51/T51n2076.xml#xpointer(//lb[@n="0448c09"]/range-
to(//lb[@n="0448c09"]))'/>
  </occurrence>
</topic>
```

 I have here encoded the quotation from the perspective of the quoted text, there are of course other possibilities and practice has to show, what proves to be better in the long run.

4.3. Interjections and comments of later Chan-masters relating to a given anecdote

```
<association id="T51n2076p0446c08-C-05-07-32-comments">
  <member>
    <roleSpec>
      <topicRef xlink:type="simple" xlink:href="#commentator"/>
    </roleSpec>
    <resourceRef xlink:type="simple" xlink:href="#C05-07-32"/>
  </member>
  <member>
    <roleSpec>
      <topicRef xlink:type="simple" xlink:href="#comment-content"/>
    </roleSpec>
    <resourceRef                                                    xlink:type="simple"
xlink:href='http://www.cbeta.org/xml/T51/T51n2076.xml#xpointer(//lb[@n="0446c08"])'/>
  </member>
  <member>
    <roleSpec>
      <topicRef xlink:type="simple" xlink:href="#comment-target"/>
```

```
    </roleSpec>
    <resourceRef                                        xlink:type="simple"
xlink:href='http://www.cbeta.org/xml/T51/T51n2076.xml#xpointer(//lb[@n="0446c04"]/range-
to(//lb[@n="0446c08"]))'/>
  </member>
</association>
```

In this example, an <association> element is used to establish a connection between

- a commentator
- the content of the comment
- the target of this comment.

In the topicmap paradigm, all relationships are expressed through the <association> element. The members taking part in this relationship are expressed as content of the <member> element, whereas the role they play are encoded in the <roleSpec> element.

5. Preliminary results

As can be seen, the research reported in this paper tries to go beyond the encoding of individual texts and tries to overlay a layer of abstraction on it, that will allow the exploitation of this layer, as well as any external information attached through it, when analyzing these texts.

There are however some problems still not solved satisfactorally. One of the most important one is that it is not easy and straightforward to represent time-spans in topic maps. Since topic maps work by identifying there subjects in a punctuell form, some conventions are needed to make the encoding of time-spans useful accross different applications.

Although the technology surrounding the use of topic maps is still in its infancy, it is expected that there soon will be browsers of topic maps, that allow accessing the encoded material directly through the maps. This will allow researchers to formulate and explore questions concerning the material in a way that is much closer to the needs of a researcher than is the case with current information retrieval technology. Possible questions with meaningful answers will include the following:

- Can the development of *topoi* in these texts seen in the context of lineage affiliation?
- Are there anecdotes with similar content attributed to masters in different lines?
- Is the lineage affiliation interrelated strongly with the area of origin of a master?

- Are lineages organized in geographic patterns?

Depending on the availability of sufficiently fine-grained data in the texts, and of course bearing in mind that the picture presented in the text is not necessary of historical accuracy, some of the following questions could possible also be traced in such a topic map browser:

- Which masters have been at a given temple at a given time?
- Could two masters (who are reported to have met) actually have met? If yes, where?
- What other masters did a given master encounter? At which time in his life?

There are of course an unlimited number of questions that could be posed and, hopefully, meaningfully answered with a browser for topic maps.

Notes

1. The Association for Computers and the Humanities (ACH), The Association for Computational Linguistics (ACL) and The Association for Literary and Linguistic Computing (ALLC) *Guidelines for Electronic Text Encoding and Interchange*, edited by C. M. Sperberg-McQueen and Lou Burnard, TEI P3 Text Encoding Initiative Chicago, Oxford, May 16, 1994.

2. See Chapter 16, p.387ff of the TEI Guidelines.

3. International Organization for Standardization, *ISO/IEC 13250, Information technology - SGML Applications - Topic Maps* Geneva, 2000

4. More information about the XML Topic Maps (XTM) is available at www.topicmaps.org (http://www.topicmaps.org) .

5. The XPointer notation in these examples uses only the value of the attribute "n" in the <lb> element to identify the location in the text. In terms of XPointer, this is not the most efficient and best way, it is here used since it offers the best backward compatibility to pointers from the pre-electronic area, where references where given by the line number. It also is most likely portable accross different XML files of the same text, although those would have to be identified by a different URL anyway. It is of course possible to make a automatic conversion to the document path or other more efficient ways of pointing to the desired points in the text.

As given above, the XPointer identifies only lines, although it should ideally point to individual characters as starting and ending points. Although the XPointer syntax

supports such constructs, [it could be coded as xpointer(string-range(//lb@[n="0240c19"],2,4)) for a range from the second to the fourth character on line 19, page 240c] I do not use it here, because there is a problem whether or not to count the interpunction. I would prefer to ignore the punctuation, but as far as I see, XPointer does not allow this.