XTM 1.1

Clearing up the last issues

Status

- **We expect to finish SAM at this meeting**
- **We have proposals for most of the open XTM issues**
- **With luck we may be able to finalize XTM at this meeting**
- **Text currently not in accordance with ISO style**
  - should probably rewrite, then submit as Committee Draft
- **Relevant documents**
  - N0396: SAM
  - N0398: Latest XTM draft
  - N0399: Proposed resolutions of open issues

Plan for this session

- **Do the issues one by one**
  - first introduce the issue with background
  - then present proposed resolution from the authors
  - committee discussion
  - finally: resolve issue or put it aside to return to it later
- **Note that the resolutions we propose are for the most part edited into the text of N0398 already**
- **We have two hours, so we should aim for efficiency...**

The XSDL/RELAX-NG appendices

- **N0398 contains two new appendices**
  - one with an XSDL schema for the XTM syntax
  - another with a RELAX-NG schema for the same
- **These were added because**
  - it seemed to be a political requirement
  - some users needed such schemas and it was considered that having standardized ones would be good for all concerned
  - as will be seen it solves some of the problems with XML
- **Any comments or reactions on this?**
  - is it in any way controversial?

xtm-href-xpointer

- ***“Does XTM allow full XPointer references, or only bare names?”***
- **That is, can I do the following?**
  - <topicRef xlink:href='#xpointer(id(“tosca”))'/>
  - <subjectIndicatorRef
    xlink:href='#xpointer(id(“tosca”) / baseName[1])'/>
- **Pros:**
  - allows addressing of elements which don't have IDs
- **Cons:**
  - requires processors to build a complete tree of the XTM document
  - considerably increases implementation complexity
- **Recommended resolution:**
  - only allow bare names; disallow use of xpointer in <topicRef> and <mergeMap>; allow it elsewhere

xtm-where-resourceref

- ***"Should resourceRef be allowed in roleSpec, instanceOf and parameters?"***
- **Considerations:**
  - it is already allowed in scope, so why not in parameters?
  - it is possible to use topics representing resources as classes and association role types despite the restriction in the DTD
  - adding a constraint adds implementation complexity
  - can we be sure that this will never ever under any circumstances be correct?
- **Proposal:**
  - allow it in all three cases

xtm-unused-ids

- ***"Should we retain the id attribute on element types which do not represent something that can be reified?"***
- **Considerations:**
  - why have ID on topicRef when it cannot be reified in the model?
  - removing them would not be backwards compatible
  - they may have legitimate uses
- **Proposal:**
  - keep them

xtm-schema-uri

- ***"Should the XSDL and RELAX-NG schemas declare xlink:href attributes to be of type URI?"***
- **Considerations:**
  - it does not actually improve validation
  - it does improve the use of the schemas as documentation
  - it may help smart XML editors
  - it seems like the right thing to do
- **Proposal:**
  - do it!

xtm-xlink-actuate

- ***"Should all or some element types which have xlink:href attributes also be given an xlink:actuate attribute? If so, what should the legal range of values be?"***
- **Considerations:**
  - would allow users to specify whether or not external references should be followed during deserialization
  - goes beyond simple fixes to adding new functionality
  - means taking the use of XLink more seriously
  - is of doubtful actual utility
- **Proposal:**
  - leave out of this version of XTM

xtm-xmlbase-everywhere

- ***"Should the xml:base attribute be allowed on every non-empty element type?"***
- **Considerations:**
  - would allow the use of different base URIs for different parts of the document
  - goes beyond simple fixes to adding new functionality
  - is of doubtful utility
- **Proposal:**
  - don't do it

xtm-variantname-deprecate

- ***"Should the variantName element type be deprecated, and the content model of variant be adjusted to make it redundant?"***
- **Example:**

```
<variant>
  <parameters>
    <topicRef xlink:href="#id10"></topicRef>
  </parameters>
  <variantName>
    <resourceData>via della finestra</resourceData>
  </variantName>
</variant>
```

xtm-variantname-deprecate

- **Considerations:**
  - would simplify the syntax by removing a redundant element type
  - deprecation would be backwards compatible; removal would not be
  - definitely an intrusive change
  - the benefit is small
- **Proposal:**
  - don't do it

xtm-parameters-deprecate

- ***“Should the parameters element type be deprecated and replaced by scope?”***
- **Considerations:**
  - SAM considers this to be scope
  - would simplify the syntax by removing an entire element type
  - content model differs from that of <scope> in 1.0
  - again backwards compatible only if we deprecate
  - benefit is small
- **Proposal:**
  - don't do it

xtm-topicref-notopic

- ***"Is it an error for a topicRef element to refer to a non-existing topic?"***
- **Considerations:**
  - XTM 1.0 says the link "must resolve to a <topic> element ..."
  - several users have wanted warnings about this
  - no implementation currently enforces this
  - validation against a TMCL schema would catch this sort of error
  - it is difficult to implement efficiently when dealing with large topic maps
  - cannot be checked against external topic maps unless they are loaded
- **Proposal:**
  - do not consider it an error

xtm-topicref-fragment

- ***"Is the URI given in the xlink:href attribute (of topicRef elements) required to have a fragment identifier?"***
- **Considerations:**
  - cannot actually refer to a specific element without it
  - XTM 1.0 does not require this
  - is it allowed to refer to a document that consists only of a <topic> element?
  - adds implementation complexity to check for this
  - catches typical user error where the '#' before the ID is forgotten
- **Proposal:**
  - require it

<mergeMap> principles

- **Principles for using <mergeMap>:**
  - it must refer to an XTM document
  - <topicMap> need *not* be the document element of this; it can have other XML markup around the <topicMap> element
  - this means the XTM document may contain more than one <topicMap> element
- **These principles were inherited from XTM 1.0**
  - runs counter to the principle of “interchange only” somewhat, but what's done is done

xtm-mergemap-reference

- ***"Is it an error if a mergeMap element refers to an XML document that contains multiple topicMap elements without providing a disambiguating fragment reference?"***
- **Considerations:**
  - the reference would be ambiguous
  - it could be interpreted to mean the result of merging all the TMs
  - this is not a commonly occurring scenario
- **Proposal:**
  - make it an error

xtm-version

- ***"What version number should we give the updated XTM syntax?"***
- **Considerations:**
  - we have already added typed names
  - we have also (?) relaxed restrictions on resourceRef
  - the version number should give the public a clear idea of the relation between the new and the old XTM
  - the changes all seem to fall in the category of minor cleanups
- **Proposal:**
  - 1.1

The mess that is XML

- **We have some decisions to make regarding XML support:**
  - do we, or do we not require namespace processing?
  - is the namespace URI used for name matching?
  - do we, or do we not require XTM processors to do DTD validation?
- **These decisions have several consequences:**
  - if we use namespace processing DTD validation is out the door
  - if we don't embedding XTM in other syntaxes is harder
  - the DTD actually modifies the document through inclusion of defaulted attribute values!
  - different *conforming* parsers can produce different results
  - the DTD references in the !DOCTYPE declaration tend to cause users grief
  - the DTD validator implementations that exist make it difficult to write an application that avoids the pitfalls
- **We need a strategy for dealing with this!**

Proposed XML strategy

- **Ignore how the infoset is produced altogether**
  - do not require DTD use or validation, do not forbid it
  - warn about interoperability problems
  - ignore how namespace declarations get into the document
- **Require namespace processing**
- **Do not require DTD validity; require RELAX-NG validity**
- **Make the xlink:type attribute optional**

xtm-namespace-support

- ***"The text as specified here requires that XTM processors do XML Namespace processing. Is that acceptable?"***
- **Considerations:**
  - why have a namespace we don't use?
  - are there any arguments against this other than that namespaces suck very badly?
  - it is difficult to be taken seriously if namespaces are ignored
- **Proposal:**
  - require it

xtm-namespace-uri

- ***"Should the version number in the XTM namespace URI be changed?"***
- **Considerations:**
  - if we require namespace processing this means changing all the element type names...
  - processors should be able to detect immediately whether an XTM document is 1.0 or 1.1
- **Proposal:**
  - don't change the namespace URI

xtm-topicmap-version

- ***"Should the topicMap element have a version attribute?"***
- **Considerations:**
  - allows us to achieve the same effect as changing the namespace URI but without the side effects
  - this can reasonably be considered a fix necessitated by the changes we are making
  - the attribute should not affect XTM 1.0 processors in any way
  - we are likely to need this attribute if we ever make XTM 2.0
- **Proposal:**
  - add it

xtm-normative-schema

- ***"Which schema should be the normative schema?"***
- **Considerations:**
  - we need to make *one* of the schemas the normative one
  - DTDs are politically incorrect, as is XSDL
  - RELAX-NG is an ISO standard
  - RELAX-NG has a compact syntax suitable for use in the text
  - DTDs have unfortunate implications for XML processing
  - DTDs can express all the constraints on XTM
  - using RELAX-NG/XSDL would allow us to remove the pointless ordering restrictions in the current DTD
  - RELAX-NG is easy to read/learn; XSDL hard
- **Proposal:**
  - make RELAX-NG the normative schema

xtm-fixed-attributes

- ***“Attributes declared as #FIXED in the DTD can not be guaranteed to always be present in the XML document as parsed, either because there is no DOCTYPE declaration, or because the parser does not read the DTD. ...”***
- **This was already discussed in some detail**
- **Proposal:**
  - stay out of this entirely
  - require namespace processing to work, but stay out of how
  - warn against interoperability problems with DTDs
  - make xlink:type optional, require the value to be 'simple' if given

xtm-unknown-elements

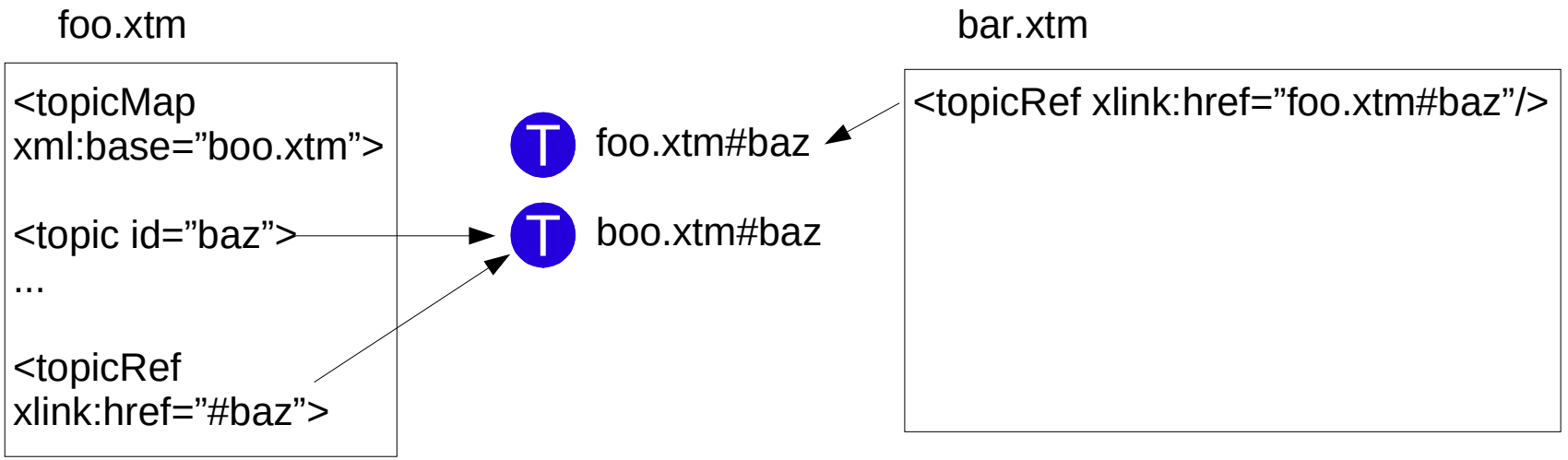
- ***"All unknown elements (regardless of namespace) are ignored in order to ensure forwards compatibility, but this means DTD compliance cannot be required. Which is more important?"***
- **Considerations:**
  - making the schemas support this will be very hard
  - it will also mean that <topciRef> and the like will pass unnoticed
  - the elements ignored may be crucial to the topic map
  - whether to allow markup in <resourceData> is a separate issue
- **Proposal:**
  - unknown elements inside the <topicMap> element should be considered errors

xtm-topicref-notatopic

- ***“Is it an error for a <topicRef> element to refer to an element that is not a <topic> element?”***
- **Considerations:**
  - clearly, this is an error, though it could be interpreted as reification
  - not all IDs are recorded by the in the topic map, which means that detecting this requires the IDs to be stored separately
  - for very large topic maps this will require persistent storage of some sort
  - this is a rather onerous requirement
- **Proposal:**
  - do not make it an error

xtm-same-doc-refs

- ***“RFC 2396, section 4.2, specifies that URI references of the form "" and "#fragment" are resolved relative to the URI of the current entity. Should XTM should follow this?”***
- **This is only relevant when xml:base is set to a URI different from that of the current document**
- **If xml:base affects such URIs we'll get the result shown below**



xtm-same-doc-refs

- **Proposal:**
  - follow RFC 2396 and make such URIs resolve relative to the *document URI* rather than the base URI

xtm-href-whitespace

- ***"Is whitespace allowed in the xlink:href attribute? If it is allowed, how is it interpreted? If it is not allowed, what action is taken when it is found?"***
- **Considerations:**
  - whitespace characters are not allowed in URIs
  - XSDL does not consider whitespace a validation error
  - XLink requires spaces to be escaped, but does not require applications to validate the URIs
  - XTM processors spend a fair amount of time during deserialization on URI processing
- **Proposal:**
  - forbid spaces

xtm-member-id

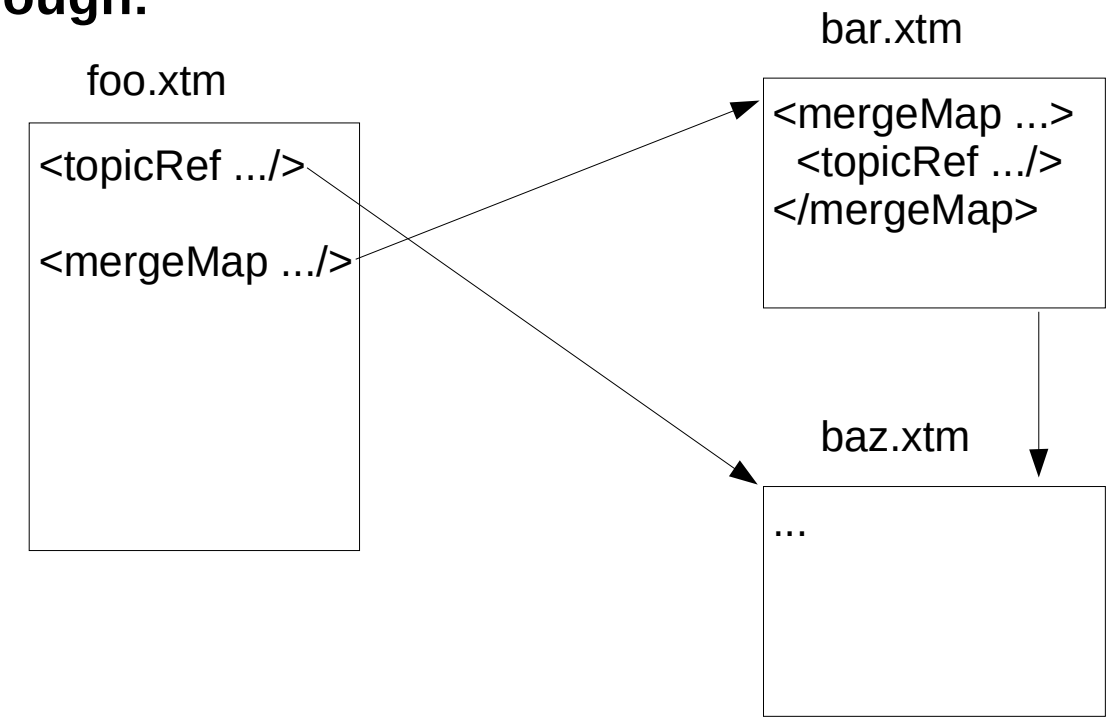
- ***“Should the id attribute of member be ignored?”***
- **Considerations:**
  - if it is ignored it is impossible to reify association role items in XTM
  - if they cannot be reified, why have them at all?
  - how should the id be handled if the member has multiple role players?
  - we don't know what the current practice is

xtm-mergeMap-and-topicRef

- ***"What is the correct behaviour if a topicRef to an external document occurs first, followed by a mergeMap with added themes?"***
- **The problem here is that a naïve implementation will first load the external TM without added themes**
- **It will then encounter the <mergeMap> and discover that it should have added the themes**
- **At this point it is impossible (using only SAM) to go back and apply the themes**
- **Note that it is also possible for <mergeMap> elements to apply different sets of added themes to the same external topic map...**

xtm-mergeMap-and-topicRef

- **As it turns out, the situation can also be more complex**
- **Not resolving external refs before end of document is not enough:**



xtm-resourcedata-markup

- ***“Should we extend the content model of resourceData to allow arbitrary markup within the element, and require implementations to be able to represent this?”***

xtm-subjectidentity-children

- ***"Should multiple resourceRef elements be allowed inside subjectIdentity?"***
- **This issue depends on an unresolved SAM issue and so should await the resolution of that issue**
- **Current content model:**
  - (resourceRef?, (topicRef | subjectIndicatorRef)*)
- **Alternative would be:**
  - (topicRef | resourceRef| subjectIndicatorRef)*