XTM without the angle brackets

Introduction to the Compact Topic Maps Syntax (CTM)

RFI Draft 0.3

```
%prefix m <http://psi.ontopia.net/music/#>
%prefix w <http://en.wikipedia.org/wiki/Tosca>

tosca w:Tosca ; "Tosca" ; ISA m:opera ;
  audio-recording  "7 47175 8" ;
  premiere-date  1900-01-14 ;
  synopsis
    http://www.metopera.org/synopses/tosca.html
      @web, operanews ;
  m:composed-by w:Giacomo_Puccini ;
  premiere teatro-costanzi ;
  takes-place-in roma ~ tosca-takes-place-in ;
  m:libretto-by
    w:Giuseppe_Giacosa ,
    w:Luigi_Illica .
```

Steve Pepper
Convenor SC34/SG3
Coordinator RDFTM Task Force

pepper@ontopia.net

WG3, Montreal, 2006-08-11

CTM document

- **Consists of a header and a body**
- **Header**
  - required and optional declarations
- **Body**
  - assertion blocks
  - verbose associations
- **White space**
  - (mostly) not significant, except inside strings
- **Comments**
  - may occur anywhere white-space is allowed

Comments

- **One line comments**

```
# This is a comment. It ends at the end of the line
```

- **Multiline comments**

```
/*
This is also a comment.
It continues across
multiple lines.
*/
```

- **Issue: Do we need multiline comments?**

Topic identifiers

- **Can be one of the following:**

Name	Form	Example
local ID	ID	Puccini
QName	ID ":" ID	wikipedia:Puccini
subject identifier	HTTP-IRI	http://en.wikipedia.org/wiki/Puccini
	"<" IRI ">"	<urn:x-myns:music:puccini>
subject locator	”=” HTTP-IRI	=http://www.puccini.it/
	”=” "<" IRI ">"	=<ftp://example.com/opera/tosca/synopsis.txt>

- **Notes:**
  - An ID is like an XML Name (except that colon is explicitly not allowed)
  - HTTP IRIs get special status, in order to encourage their use as identifiers
  - A local ID becomes an item identifier; a QName becomes a subject identifier

Datatypes (1 of 3)

- **All XML Schema data types supported, but some are built-in, i.e. automatically recognized by the CTM parser**
- **String**
  - Same as xsd:string
  - Delimited by quotes or triple quotes

```
"A string containing \"quote\" marks"
"""Another string containing "quote" marks"""
"""Quote marks at the end of a string must always
be "escaped\""""
```

- **Issue: Do we need triple quotes?**

Datatypes (2 of 3)

- **IRI**
  - Same as xsd:anyURI. Delimited by ”<” and ”>” unless HTTP IRI

```
<http://en.wikipedia.org/wiki/>
http://en.wikipedia.org/wiki/
<urn:x-myns:music:puccini>
<ftp://example.com/opera/tosca/synopsis.txt>
```

- **Date**
  - Same as xsd:date

```
1858-12-22
```

- **Integer**
  - Same as xsd:integer

```
42
```

Datatypes (3 of 3)

- **Decimal**
  - Same as xsd:decimal

```
+42.0
```

- **Other datatypes expressed as strings**
  - Qualified using ^^

```
"2A" ^^xsd:hexBinary
"P1Y2M3DT10H30M" ^^xsd:duration
"12-22" ^^xsd:gMonthDay
```

Header

- **Encoding directive (opt)**
  - Specifies the character encoding:

```
%encoding "Shift-JIS"
```

- **Version directive (opt)**
  - Specifies the notation and version number:

```
%version ctm 1.0
```

- **Prefix declarations (opt)**
  - Specify a URI prefix:

```
%prefix w <http://en.wikipedia.org/wiki/>
```

- **Templates (opt)**

- **Issue: Should templates be allowed in the body as well?**

Templates (1 of 3)

- **Currently three kinds:**
  - name, occurrence, association
- **Name**
  - Tells parser to interprete typing topic as a name type and allows specification of scope

```
%name  foaf:name         # foaf:name is a name type
%name  country-code  @iso639  # all names of this type
                              # are scoped by iso639
```

- **Issue: Is scope on a name type bogus?**

Templates (2 of 3)

- **Occurrence templates**
  - Tells parser to interprete typing topic as an occurrence type and (optionally) the datatype and/or scope

```
%occur  bio:birthYear  ^^xsd:gYear
%occur  geo:lat_long   @deprecated
```

- **Issue: Is scope on an occurrence type bogus?**

Templates (3 of 3)

- **Association templates**
  - Allows role types to be omitted and scope to be specified globally

```
# binary (with scope)
%assoc  person  bio:born-in  place  @bio
# ternary
%assoc  victim  bio:killed-by  killer  how
# unary
%assoc  work    opera:unfinished
```

- **Issue: Is scope on an association type bogus?**

Document body

- **Consists of**
  - assertion blocks
  - verbose associations
- **Corresponds to (topic | association)* in XTM**
  - Except that associations can also be specified within an assertion block (see below)
- **Verbose associations**
  - So named because they do not exploit the compactness offered by templates (see below)

Assertion blocks

- **Consist of assertions about a single topic (referred to as the subject)**
  - Much like <topic> elements, except for ability to include associations
  - Assertions are delimited by semicolons
  - Assertion blocks are terminated by a period or an empty line

```
subject  # the subject of all assertions in this block
  assertion-1 ; # note the semicolon
  assertion-2 ;
  [...]
  assertion-n . # note the period
```

- **Issue: Should we call these topic blocks (or just topics)?**

Assertion blocks (cont.)

- **White space not significant, except**
  - Empty line terminates a block
  - White space required before a period (hmm...)

```
subject
  assertion-1 ;   # space before semicolon
  assertion-2;    # no space before semicolon
  assertion-3     # period omitted; replaced by empty line

# multiple assertions on one line:
subject
  assertion-1; assertion-2; assertion-3; assertion-4 .
# multiple blocks on one line:
subject assertion-1 . subject2 assertion-2 .
```

Assertions

- **Can be either identifier assignments or statements**
  - assignment of subject identifier
  - assignment of subject locator
  - name (or list of names of the same type)
  - occurrence (or list of occurrences of the same type)
  - association (or list of associations of the same type)
- **General form of all assertions is**

```
type   value   scope?   reifier?
```

  - However, in order to achieve compactness, some assertions are abbreviated
  - Note how subject-type-value corresponds closely to subject-verb-object (SVO)
- **Issue: Would an SOV variant syntax be of use in Korea and Japan?**

type value scope? reifier?

- **Can be**
  - either the ”kind” of identifier (subject identifier or subject locator)
  - or the TMDM [type] of the statement
- **When omitted implies that the assertion is either**
  - a subject identifier assignment (if the value is a URI)

    ```
    tosca w:Tosca .
    ```

  - a **tm:name-type** name (if the value is a string)

    ```
    tosca "Tosca" .
    ```

- **Otherwise must be a topic identifier or "="**
  - The latter can be thought to signify a subject locator

    ```
    tosca premiere-date 1900-01-14 .
    csgp-homepage =<http://www.puccini.it/> .
    ```

type **value** scope? reifier?

- **A topic identifier or a datatyped value**
  - Can be a single value

```
tosca premiere-date 1900-01-14 ;
      composed-by puccini .
```

  - Or a comma-separated list

```
tosca m:libretto-by illica, giacosa .
```

which is equivalent to

```
tosca m:libretto-by illica ;
      m:libretto-by giacosa .
```

type value **scope?** reifier?

- **A @ followed one or more topic identifiers**
  - Must be separated by white space (not commas)

```
opera "Opera" ;  "Oopera" @fi ;
  dc:description """A drama set to music; consists of...
                    overture and interludes."""
                    @wordnet ,
                 """Opera refers to a dramatic art form...
                    through the lyrics.""" @wikipedia en ,
                 """L'opera lirica è un genere...ed
                    il canto.""" @wikipedia it .
```

type value scope? **reifier?**

- **A ~ followed by a topic identifier**

```
tosca "Tosca"
  takes-place-in  rome  ~ tosca-in-rome .

tosca-in-rome "The Setting of Tosca in Rome" ;
  bibref  """Nicassio, Susan Vandiver:
             "Tosca's Rome: The Play and the Opera
             in Historical Perspective",
             University of Chicago Press
             (Chicago, 2002)""" .
```

- **Issue: How to reify the topic map itself (see below)?**

Names

- **Default name type inferred if no *type* component**

```
puccini "Puccini, Giacomo" .
```

- **Specified name type must be declared using a template:**
  - (otherwise the assertion will be interpreted as an occurrence) ...

```
%name foaf:name

puccini foaf:name "Giacomo Puccini" .
```

  - ... unless a *name flag* is present:

```
puccini %name foaf:name "Giacomo Puccini" .
```

  - Flags allow defaults and template information to be overridden, and templates to be omitted (e.g. in a simple TMQL update)

- **Issue: What would be a good syntax for this flag?**

Variants

- **Specified in parens after the base name**

```
boito
    "Boïto, Arrigo" ( "boito, arrigo"  @sort ) .
tchaikovsky
    "Tchaikovsky"
      ( "Tsjaikovski"  @search )
      ( "Tschaikovski" @search ) .
```

Occurrences

- **Type must be specified**
  - Value can be any datatyped value

```
puccini
  article        http://en.wikipedia.org/wiki/Giacomo_Puccini ;   # IRI
  description   "The greatest of the verismo composers" ;         # string
  date-of-birth  1858-12-22 ;                                     # date
  bibref         """Budden, Julian: "Puccini: His Life and Works",
                 Oxford University Press (Oxford, 2002)""" ;      # string
  xml-bibref                                                      # XML
    """<bibitem id="budden"><bib>Budden</bib>
       <pub>Budden, Julian:
       <highlight style="ital">Puccini: His Life and Works</highlight>,
       Oxford University Press (Oxford, 2006)</pub>
       </bibitem>""" ^^xsd:anyType .
```

- **Issue: Can quotes be avoided for XML values?**

Associations (1 of 3)

- **Verbose associations are specified outside assertion blocks**
  - Correspond to <association> elements in XTM
  - Templates are not required (or can be overridden)
  - Semicolon delimited role-type/role-player pairs

```
born-in( person puccini ; place lucca ) .

unfinished( work turandot ) .

killed-by( victim scarpia; killer tosca; how stabbing) .
```

- **Reifying roles**
  - Roles can only be reified in a verbose association:

```
pupil-of( pupil puccini ~ puccini-pupil-role ;
          teacher angeloni ) .
```

Associations (2 of 3)

- **Compact associations are specified inside assertion blocks**
  - Must have corresponding templates; allow role types to be omitted

```
%assoc person born-in place            # binary
puccini born-in lucca .
= born-in( person puccini ; place lucca ) .

%assoc work unfinished                 # unary
turandot unfinished .
= unfinished( work turandot ) .

%assoc victim killed-by killer how     # ternary
scarpia killed-by tosca stabbing .
= killed-by( victim scarpia; killer tosca; how stabbing) .
```

Associations (3 of 3)

- **Omitted roles**
  - It is possible to omit roles specified by the template in an instance of an association type
  - In the example, there is no role player for the role of "killer"

```
%assoc victim killed-by killer how

cavaradossi killed-by %role shooting .
```

- **Built-in association types**
  - type-instance and supertype-subtype have shorthands

```
puccini ISA composer .

composer AKO person .
```

- **Issues: Is AKO necessary? Are the keywords ISO and AKO acceptable?**

Some additional thoughts

Steve Pepper
Convenor SC34/SG3
Coordinator RDFTM Task Force

pepper@ontopia.net

WG3, Montreal, 2006-08-11

CTM as surface syntax?

- bio:killed-by(victim scarpia; killer tosca; how stabbing

```
%assoc victim is-killed-by killer how

is-killed-by bio:killed-by; "Killed by” .

scarpia is-killed-by tosca stabbing .
```

CTM as surface syntax?

- `bio:killed-by( victim scarpia; killer tosca; how stabbing)`

```
%ignore by
%assoc victim is-killed-by killer how
%assoc killer kills victim (by) how

is-killed-by bio:killed-by; "Killed by” .
kills bio:killed-by .

scarpia is-killed-by tosca stabbing .
tosca kills scarpia by stabbing .
```

CTM as surface syntax?

- `bio:killed-by( victim scarpia; killer tosca; how stabbing)`

```
%ignore by
%assoc victim is-killed-by killer how
%assoc killer kills victim by how
%assoc killer commits-suicide %self victim by how
is-killed-by bio:killed-by; "Killed by" .
kills bio:killed-by .
commits-suicide bio:killed-by.
scarpia is-killed-by tosca stabbing .
tosca kills scarpia by stabbing .
tosca commits-suicide by stabbing .
```

CTM as surface syntax?

- bio:killed-by(victim scarpia; killer tosca; how stabbing)

```
%ignore by
%assoc victim is-killed-by killer how
%assoc killer kills victim by how
%assoc killer commits-suicide %self victim by how
is-killed-by bio:killed-by; "Killed by” .
kills bio:killed-by .
commits-suicide bio:killed-by.
scarpia is-killed-by tosca stabbing .
tosca kills scarpia by stabbing .
tosca commits-suicide by stabbing .

# using CSS-style multiple subjects
kills, is-killed-by, commits-suicide { bio:killed-by }
```

SOV example (Korean)

- 우리 는 대한민국 입니다
  - uri neun te-han-min-guk imnida
    we [sub] Republic of Korea are

```
%version ctm 1.0 ov
%ignore  는
%assoc subject object  입니다
우리 는 대한민국 입니다  .

입니다( subject  우리  ; object  대한민국  ) .
are( subject  we  ;  object  RepublicOfKorea ) .
```

Indentation for hierarchies?

```
[
  animal
    mammal
      canine
        dog
        wolf
      feline
        tiger
        lion
    reptile
      snake
      neo-con
    insect
]
```

```
%assoc topic subtopicOf subtopic
 %hierarchy subtopicOf [
   arts
     movies
     television
     music
     ...
   business
     jobs
     real-estate
     investing
     ...
   computers
     internet
     software
     hardware
     ...
 ]
```

Discussion

Steve Pepper
Convenor SC34/SG3
Coordinator RDFTM Task Force

pepper@ontopia.net

WG3, Montreal, 2006-08-11

type-instance & subtype-supertype

- **How to represent type-instance and subtype-of shortcuts?**
  - ISA and AKO; isa and subtype-of; or delimiters?
- **Resolution**
  - Replace QNames with new concept for abbreviated URIs called “bangers” that use the bang instead of the colon

    ```
    foo!bar
    ```
  - Rule for concatenation of tuples is just concatenate without inserting any other characters
  - Then we use : and :: for instance-of and subtype-of respectively
  - Those that prefer a keyword (isa, ako) can define local IDs through templates, but the WG advises against doing this...
  - *...but see examples later that use isa and subtypeOf instead of : and ::*

Is the new “QName” syntax sufficient?

- What about query URIs?

```
%prefix foo <http://psi.example.org?id=>
foo/puccini, foo&puccini, foo+puccini
foo*puccini, foo’puccini, foo`puccini
foo!puccini
http://psi.example.org?id=puccini
```

- What about local IDs beginning with a number?

```
banger := URI "!" NMTOKEN
http://psi.oasis-open.org/iso/3166/#004
%prefix ctry http://psi.oasis-open.org/iso/3166/#
ctry!004
```

The CSS inspired syntax

```
%prefix m <http://psi.ontopia.net/music/#>
%prefix w <http://en.wikipedia.org/wiki/Tosca>

tosca w:Tosca ; "Tosca" ; : m:opera ;
  audio-recording  "7 47175 8" ;
  premiere-date  1900-01-14 ;
  synopsis
    http://www.metopera.org/synopses/tosca.html
      @web, operanews ;
  m:composed-by w:Giacomo_Puccini ;
  premiere teatro-costanzi ;
  takes-place-in roma ~ tosca-takes-place-in ;
  m:libretto-by
    w:Giuseppe_Giacosa ,
    w:Luigi_Illica .
```

```
%prefix m <http://psi.ontopia.net/music/#>
%prefix w <http://en.wikipedia.org/wiki/Tosca>

tosca {
  : m:opera ;
  w:Tosca ;
  "Tosca" ;
  audio-recording  "7 47175 8" ;
  premiere-date  1900-01-14 ;
  synopsis
    http://www.metopera.org/synopses/tosca.html
      @web, operanews ;
  m:composed-by w:Giacomo_Puccini ;
  premiere teatro-costanzi ;
  takes-place-in roma ~ tosca-takes-place-in ;
  m:libretto-by
    w:Giuseppe_Giacosa ,
    w:Luigi_Illica
}
```

Should WS+ be required before "."?

- **Problem**
  - Sometimes necessary in order to avoid "." being regarded as part of the local ID or QName

```
puccini born-in lucca.  # "lucca." taken as local ID
```

  - Inconsistency with ";" and "," unless WS also required here

```
puccini : composer ; born-in lucca ; died-in brussels .
```

  - Cf Andreas Sewe's proposal for a CSS inspired, period-free syntax
  - **Solution:** delimit assertion blocks w/ curlies; allow *local* IDs outside*

```
puccini {
   "Giacomo Puccini";
    isa:     composer;
    sid:     music!puccini;
    born-in: lucca;
    died-in: brussels;
  }
```

* Local IDs are purely syntactic and do not become part of the model, i.e. they explicitly *do not* result in item identifiers; these have their own syntax and the keyword **iid**.

Examples

```
foafName {
  sid: foaf!name;
  isa: tm/nametype;

}

puccini {
  "Giacomo Puccini";
  isa: composer;
  sid: music!puccini;
  born: 1858-12-22;
  died: 1924-01-14;
  foafName: "Fred Blogs" @pseudonym;
  iid: http://www.ontopia.net/tms/opera.ltm#puccini;
  composed: tosca @foo, butterfly @bar, turandot @baz;
  pupil-of: ponchielli @bar baz;
  died-in:  brussels;
  }
```

sid: subject identifier
slo: subject locator
iid: item identifer

isa: type-instance

NOTE: Willingness to consider use of colon after type, but no decision taken. Also willingness to consider replacing : and :: with isa and subtypeOf.

Examples

```
tosca { ”Tosca”;
  consists-of: [act1, act2, act3];
  }                            # possible syntax for ordering

csgp-homepage {”Centro Studie home page”;
  slo: http://www.csgp.it/;
  }

scarpia { ”Baron Scarpia”;
   isa: character;                # use instead of ":"?
   appears-in: tosca;
   killed-by: floria stabbing;
  }

{ "Some topic"; sid: foo/bar; descr: "This topic has no item
  identifier. Without the SID (or a SLO or IID) it would be
  invalid.” ; }

composer { subtypeOf: person }  # use instead of "::"?
```

Naming of association concepts

- **What (if anything) gets to be called “associations”?**
  - “associations” and “verbose associations” (as in the text)
  - “template expansions” and “associations” (as in the EBNF)
- ***Not discussed.***

SOV variant of CTM?

- **Do we want to allow an SOV variant of CTM**
  - http://en.wikipedia.org/wiki/Subject_Object_Verb

```
%version ctm 1.0 ov
puccini {lucca born-in:}
tosca {1900-01-14 premiere-date:}
```

- **Conclusion:**
  - Sam Oh will consider this from the perspective of the Korean language and create some examples based on the current proposal.

Templates

- **Restrict templates to header or allow anywhere?**
  - http://www.petesbox.net/pipermail/sc34wg3/2006-August/003258.html
  - ***Consensus:*** *Restrict to header*
- **What functionality for templates?**
  - Do we need topic templates? *Not discussed.*
- **What syntax for templates?**
  - http://www.petesbox.net/pipermail/sc34wg3/2006-August/003260.html
  - *Some ideas produced, but these are not recommendations*
- **What form should name and occurrence flags have (related to template syntax)?**
  - *Not discussed.*

Templating ideas (DB)

```
foaf-name {
  sid: foaf!name;
  isa: tm/nametype;
  }

composed {
  isa: ctm!object-property;
  type: music!composed-by;
  subject-role: composer;
  object-role: work;
  }

composed-by {
  isa: ctm!object-property;
  type: music!composed-by;
  subject-role: work;
  object-role: composer;
  }

puccini {
  foaf-name: ”Fred Blogs”;
  composed: tosca, butterfly, turandot;
  }
```

Syntax for reifying the topic map?

- Current proposal

  ```
  ~myMap
  ```

- Directive

  ```
  %topicmap myMap
  ```

- Predefined topic

  ```
  ctm!thisMap
  ```

- *Issue not discussed.*

Other issues

- **Assertion block with multiple subjects**
  - The following should be possible:

```
puccini, verdi, ponchielli { : composer }
```

- **Hierarchies**
  - Favourable reaction, but no recommendation.
- **Proposal for modifiers in parentheses**
  - Discussed but not adopted in current form
- **Strings**
  - Triple quotes not needed

Other ideas

- **The following ideas came up and need to be explained and justified by their various proponents**
  - Named regions of ctm doc (GDM)
  - Blocks of “subdocuments” w/ own header/body (LJH)
  - Named collections of topics (LJH)
  - Named scopes (SP)
  - Ordered lists (DB)
  - Embedded assertions
    May have value for reification. Should probably be avoided otherwise.
  - GDM’s proposal to be able to specify the base URI (for the construction of item identifiers) is no longer considered relevant