Meeting Minutes 23 January 2001

Warren's Metadata Review

Author:
Warren Hedley (Bioengineering Institute, University of Auckland)
Contributor:
Melanie Nelson (Physiome Sciences Inc.)

1 Introduction

This document lists some concerns that Warren had with the metadata part of the CellML specification that was principally devised by Melanie in October 2000. The idea is to have a reasonably useful specification sorted out in time for Warren and Melanie's visit to the SBML development team on January 29. Should be no problem!

OK, OK, so it was a bit harder than anticipated. No significant progress had been made on metadata before the Caltech visit. This document was picked up again on January 31 in Princeton, and some new topics of research added.

2 A Close Reading of the Metadata Spec

Warren's close reading of the October 2000 metadata spec caused him to worry about the following issues:

- **`<rdf:RDF>` and `<rdf:Description>`** — are these necessary? The `<rdf:RDF>` element doesn't really add anything (it certainly isn't descriptive); maybe a `<metadata>` element would be preferable. The `rdf:about` attribute on the `<rdf:Description>` element is clearly not appropriate when the element is embedded in the target file.
- **Uppercase RDF element names** — what the hell were they thinking? No other W3C vocabulary uses uppercase element names. It looks ugly.
- **Capitalisation** — Elements and attributes that appear in the CellML metadata schema should use underscores for word separation rather than capitalisation, regardless of what the RDF standard does.
- **Alternative Names** — is the Dublin Core title element really appropriate for storing alternative names?
- **Model Builder** — we need to discuss the contents of the `<dc:creator>` element and where they came from. (Are they based on a standard or existing data model?)
- **Creation Date** — In what way is the Dublin Core `<date>` element is extended for CellML?
- **Annotations** — Figure 7 has three levels of nested `<rdf:Description>` element. This shouldn't be necessary.

3 A Close Reading of the RDF Model and Syntax Specification

The following issues were regarded as potentially important by Warren during his close reading of the RDF Model and Syntax Specification[1]:

[1] http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

- **RDF Abbreviated Syntax** — The RDF specification offers two XML syntaxes for a model instance: a *serialization syntax* and a *abbreviated syntax*. In Section 2.2, the specification says "RDF interpreters are expected to implement both the full serialization syntax and the abbreviated syntax. Consequently, metadata authors are free to mix the two." Maybe our examples should use the abbreviated syntax where appropriate, where the metadata isn't structured. (See Section 2.2.2 of the specification for examples of this.)
- **The `<RDF>` element isn't necessary** — The spec says in Section 2.2.1[2] "The RDF element is a simple wrapper that marks the boundaries in an XML document between which the content is explicitly intended to be mappable into an RDF data model instance. The RDF element is optional if the content can be known to be RDF from the application context." It is possible to signal that elements are intended as metadata by just putting them in the RDF or some CellML-metadata namespace.
- **The `<Description>` element isn't necessary** — We currently embed `<Description>` elements within CellML elements and use the `about` attribute to reference the parent file, which is pretty odd. However the spec says in Section 2.2.1 that `<Description>` elements must have either an `ID` attribute or an `about` attribute, and we don't really want either. So we could indicate that data is metadata by using an appropriate namespace, or create a new `<cellml:metadata>` element. (Note that the `<Description>` element is useful when the metadata is stored in a file separate from the model definition that it refers to.)
- **Nested `<Description>` elements** — Section 2.2.2 of the specification shows the use of nested `<Description>` elements to structure metadata. However it also shows an abbreviated syntax (the last example in that section) with no intermediate `<Description>` elements. The presence of the intermediate elements is not illegal, but best practice would probably leave them out.

4 A Close Reading of the Dublin Core Documentation

Unbelievably, Warren could find absolutely no potentially important issues during his close reading of the Dublin Core Metadata Element Set 1.1[3] and Dublin Core Qualifiers[4] documentation. It appears that at least that much may have been done correctly.

5 Further Research

A design goal of CellML has always been to steal the work of others wherever possible and incorporate it into CellML. This means we can blame others if we later decide we don't like it. When the metadata part of the CellML specification was first drafted by Melanie Nelson in October 2000, some standards on which it was based were still unstable, so these have been revisited here. Also, additional documents made available since then have suggested new ideas ripe for the plagiarising.

5.1 Re-visiting BQS

The original October 2000 version of the metadata part of the CellML specification based its literature reference section on the January 2000 Bibliographic Query Service (BQS) submission from the European Bioinformatics Institute (EBI) to the Object Management Group (OMG). The EBI submitted a revised version of this documentation in July 2000 and then re-submitted this version with errata corrected in

[2] http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/#basicSyntax
[3] http://purl.org/dc/documents/rec-1990702.htm
[4] http://purl.org/dc/documents/rec/dcmes-qualifi ers-20000711.htm

September 2000, although these didn't appear on the web until later. The changes made in the revised version don't affect an XML serialization of that data model.

The BQS specification defines a number of bibliographic objects (or classes) using Interface Definition Language (IDL). The XML serialization of this data model was created from the IDL by Warren. The mapping is fairly straightforward, although it is not always clear whether list-type elements should be added to wrap lists of similar elements in a single root element.

5.2 ... and, of course, DocBook

Way back in the good 'ol days of CellML development, the website documentation and the presentations spoke of using Norman Walsh's DocBook[5] for citation information in CellML. To tell the truth, I think everyone just forgot about it in the exciting days of October 2000, when a preliminary specification was composed in a small number of days. DocBook defines two ways of specifying citation information: a `BiblioEntry` method and a `Bibliomixed` method. The first is an extremely comprehensive and structured schema for storing all kinds of bibliographic information. The second allows the insertion of tags directly into the information as it is to be presented to the user (mixed-content style.)

The `BiblioEntry` schema is more appropriate for use within CellML, and it is by far the most powerful citation DTD around. However it's complexity is its biggest drawback — we don't really want to force model authors or processing software, specifically search engine software, to have to deal with this kind of complexity. DocBook lacks clear facilities for defining reference types (e.g., journal article versus online resource), database identifiers (e.g., Medline identifiers), keywords, and doesn't provide a particularly intuitive structure for citing articles (see the example in Figure 3). Another weakness is the lack of a namespace for the DocBook syntax — it is implied that DocBook documents stand alone, and should not be embedded within other XML vocabularies.

Although DocBook is widely used and has been adopted as an OASIS[6] (Organization for the Advancement of Structured Information Standards) standard in February 2001. Several commercially available tools claim to support DocBook to some extent, and software and stylesheets that convert DocBook to other formats such as HTML and UNIX man pages is freely available.

5.3 vCard 3.0 in RDF

Renato Iannella at the University of Queensland, Australia has proposed a method for encoding vCard information in RDF. vCard is described in two standards documents at the Internet Engineering Task Force[7]:

- A MIME Content-Type for Directory Information[8]
- vCard MIME Directory Profile[9]

The vCard standards define a means of specifying a "Virtual Business Card" for use in applications such as e-mail. vCard can be used to store the types of personal information commonly found on business cards, for example names, titles, telephone numbers and addresses. The standards define a `text/directory` MIME Content-Type and the format of the content which may appear within blocks of that type. The vCard technology was originally developed by the versit consortium, and development and promotion is now actually managed by the Internet Mail Consortium[10].

[5] http://www.oasis-open.org/docbook/
[6] http://www.oasis-open.org/
[7] http://www.ietf.org/
[8] http://www.ietf.org/rfc/rfc2425.txt?number=2425
[9] http://www.ietf.org/rfc/rfc2426.txt?number=2426
[10] http://www.imc.org/

Ianella's proposal Representing vCard v3.0 in RDF[11] was last updated in January 1999. It appears to have not been submitted to any kind of standards organisation. It gives a comprehensive overview of how the vCard data model could be specified in an XML-based RDF format, and demonstrates how elements from this vocabulary can be combined with elements from the Dublin Core data model. The vocabulary is exactly what we need to specify personal information within the CellML data-model, which might be useful in model author, modification and reference contexts. However, it still remains to be seen if there is a more common or standard way of specifying this kind of information.

5.4 Comparison of Citation Formats

In this section possible methods of specifying citation information are demonstrated and compared. To make the examples more concise, namespace declarations are omitted and a consistent mapping of prefix to namespace is used, where the mappings are defined on the following **<model>** element.

```
<model
    xmlns:rdf="http://www.w3c.org/1999/02/22-rdf-syntax-ns#"
    xmlns:bqs="http://www.omg.org/lifesci/bqs/2000-09-14"
    xmlns:crdf="http://www.cellml.org/2000/cellml/RDF"
    xmlns:dc="http://purl.org/dc/elements/1.0"
    xmlns:dcq="http://purl.org/dc/qualifiers/1.0"
    xmlns:vCard="http://imc.org/vCard/3.0#" />
```

The citation part of the metadata for Melanie's EGF/EGFR example reaction model is given in Figure 1. That citation uses Dublin Core where appropriate and custom elements where necessary. Another representation of the information using nothing but the BQS data model is given in Figure 2. In Figure 3, the same information is marked up using the citation part of the DocBook DTD. If we were to get really tricky we could try and combine as much Dublin Core as possible with BQS and vCard concepts as shown in Figure 4.

Bearing in mind that one of our key goals is to plagiarise as much work as possible, the current scheme shown in Figure 1 is sub-obtimal because it uses a lot of custom elements from the CellML-RDF namespace. The BQS data model defines elements that match each of these custom elements, and so it seems a natural choice for defining things like journal article information.

Both the BQS and vCard models define fairly complete schema for defining personal information. The vCard model is fairly complete although the naming of some of the properties within the name and address structures are non-standard. For instance, **<vCard:Family>**, which corresponds to **<bqs:surname>**, is not consistent with current practice regarding internationalization. However the vCard model provides good facilities for specifying alernative versions of the same resouce, e.g., multiple e-mail addresses with appropriate metadata. In fact, it is probably too comprehensive for the needs of citation metadata — we are probably not interested in specifying the birthday of an author, or breaking down their address into fields. The BQS model is more suitable for specifying the appropriate levels of personal information that we require, although it lacks some basic elements like a person's title or suffix.

It has been suggested that we use as many relevant Dublin Core elements as possible to maximise the extraction of information from CellML documents by CellML-ignorant processing software. This is a nice idea, but depends heavily on the behaviour that we expect from such software. In particular, if we are using Dublin Core elements to describe the model, and then to describe citation information within the same block of metadata, CellML-ignorant software might confuse the two sets of data. It is worth checking whether the nesting of rdf:Description elements can be used to effectively create new sub-resources. In general, we should be looking for places where RDF is already used and how to establish a best practice.

[11] http://www.dstc.edu.au/Research/Projects/rdf/draft-iannella-vcard-rdf-00.txt

```
<rdf:RDF>
  <rdf:Description>

    <crdf:reference crdf:type="primary">
      <rdf:Description>
        <dc:identifier crdf:identifierType="medline">
          20241869
        </dc:identifier>
      </rdf:Description>
    </crdf:reference>

    <crdf:reference crdf:type="secondary">
      <rdf:Description>
        <dc:date
            dcq:dateScheme="W3CDTF"
            dcq:dateType="created">
          2000-10-05
        </dc:date>
        <dc:title>
          Action potential and contractility changes in [Na(+)](i)
          overloaded cardiac myocytes: a simulation study
        </dc:title>
        <dc:type>journal article</dc:type>
        <crdf:volume>78</crdf:volume>
        <crdf:issue>5</crdf:issue>
        <crdf:first_page>2392</crdf:first_page>
        <crdf:last_page>2404</crdf:last_page>
        <crdf:journal crdf:journalScheme="abbreviation">
          Biophys J
        </crdf:journal>
        <dc:creator>
          <rdf:Sequence>
            <rdf:li>
              <rdf:Description>
                <crdf:first_name>G</crdf:first_name>
                <crdf:surname>Faber</crdf:surname>
                <crdf:mid_initials>M</crdf:mid_initials>
              </rdf:Description>
            </rdf:li>
            <rdf:li>
              <rdf:Description>
                <crdf:first_name>Y</crdf:first_name>
                <crdf:surname>Rudy</crdf:surname>
              </rdf:Description>
            </rdf:li>
          </rdf:Sequence>
        </dc:creator>
      </rdf:Description>
    </crdf:reference>

  </rdf:Description>
</rdf:RDF>
```

FIGURE 1: The citation part of the metadata for Melanie's EGF/EGFR example reaction model, as first coded in 21 November 2000. Melanie noted that the **<dc:type>** element may not be the preferred way to indicate reference type, and that the **crdf:journalScheme** attribute on the **<crdf:journal>** element is non-standard. Note that journal and person information is defined using elements in the CellML-RDF namespace.

The authors are listed inside a **<rdf:Sequence>** element which implies that the order matters.

```
<rdf:RDF>

  <rdf:Description crdf:reference_type="primary">
    <bqs:identifier> Medline/20241869  </bqs:identifier>
  </rdf:Description>

  <rdf:Description crdf:reference_type="secondary">
    <bqs:type> JournalArticle </bqs:type>
    <bqs:title>
      Action potential and contractility changes in [Na(+)](i)
      overloaded cardiac myocytes: a simulation study
    </bqs:title>
    <bqs:subject>
      <bqs:keyword> signalling pathway </bqs:keyword>
      <bqs:keyword> enzyme kinetics </bqs:keyword>
    </bqs:subject>
    <bqs:authors>
      <bqs:person>
        <bqs:surname> Faber </bqs:surname>
        <bqs:first_name> G </bqs:first_name>
        <bqs:mid_initials> M </bqs:mid_initials>
      </bqs:person>
      <bqs:person>
        <bqs:surname> Rudy </bqs:surname>
        <bqs:first_name> Y </bqs:first_name>
      </bqs:person>
    </bqs:authors>
    <bqs:date> 2000-10-05 </bqs:date>
    <bqs:from_journal>
<!--  <bqs:name></bqs:name>
      <bqs:issn></bqs:issn>  we could use these too!  -->
      <bqs:abbreviation>Biophys J</bqs:abbreviation>
    </bqs:from_journal>
    <bqs:volume> 78 </bqs:volume>
    <bqs:issue> 5 </bqs:issue>
    <bqs:first_page> 2392 </bqs:first_page>
    <bqs:last_page> 2404 </bqs:last_page>
  </rdf:Description>

</rdf:RDF>
```

FIGURE 2: A representation of the citation information from Figure 1 defined using the BQS data model, with some additional subject data.

```
<rdf:RDF>

  <biblioentry crdf:reference_type="primary">
    <bibliomisc role="identifier"> Medline/20241869 </bibliomisc>
  </biblioentry>

  <biblioentry crdf:reference_type="secondary">
    <biblioset relation="article">
      <title>
        Action potential and contractility changes in [Na(+)](i)
        overloaded cardiac myocytes: a simulation study
      </title>
      <authorgroup>
        <author>
          <firstname> G </firstname>
          <othername role="middle"> M </othername>
          <surname> Faber </surname>
        </author>
        <author>
          <firstname> Y </firstname>
          <surname> Rudy </surname>
        </author>
      </authorgroup>
      <pubdate> 2000-10-05 </pubdate>
      <volumenum> 78 </volumenum>
      <issuenum> 5 </issuenum>
      <pagenums> 2392-2404 </pagenums>
    </biblioset>
    <biblioset relation="journal">
      <abbrev> Biophys J </abbrev>
    </biblioset>
    <bibliomisc role="keyword"> signalling pathway </bibliomisc>
    <bibliomisc role="keyword"> enzyme kinetics </bibliomisc>
  </biblioentry>

</rdf:RDF>
```

FIGURE 3: Another representation of the citation information from Figure 1 defi ned using the citation part of the DocBook DTD.

```
<rdf:RDF>

  <rdf:Description crdf:reference_type="primary">
    <bqs:identifier> Medline/20241869  </bqs:identifier>
  </rdf:Description>

  <rdf:Description crdf:reference_type="secondary">
    <dc:type> JournalArticle </dc:type>
    <dc:title>
      Action potential and contractility changes in [Na(+)](i)
      overloaded cardiac myocytes: a simulation study
    </dc:title>
    <dc:date
        dcq:dateScheme="W3CDTF"
        dcq:dateType="created">
      2000-10-05
    </dc:date>
    <bqs:from_journal>
      <bqs:abbreviation>Biophys J</bqs:abbreviation>
    </bqs:from_journal>
    <bqs:volume> 78 </bqs:volume>
    <bqs:issue> 5 </bqs:issue>
    <bqs:first_page> 2392 </bqs:first_page>
    <bqs:last_page> 2404 </bqs:last_page>
    <dc:creator>
      <rdf:Sequence>
        <rdf:li>
          <vCard:FN> G M Faber </vCard:FN>
          <vCard:N parseType="Resource">
            <vCard:Family> Faber </vCard:Family>
            <vCard:Given> G M </vCard:Given>
          </vCard:N>
        </rdf:li>
        <rdf:li>
          <vCard:FN> Y Rudy </vCard:FN>
          <vCard:N parseType="Resource">
            <vCard:Family> Rudy </vCard:Family>
            <vCard:Given> Y </vCard:Given>
          </vCard:N>
        </rdf:li>
      </rdf:Sequence>
    </dc:creator>
  </rdf:Description>

</rdf:RDF>
```

FIGURE 4: Yet another representation of the citation information from Figure 1. This example uses as much of the Dublin Core as possible, with elements from the BQS thrown in where appropriate, and vCard syntax used for personal information.