

Meeting Minutes 15 January 2001

Major Changes to CellML from Royal Society Paper

Author:

Warren Hedley (Bioengineering Institute, University of Auckland)

Contributors:

Melanie Nelson (Physiome Sciences Inc.)

Poul Nielsen (Bioengineering Institute, University of Auckland)

1 Introduction

January 15 2001 was the final deadline for submission of the "A Short Introduction To CellML" paper for publication in the July 2001 "Philosophical Transactions of the Royal Society of London". Obviously, this would be the ideal time to re-define most of the CellML syntax. A minor re-naming of some attributes on the `<unit>` element (the writing of the units section of the CellML spec was a major component of the paper's writing) turned into a complete re-think on the definition of component references, variable references and relationship definitions in CellML.

2 The Problem With References

The driving force behind the changes was a need for consistency across the CellML language. At this stage, different parts of the syntax had been developed at different times, and the thinking at the time of development had left its mark. The most notable of the influences was the CRIM documents of early 2000. This excellent work, which had been contributed to by intellectuals from all walks of life (like Karl T and Richard C), proposed creating an element with a name equal to the name of the element being referenced followed by an underscore and the keyword "ref" and a `name_ref` attribute that could be used to reference the target element by the value of its `name` attribute. This had led to, for instance, the `<component_ref>` and `<variable_ref>` elements in CellML.

At the same time the `<variable>` element had a `units` attribute which was now used to reference a keyword from the CellML units dictionary or, more commonly, to reference some complex units defined within the same component or model. The `units` attribute was named so because in versions of CellML from 1999, the units were actually fully defined in each `units` attribute using the "[m,mol,1]" notation discussed in the [10 January 2001 meeting minutes](#)¹.

Finally, it could be said that the `<relationship>` element was really a reference to a relationship, where this was not actually defined in XML, but was defined in the specification. When one considers that we permit the model author to define their own new classes of relationship (such as "is_connected_to") where these relationships could conceivably be defined in XML (using an ontology definition language like OIL, for instance), it seemed appropriate to view the `<relationship>` element as a reference and use a scheme consistent with other references in CellML. Another problem with the `<relationship>` element at that point was the `class` attribute, which was a keyword that we didn't want to be confused with any upcoming class definition ideas or ontology definitions.

3 The Solution

It was pointed out that so far we had had no problems dealing with a reference to the name of a units definition being defined with the `units` attribute on the `<variable>` element. So instead of using the

¹http://www.cellml.org/private/progress_reports/20010110_meeting_minutes.html

name_ref attribute everywhere else to reference components by name, why not just use the element name as the attribute name?

And in addition, let the element name of the referencing element better reflect that element's role. So if the sole purpose of the **<major_component_ref>** element is to reference a component, then make it a **<component_ref>** element with a **role** attribute value of "major". This removes both the **<major_component_ref>** and **<minor_component_ref>** elements from the DTD.

Finally, the **class** attribute on the **<relationship>** element can also be thought of as a reference where the target is actually a definition in the specification. The **<relationship>** element thus became the **<relationship_ref>** element, and its **class** attribute became a **relationship** attribute.

3.1 Changes To Units Definition

The writing of the Royal Society paper had led to a fairly thorough rethink on the specification of units, that wound up on January 12 with the scheme proposed in the [10 January 2001 meeting minutes](#)².

Just when you think you have something good going, Poul Nielsen waltzes (struts?) in and decides he's not happy with it. Not long after we had what we thought was a complete units scheme with documentation suitable for inclusion in the CellML specification (above), Poul started recommending changes. A quick summary of the changes and the justifications are as follows:

1. The **scale** attribute on the **<unit>** element was re-named the **prefix** attribute. This is what the people at NIST call it, and as we hope to be a respected standards organisation like NIST, we should follow existing standards ourselves wherever possible.
2. The **type** attribute on the **<unit>** element was re-named the **units** attribute. The word "type" was too ontology-like: Poul favoured a re-naming to **units**, Warren favoured a re-naming to **name_ref**, and Melanie suggested **units_ref**. Basically, we wanted something that could be thought of as consistent with everything else, and since we'd already broken consistency with SBML with the previous change, it didn't really matter if we became more inconsistent. Using **units** is consistent with the use of **<variable>** elements and also all referencing schemes throughout CellML.
3. It was decided that the values of the **prefix** attribute on the **<unit>** element should be the full name of the prefixes as they appear in the NIST table to be consistent with the values of the **units** attribute. So "milli" should be used in place of "m". This also neatly sidesteps the mu/u/micro problem.
4. The **numberOf** keyword was removed from the units dictionary. Poul didn't like **numberOf** because it was a quantity (like "length") rather than a unit, and because there is implied integer behaviour that we possibly don't want. He suggested that the word **items** would be more technically correct. This keyword was left out of the paper, but may yet be re-introduced into the CellML specification after correspondence with the SBML folks.

4 Before And After

Just to be absolutely clear on the changes that were made on January 15 2001, Figure 1 and Figure 2 contain examples of the affected portions of a CellML document *before* and *after* the changes are applied.

²http://www.cellml.org/private/progress_reports/20010110_meeting_minutes.html

```
<model name="example_model">

  <units name="concentration">
    <unit scale="m" type="mole" />
    <unit exponent="-1" type="litre" />
  </units>

  <component name="A">
    <variable name="A" private_interface="out" units="concentration" />
  </component>

  <component name="B">
    <variable name="B" public_interface="in" units="concentration" />
  </component>

  <group>
    <relationship class="is_encapsulated_by" />
    <relationship class="is_contained_in" />
    <major_component_ref name_ref="A" />
    <minor_component_ref name_ref="B" />
  </group>

  <connection>
    <component_ref name_ref="A">
      <variable_ref name_ref="A" />
    </component_ref>
    <component_ref name_ref="B">
      <variable_ref name_ref="B" />
    </component_ref>
  </connection>

</model>
```

FIGURE 1: A CellML fragment containing examples of the CellML elements that were affected by the changes made to CellML syntax on January 15 2001 *before* the changes.

```
<model name="example_model">

  <units name="concentration">
    <unit prefix="milli" units="mole" />
    <unit units="litre" exponent="-1" />
  </units>

  <component name="A">
    <variable name="A" private_interface="out" units="concentration" />
  </component>

  <component name="B">
    <variable name="B" public_interface="in" units="concentration" />
  </component>

  <group>
    <relationship_ref relationship="is_encapsulated_by" />
    <relationship_ref relationship="is_contained_in" />
    <component_ref component="A" role="major" />
    <component_ref component="B" role="minor" />
  </group>

  <connection>
    <component_ref component="A">
      <variable_ref variable="A" />
    </component_ref>
    <component_ref component="B">
      <variable_ref variable="B" />
    </component_ref>
  </connection>

</model>
```

FIGURE 2: The CellML fragment from Figure 1 *after* the changes made to CellML syntax on January 15 2001 have been applied.
