Coordination Topics for Discussion with Developers

SBML Team

September 2011

Things we think would benefit from discussion

Things we think would benefit from discussion

(not necessarily now)

Things we think would benefit from discussion

Why do we keep lists of tools that support a standard?

- Benefits to users

- Benefits to tool developers

- But are the goals for these two groups the same ?

Data format validation

- Standardise the output of validation for all formats?

Data format validation

- Validation of SBML has evolved

libsbml-2

```
ParseMessage  (  unsigned int id = 0,
          const std::string& message = “”,
                           unsigned int line = 0,
                           unsigned int col = 0);
```

libsbml-5

```
SBMLError (const unsigned int errorId = 0
, const unsigned int level = DEFAULT
, const unsigned int version = DEFAULT
, const std::string& details = “”
, const unsigned int line = 0
, const unsigned int column = 0
, const unsigned int severity = ERROR
, const unsigned int category = SBML );
```

- Worth formalising ?

Data format validation

- Worth formalising ?

```
<problem category='%s' code='%u' severity='%s'>
   <location line='%u' column='%u'/>
   <message lang='en_US'>
     <![CDATA[ ... ]]>
   </message>
   <shortmessage lang='en_US'>
     <![CDATA[ ... ]]>
   </shortmessage>
   <detail category='%u' severity='%u' />
   <excerpt> <![CDATA[ ... ]]> </excerpt>
</problem>
```

Changing repository from svn to git

- Is git a better option ?

- Is it worth the change ?

SBML L3 package code development

- Feedback on developing code

- Should we invest time in creating autogeneration systems ?

GNU Make vs CMake

- Feedback on this change

- Do we need to continue to support both ?

Things we think would benefit from discussion

- Lists of tools supporting a standard

- Validation: standards for validator output

- Repository change

- L3 package code development

- GNU Make vs CMake