

SBML Level 2 features requested

Andrew Finney

Chemical Entities are *not* Compartment Specific

- Issues with SBML Levels 1 and 2
 - No way to relate the two species together when they represent the same biochemical entity in two different compartments
 - SBML ‘species’ are really ‘pools’ (Pedro Mendes)
 - In reality chemical entities can have any number of pools in different compartments
 - The same reaction can occur in several compartments
 - SBML has no way to directly represent this
 - SBML represents transport between compartments as a reaction between ‘pools’ in different compartments
 - Need to retain this flexibility

Raised by Fabien Campagne, Jeremy Zucker and others

Interim Level 2 Solution

- place the common identifier in the `species name` field
- place the 'pool' identifier in the `species id` field
- will work in tools that display the `name` field to the users but don't have any concept of common species
- however really this is a poor use of SBML Level 2
 - misuses the `name` field to store relational data
 - parsers that do have a concept of common species won't be able to interpret the `name` field reliably this way

Outline Level 3 Solution

- Part of a complex species proposal
- Create a new optional element `speciesType`
 - represents common species
 - has at least `id` and `name` attributes
- Add a new `speciesType` attribute to `species`.
- `species` elements then represent pools.
- All these new components would be optional.

- How do we relate identical reactions that occur in different compartments?
- How do we merge this with the other concepts in the complex species proposal?

Identification of Dependent Variable in Algebraic Equation

- It can be important for reproducing a given numerical solution to specify the dependant variable on an algebraic rule.
 - This is not possible in SBML.
- **Solution:**
 - add an optional attribute `dependantVariable` on the `algebraicRule` element that contains a variable id.

Raised by: Mark Vass and Herbert Sauro

Other Features Requested

- **Define functions anywhere in MathML not just in function definitions**
 - limited use as these functions would have no scope outside the expression that encodes them
- **Define Species molecular weight**
 - use Chemical Markup Language through BioPAX?
- **Add biochemical constants**
- **Nested Unit definitions**
- **Bounds on**
 - reaction flux
 - initial conditions
 - parameters
 - (every numerical value apart from stoichiometry)
- **Label algebraic rules that are conservation laws**