Quick introduction to the rsbml package

Michael Lawrence

April 11, 2007

Importing an SBML document

Most users will begin an rsbml session by importing an SBML file into R. In the example below, we load an SBML file describing the glycolysis pathway. It is also possible to parse an R character vector instead of an external file.

```
> library(rsbml)
> doc <- rsbml_read(system.file("sbml", "GlycolysisLayout.xml",
+     package = "rsbml"))
```

If errors are encountered, the function throws an error along with warnings describing the specific problem(s) with the document. Otherwise, the result is an opaque object referring to a low-level libsbml data structure. From here, the user currently has two options for accessing the data: as an S4 object conforming to the SBML document object model or as a Bioconductor graph object.

The S4 object representation

Converting the opaque libsbml parse result to an S4 object is simple:

```
> dom <- rsbml_dom(doc)
```

The result contains all of the information from the SBML. Methods exist for accessing every element of the SBML specification (up to L2V1). Here is how one would retrieve all of the species ID's from the SBML model:

```
> sapply(species(model(dom)), id)
```

```
 [1] "Glucose"                   "Glucose-6-phosphate"
 [3] "Fructose-6-phosphate"      "Fructose-1_6-bisphosphate"
 [5] "Dihydroxyacetonephosphate" "Glyceraldehyd-3-phosphate"
 [7] "1_3-Bisphosphoglycerate"   "3-Phosphoglycerate"
 [9] "2-Phosphoglycerate"        "Phosphoenolpyruvate"
[11] "Pyruvate"                  "ATP"
[13] "ADP"                       "H+"
[15] "NAD+"                      "NADH"
[17] "H2O"                       "Pi"
```

The user may also modify every part of the model. Note that very little validation is performed in response to modifications. See **??** for a guide to checking SBML models for consistency problems.

The graph representation

All SBML models have an implicit graphical structure. To extract this into a Bioconductor graph object, type the following:

```
> g <- rsbml_graph(doc)
> nodes(g)
```

```
 [1] "Glucose"                    "Glucose-6-phosphate"
 [3] "Fructose-6-phosphate"       "Fructose-1_6-bisphosphate"
 [5] "Dihydroxyacetonephosphate"  "Glyceraldehyd-3-phosphate"
 [7] "1_3-Bisphosphoglycerate"    "3-Phosphoglycerate"
 [9] "2-Phosphoglycerate"         "Phosphoenolpyruvate"
[11] "Pyruvate"                   "ATP"
[13] "ADP"                        "H+"
[15] "NAD+"                       "NADH"
[17] "H2O"                        "Pi"
[19] "Hexokinase"                 "Phosphoglucoseisomerase"
[21] "Phosphofructokinase"        "Aldolase"
[23] "triose_phosphate_isomerase" "GAP_Dehydrogenase"
[25] "Phosphoglyceratekinase"     "Phosphoglyceratemutase"
[27] "Enolase"                    "Pyruvatekinase"
```

Note that the list of node ID's contains all of the species ID's retrieved from the S4 object above. The additional ID's are for the reaction nodes. At this point, the graph can be passed to other packages, such as RBGL, Rgraphviz, etc.

Checking SBML models for consistency

The SBML specification provides many complex rules that ensure an SBML model is internally consistent. The following is an example of checking a document against those rules.

```
> rsbml_check(doc)
```

```
[1] TRUE
```

If there were any problems with the document, they would be communicated as warnings. If you would like to compute on the problems, the following returns a data structure describing the errors along with their positions in the SBML, if available.

```
> doc <- rsbml_read("non-existent-file.xml", check = FALSE)
> rsbml_problems(doc)

$warnings
list()

$errors
list()

$fatals
$fatals[[1]]
$fatals[[1]]$line
[1] 0

$fatals[[1]]$column
[1] 0

$fatals[[1]]$message
[1] "File 'non-existent-file.xml' does not exist."
```

Writing SBML documents

After creating/manipulating SBML objects in R, the result may be translated back to XML in two different ways: directly to a file or to an R character vector. This example is of the latter:

```
> xml <- rsbml_xml(doc)
```

The result is a string XML representation of the SBML model.

More information

For more details, please see the online help for the rsbml package. If you encounter problems, please email lawremi at the domain iastate.edu.