



0100100010011101

ISB

SBML level 3

Composite (hierarchical) models

Fedor Kolpakov

Institute of Systems Biology

Laboratory of Bioinformatics,
Design Technological Institute of Digital Techniques

Novosibirsk, Russia

- Requirements
- Main ideas
- Contact concept
- DTD
- User interface
- Algorithm for conversion of CompositeModel to flat SBML model
- Proof of concept - implementation in BioUML workbench
- Open issues and alternatives

Requirements

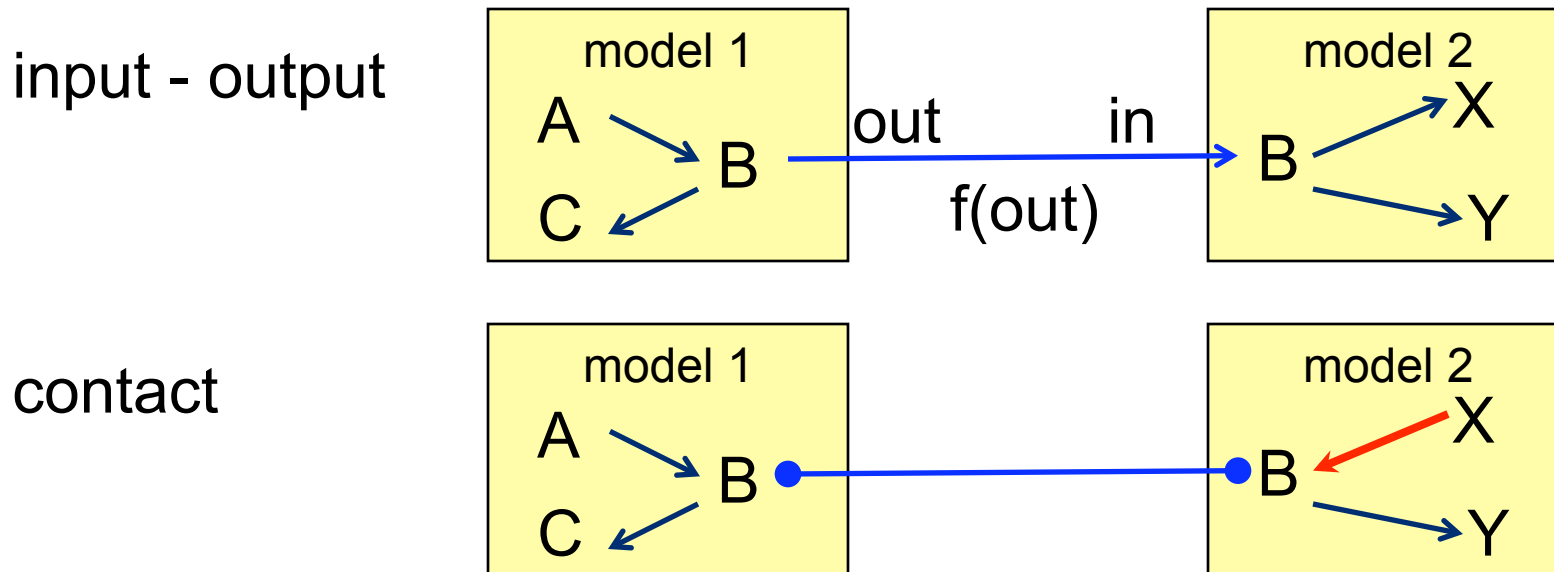
1. Facilitate the reuse of existing models (as components)
2. Easy adaptation of existing software
 - 2.1. Algorithm for conversion of composite (hierarchical) models into flat SBML model
 - 2.2. Simple API (for example in libSBML)
 - 2.3. Maximum backward compatibility
3. User interface should be intuitive and straightforward for biologists.

Main ideas

- Starting point:
Hierarchical Modelling spec. by Stefan Hoops and Ion Moraru
<http://ntcnp.org/twiki/bin/view/VCell/HierarchicalModeling>
- Contact concept
- Differences with initial proposal:
 - listOfReplacements -> listOfChanges
 - listOfPorts -> listOfConnections
 - conversionFactor -> connection.transferFunction
- Algorithm for conversion of composite (hierarchical) models into flat SBML model

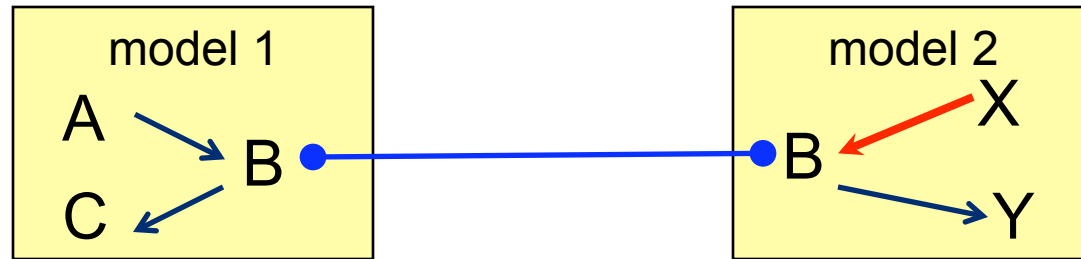
Contact concept

- Contact between two models specifies that two species in these models are the same entity (for example, chemical substance or protein).
- Contact relationship between substances can not be replaced by port or input-output relationship.
- Example – 2 metabolic pathways where ATF is used and produced.



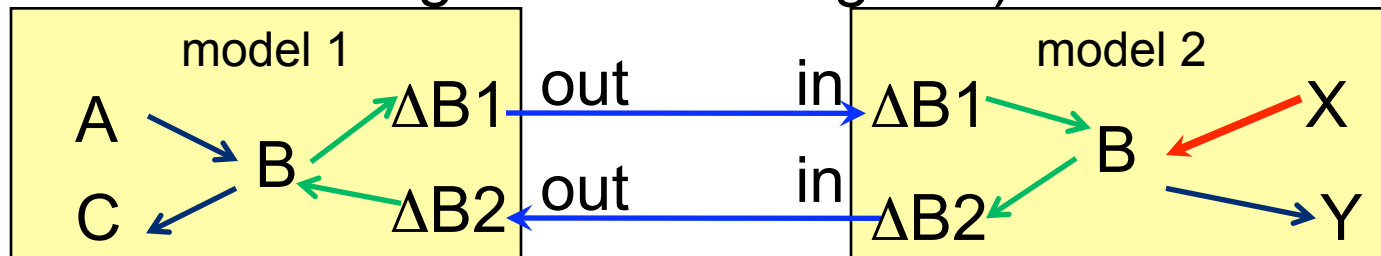
Contact – alternative (CellML)

contact



input – output

(like CellML or messages between agents)



This approach is hardly suitable for SBML:

- there is no concept of delta variable in SBML
- it does not supported by SBML software
- computational problems
(stability of numeric solution of ODE)

XML elements

- <listOfSummodels>
- <listOfChanges>
 - <propertyChange>
 - <added>
 - <deleted>
- <listOfConnections>
 - <connection>
 - <contact>

<listOfChanges>

All models changes are divided into 3 groups:

- **propertyChange** – changing some parameter value, initial value, species property (bound, const).
- **deleted** – deleted model elements or array items (for example, <assignment> element in event)
- **added** – newly created model element or array items (for example, <assignment> element in event)

This approach allows to minimize modeler efforts when only some properties of model element should be changed.

Alternative approach with adding/deleting element is much more complicated from modeler view point. Let us suppose that we need to change boundaryCondition property for one specie that is involved in 12 reactions. In this case for semantic integrity we need to delete previous 12 reactions and create 12 new reactions.

DTD: <listOfChanges>

```
<!ELEMENT listOfChanges (propertyChange*, deleted*, added*)>
```

```
<!ELEMENT propertyChange>
```

```
<!ATTLIST propertyChange
```

```
  elementID                CDATA                #REQUIRED
```

```
  property                  CDATA                #REQUIRED
```

```
  newValue                  CDATA                #REQUIRED
```

```
  oldValue                  CDATA                #REQUIRED
```

```
  comment                   CDATA
```

```
>
```

```
<!ELEMENT deleted>
```

```
<!ATTLIST deleted
```

```
  elementId                 CDATA                #REQUIRED
```

```
  comment                   CDATA
```

```
>
```

```
<!ELEMENT added (SBML_top_level_element | Event.Assignment)* >
```

<propertyChange>

```
<!ELEMENT propertyChange>
<!ATTLIST propertyChange
  elementID          CDATA
                    #REQUIRED
  property           CDATA                    #REQUIRED
  newValue           CDATA                    #REQUIRED
  oldValue           CDATA                    #REQUIRED
  comment            CDATA
>
```

Attributes:

- **elementId** – SBML model element identifier
- **property** – name of property that was changed
- **newValue** – new property value
- **oldValue** – old property value
- **comment** – comment about changes

Suggested approach allows to change only properties, that are:

- simple, that is can be represented by simple string value. This value can be further parsed as number or boolean if necessary;
- meaningful for executable model generation.

SBML elements and their properties that can be changed using <propertyChange>

SBML component	Property	Comment
compartment	Size	
species	initialAmount initialConcentration boundaryCondition constant	
reaction	kineticLaw	
reaction/specieReference	Stoichiometry	
parameter	value constant	reaction parameters are also mapped into parameters
initialAssignment	Math	
function	Lambda	
rule	Math type	?
event	Trigger delay assignment[i]/math	
constraint	Math message	

<deleted>

Allows to delete model elements or array items (for example, <assignment> element in event)

```
<!ELEMENT deleted>
<!ATTLIST deleted
  elementId          CDATA
  #REQUIRED
  comment            CDATA
>
```

Attributes:

- **elementId** – SBML model element identifier
- **comment** – comment about changes

<added>

Allows to add any top level SBML element (compartment, specie, reaction, parameter, initialAssignment, function, rule, event, constraint) as well as event assignments.

```
<!ELEMENT added (SBML_top_level_element |  
Event.Assignment)* >
```

SBML_top_level_element:

compartment | species | reaction | parameter | initialAssignment |
function | rule | event | constraint

<listOfConnections>

<!ELEMENT **listOfConnections** (connection|contact)>

<!ELEMENT **connection** (**transferFunction**) >

<!ATTLIST connection

input CDATA #REQUIRED

output CDATA #REQUIRED

comment CDATA

>

<!ELEMENT **transferFunction** (**mathML**) >

<!ELEMENT **contact**>

<!ATTLIST contact

input CDATA #REQUIRED

output CDATA #REQUIRED

comment CDATA

>

Algorithm for conversion of composite (hierarchical) models into flat SBML model

For each submodel:

1. Provide that all ID in flat model are unique: replace ID by modelID_componentID (species, reactions, events, parameters, ...)
2. Apply <listOfChanges> to each submodel
3. Move all assignments, functions and events from submodels to top model
4. Replace all connections:
 - generate an assignment out = in
 - in (specie) – set boundaryCondition = true
5. Replace all contacts:
replace out_ID by in_ID for all expressions
6. Convert submodel into compartment with volume 1
 - create compartment with name equal to submodel name
 - move all elements (species, reactions, etc.,) into corresponding listOf...
 - set up for them corresponding compartment (former submodel)

Initial implementation

BioUML workbench

File Module Diagram Help

Modules \ Data \ Plugins \

- modules
 - BMOND2
 - Biopath
 - ChEBI
 - Ensembl
 - GO
 - Reactome - Clamidia
 - SBO
 - UniProt
 - test
 - Data
 - Diagrams
 - composite
 - model1
 - model2
 - Dictionaries
 - Images
 - Simulation

test : model1

```

graph LR
  A[A 20.0] -- J1 --> B[B 0.0]
  B -- J2 --> C[C 0.0]
  P1((P1 1.0)) --- J1
  P2((P2 1.0)) --- J2
  
```

Diagram

Title: model1
Diagram type: Pathway simulation
Layouter: "...forceDirectedLayout..."
Role:
Initial time: 0.0
Completion time: 10.0

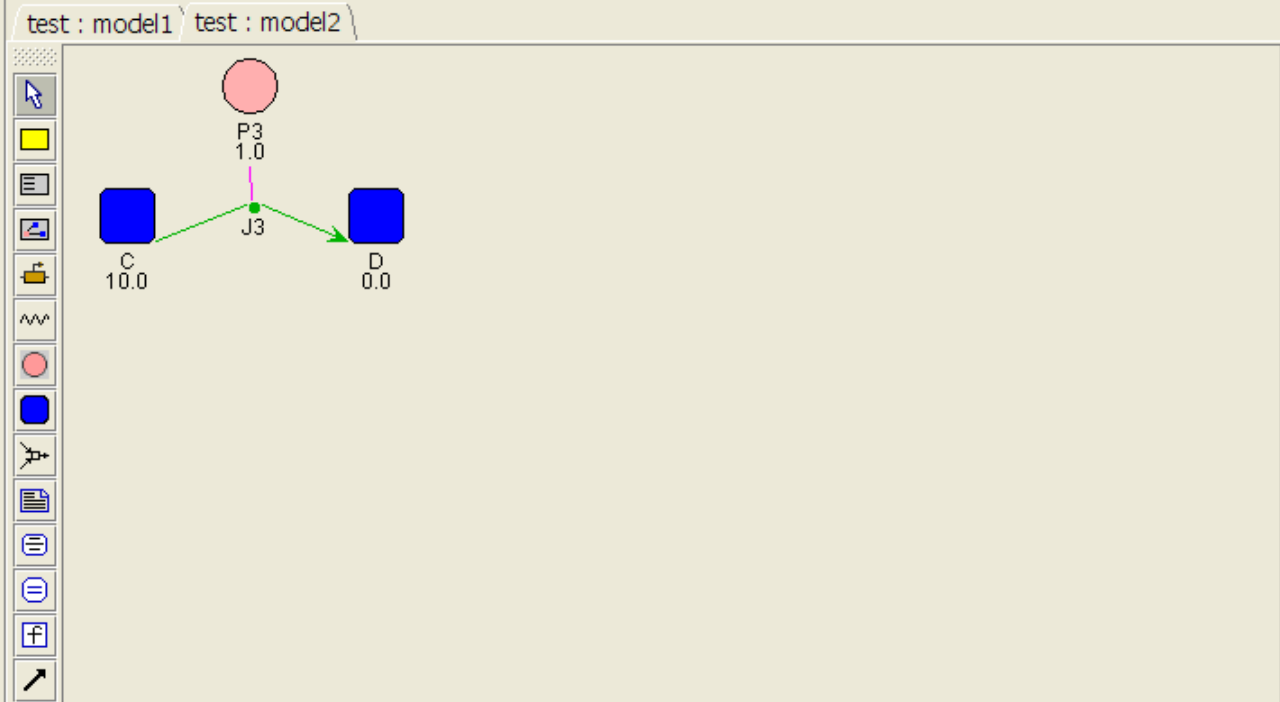
INFO : Communicating with update site:<http://server2.biouml.org/update/> ...
 ERROR : Network connection problems encountered during search.

Description Edges Experiments Math editor References Microarray Filter Application Log
 Clipboard Layouter pane Parameters Variables JAVA MATLAB JavaScript Search linked Skin



Modules \ Data \ Plugins \

- modules
 - BMOND2
 - Biopath
 - ChEBI
 - Ensembl
 - GO
 - Reactome - Clamidia
 - SBO
 - UniProt
 - test
 - Data
 - Diagrams
 - composite
 - model1
 - model2
 - Dictionaries
 - Images
 - Simulation



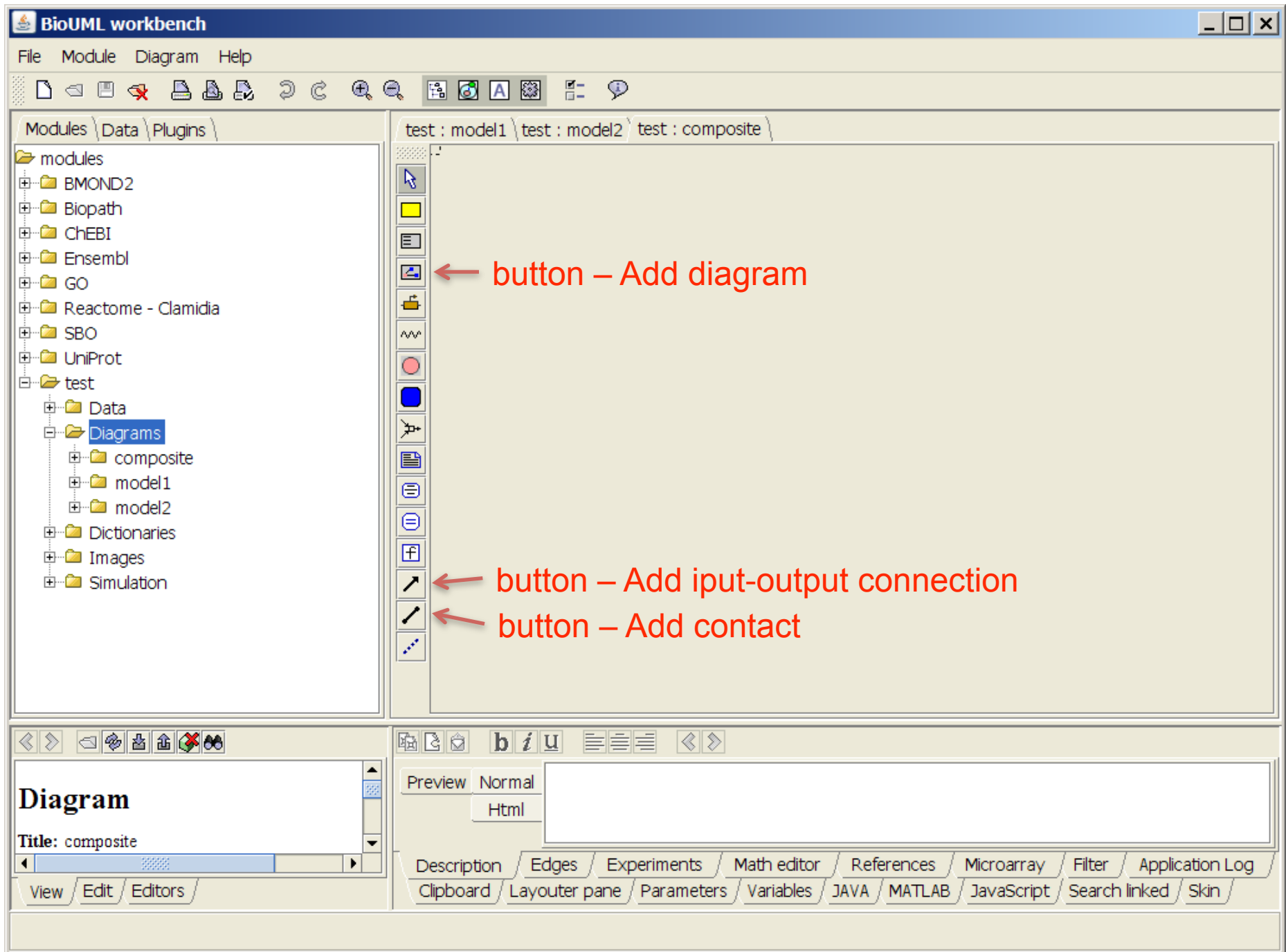
Diagram

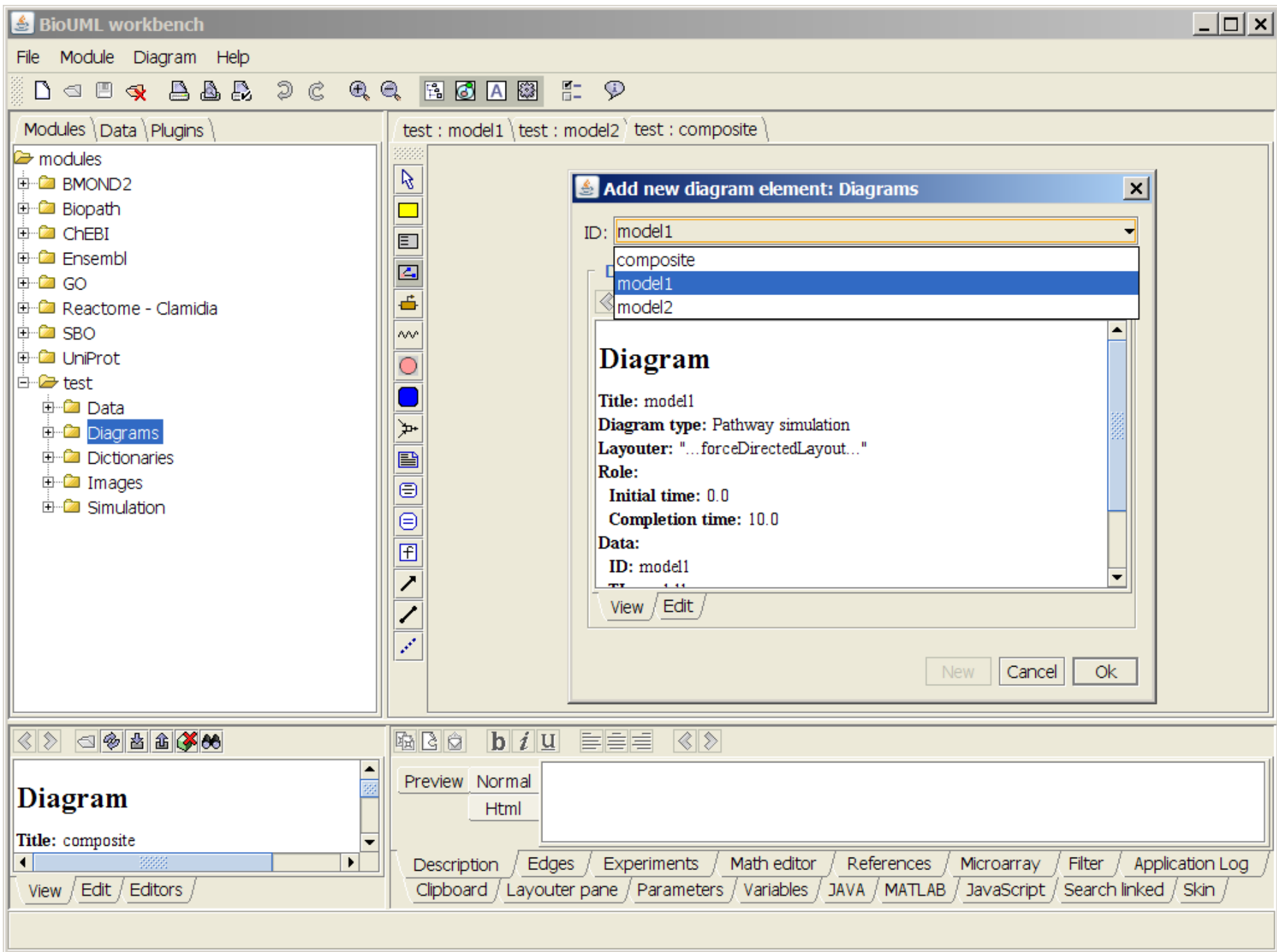
Title: model2
Diagram type: Pathway simulation
Layouter: "...forceDirectedLayout..."
Role:
Initial time: 0.0
Completion time: 10.0

View Edit Editors

INFO : Communicating with update site:<http://server2.biouml.org/update/> ...
ERROR : Network connection problems encountered during search.

Description Edges Experiments Math editor References Microarray Filter Application Log
Clipboard Layouter pane Parameters Variables JAVA MATLAB JavaScript Search linked Skin

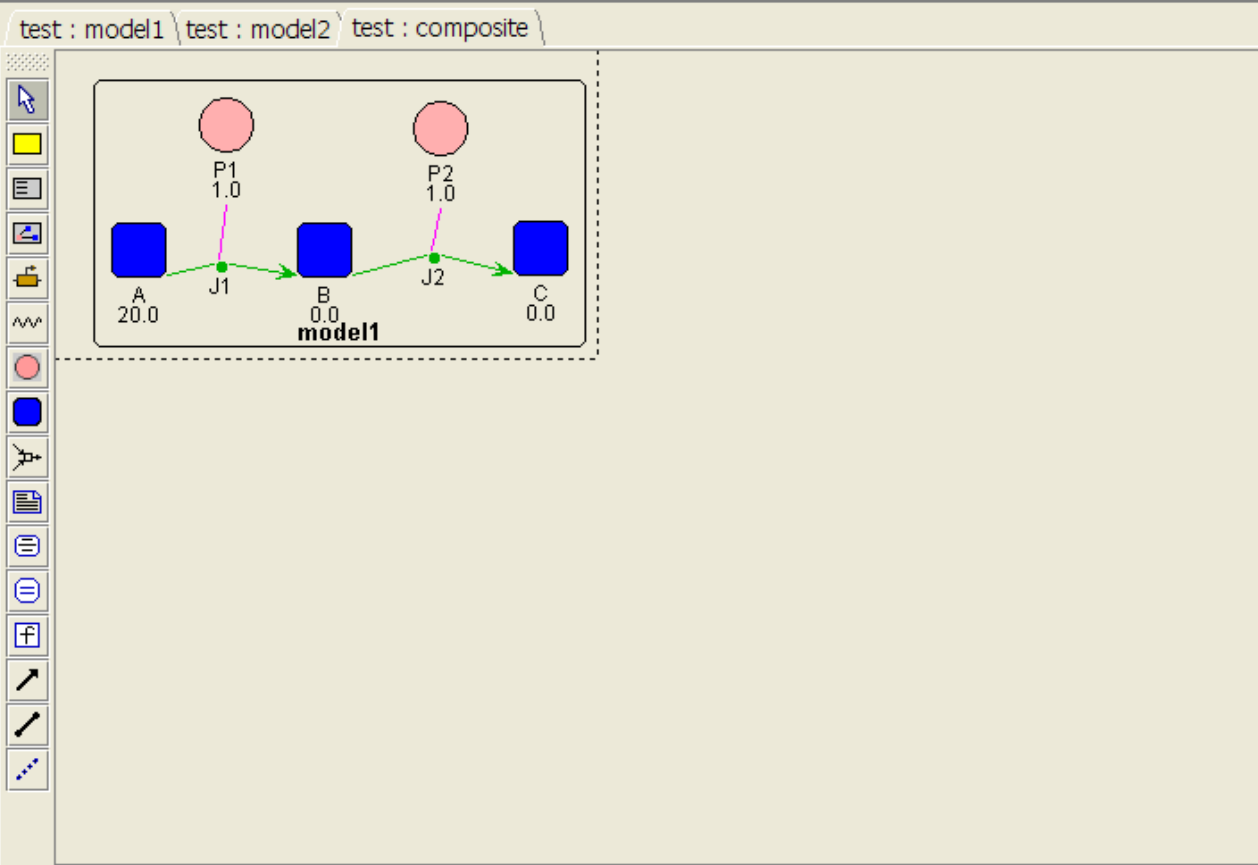






Modules \ Data \ Plugins \

- modules
 - BMOND2
 - Biopath
 - ChEBI
 - Ensembl
 - GO
 - Reactome - Clamidia
 - SBO
 - UniProt
 - test
 - Data
 - Diagrams
 - Dictionaries
 - Images
 - Simulation



Diagram

Title: model1

View Edit Editors



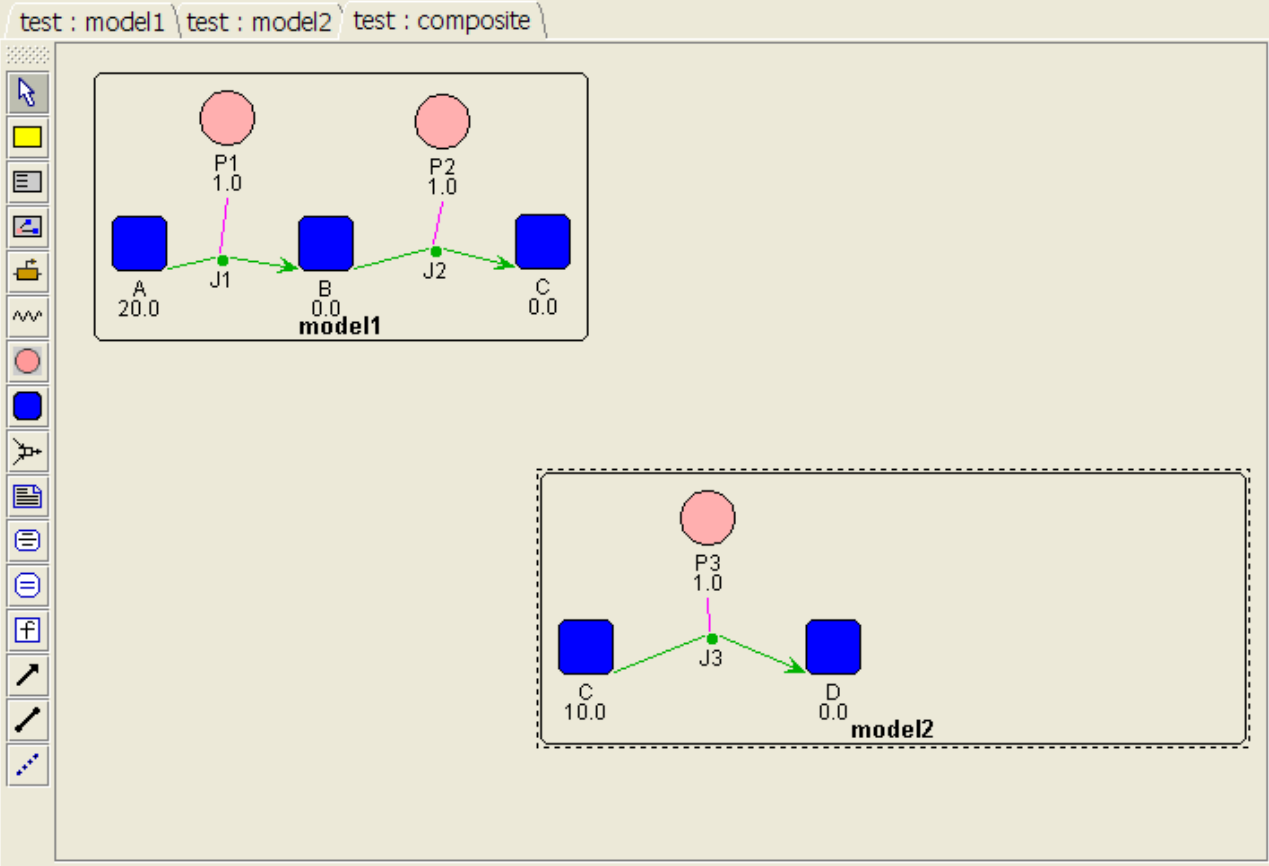
Preview Normal
Html

Description Edges Experiments Math editor References Microarray Filter Application Log
Clipboard Layouter pane Parameters Variables JAVA MATLAB JavaScript Search linked Skin



Modules \ Data \ Plugins \

- modules
 - BMOND2
 - Biopath
 - ChEBI
 - Ensembl
 - GO
 - Reactome - Clamidia
 - SBO
 - UniProt
 - test
 - Data
 - Diagrams
 - Dictionaries
 - Images
 - Simulation



Diagram

Title: model2

View Edit Editors



Preview Normal

Html

Description Edges Experiments Math editor References Microarray Filter Application Log

Clipboard Layouter pane Parameters Variables JAVA MATLAB JavaScript Search linked Skin

BioUML workbench File Module Diagram Help

Modules \ Data \ Plugins \

- modules
 - BMOND2
 - Biopath
 - ChEBI
 - Ensembl
 - GO
 - Reactome - Clamidia
 - SBO
 - UniProt
 - test
 - Data
 - D
 - Ir
 - S

test : model1 | test : model2 | test : composite

```

    graph LR
      A[A 20.0] --> J1((J1))
      J1 --> B[B 0.0]
      B --> J2((J2))
      J2 --> C[C 0.0]
      P1((P1 1.0)) --- J1
      P2((P2 1.0)) --- J2
  
```

model1

```

    graph LR
      C[C 10.0] --> J3((J3))
      J3 --> D[D 0.0]
      P3((P3 1.0)) --- J3
  
```

model2

New connection

In: model1.SBS000025

Out: model2.SBS000025

Specify 'in' and 'out' nodes.
For this purpose click by mouse under the corresponding node on the diagram.

Cancel Ok

Node

Title: C

View Edit Editors

Preview Normal Html

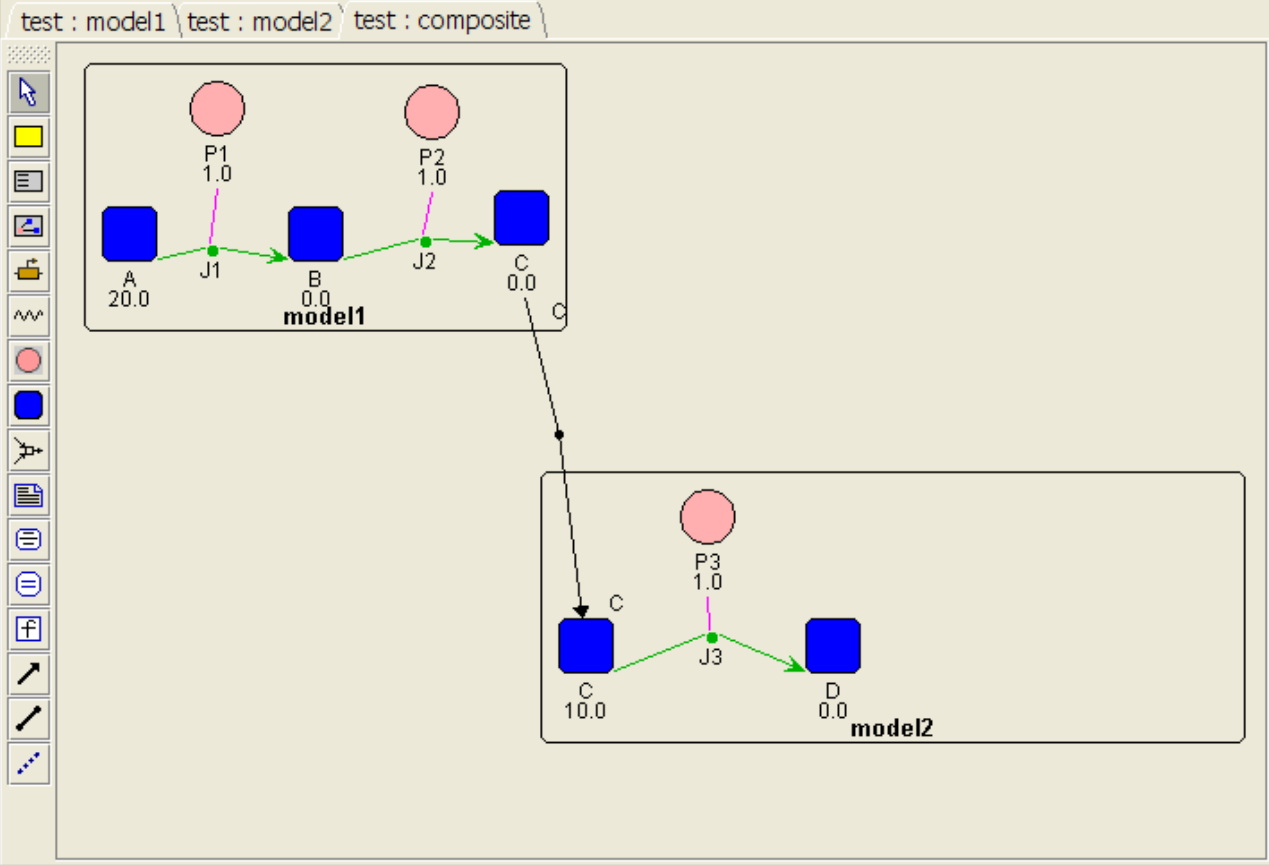
Description Edges Experiments Math editor References Microarray Filter Application Log

Clipboard Layouter pane Parameters Variables JAVA MATLAB JavaScript Search linked Skin



Modules \ Data \ Plugins \

- modules
 - BMOND2
 - Biopath
 - CHEBI
 - Ensembl
 - GO
 - Reactome - Clamidia
 - SBO
 - UniProt
 - test
 - Data
 - Diagrams
 - Dictionaries
 - Images
 - Simulation



Edge

Title: CONNECTION0003

View | Edit | Editors



Preview | Normal | Html

Description | Edges | Experiments | Math editor | References | Microarray | Filter | Application Log

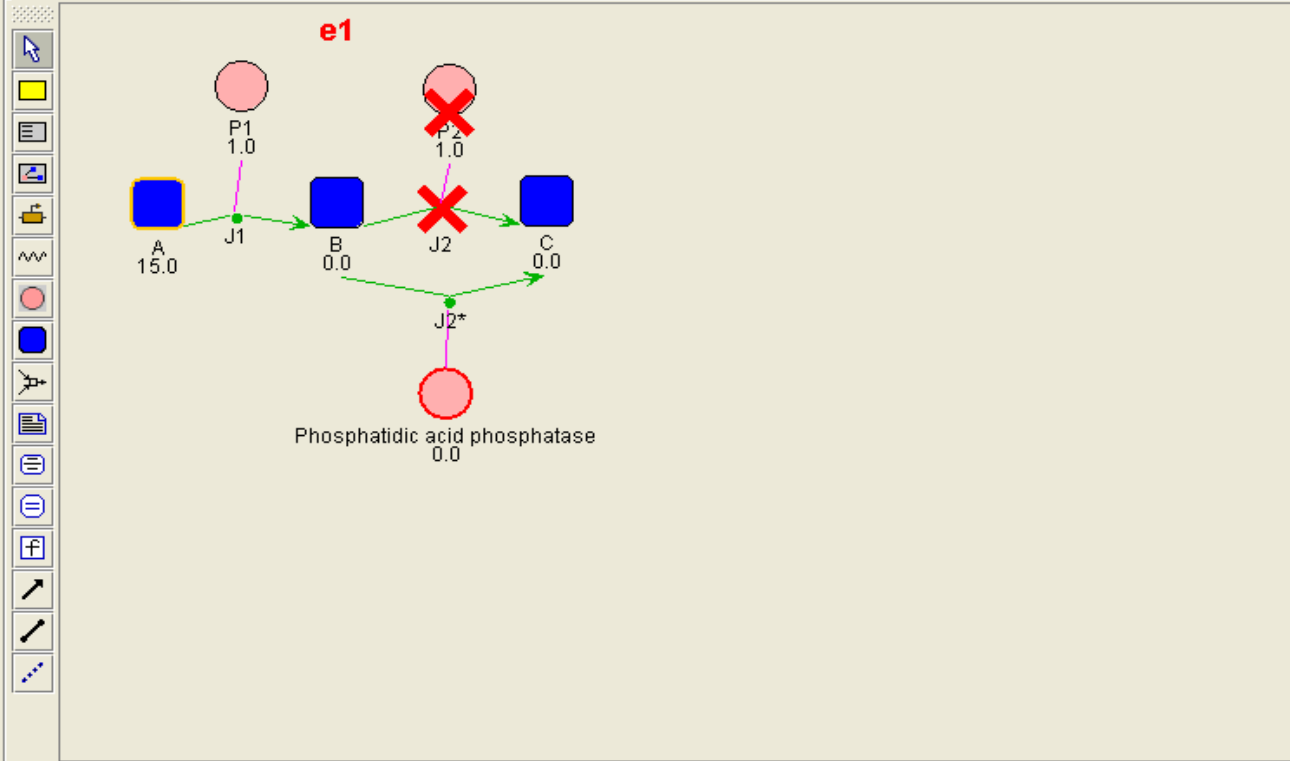
Clipboard | Layouter pane | Parameters | Variables | JAVA | MATLAB | JavaScript | Search linked | Skin



Modules \ Data \ Plugins \

- modules
- ├─ BMOND2
- ├─ Biopath
- ├─ ChEBI
- ├─ Ensembl
- ├─ GO
- ├─ Reactome - Clamidia
- ├─ SBO
- ├─ UniProt
- └─ test
 - ├─ Data
 - ├─ **Diagrams**
 - ├─ Dictionaries
 - ├─ Images
 - └─ Simulation

test : model1 \ test : model2 \ test : composite \



Property	Value
Node	
Title	A
Comment	
Size	
Role	
Name	\$SBS000023

Experiments: e1

Changes	Identifier	Type	Comment
0	RCT000037: PRT000023 as m...	removed	
1	RCT000037: SBS000024 as re...	removed	
2	RCT000037: SBS000025 as pr...	removed	
3	RCT000037	removed	

View Edit Editors

Description Edges Experiments Math editor References Microarray Filter Application Log Clipboard
 Layouter pane Parameters Variables JAVA MATLAB JavaScript Search linked Skin