

Improving the JSBML/CellDesigner Plugin Interface

Ibrahim Vazirabad

20-8-2014



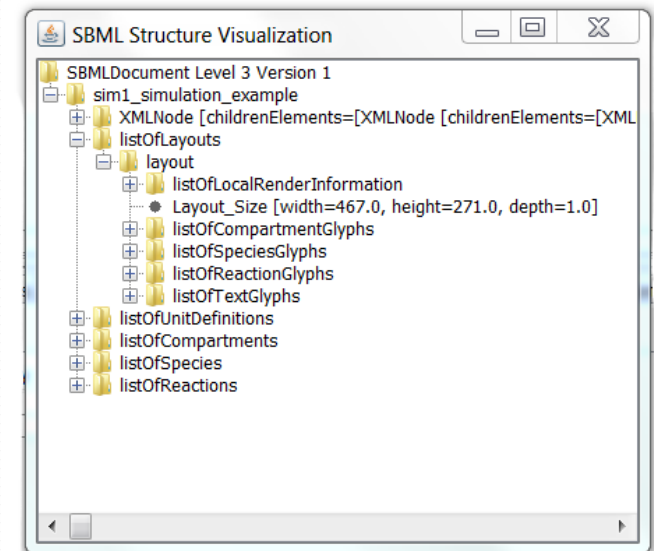
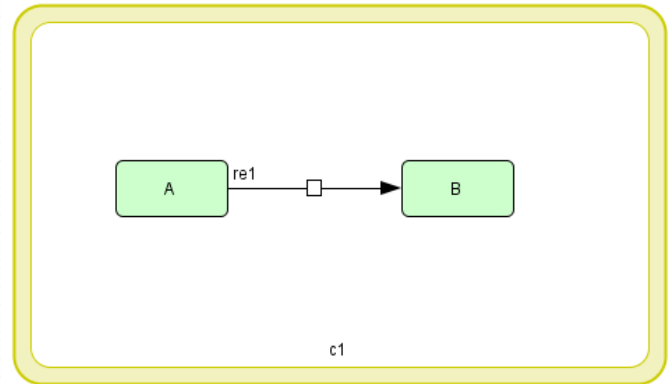
JSBML



O|B|F

What is JSBML?

- Java library to read/write/manipulate SBML files.
- Offers full platform independence.
- CellDesigner plugin interface.



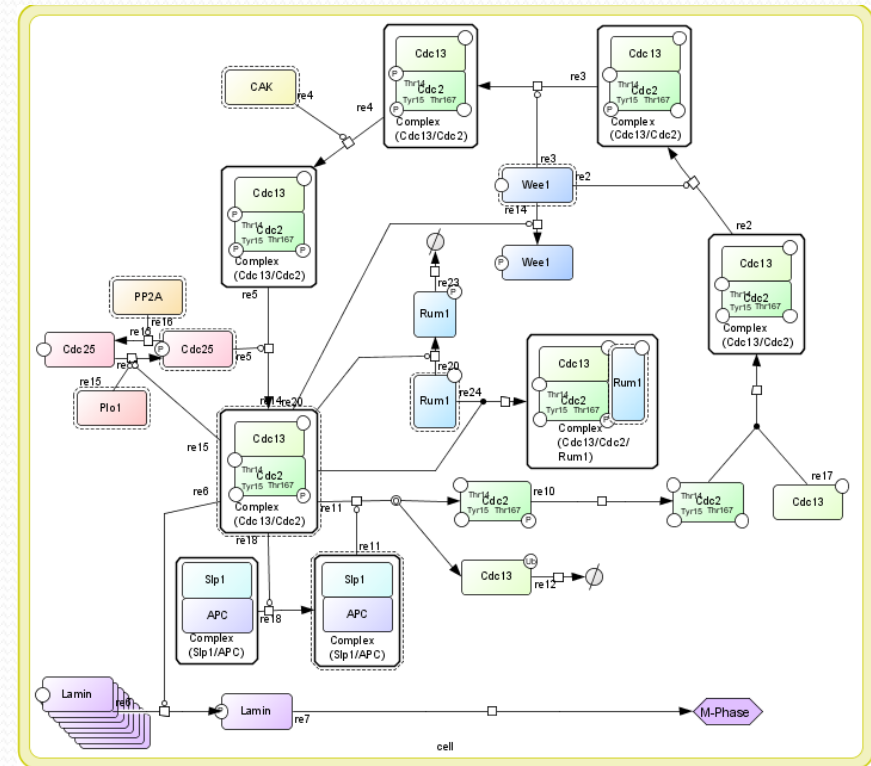
CellDesigner model and JSBML plugin

What is CellDesigner?

- CellDesigner is a process diagram editor for biochemical networks.
- One of most frequently used tools in systems biology.
- Accesses many SBML model databases: BioModels, PANTHER.



- Function augmentation: plugins
 - Essential for project.



-CellDesigner sample file: M_phase.xml
-PANTHER logo: <http://www.pantherdb.org/>
-BioModels logo: <http://www.ebi.ac.uk/biomodels-main/>

SBML Layout and Render Extension

- Pre-L3 SBML, no way to specify network layout or appearance.
- 2006: Layout and Render packages proposed.
 1. Layout: element position, size.
 2. Render: colors, shapes, line widths, etc.
- CellDesigner has own specification.
 - Would like to merge both SBML's and CellDesigner's specifications.

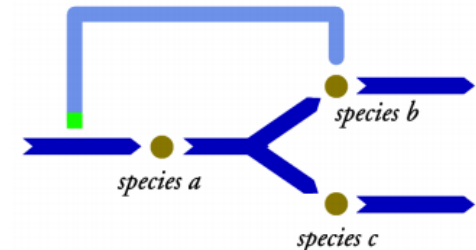
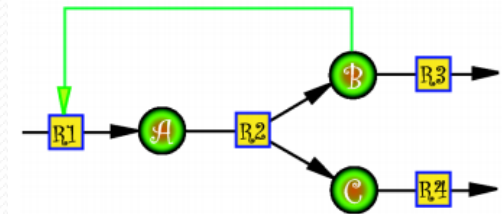
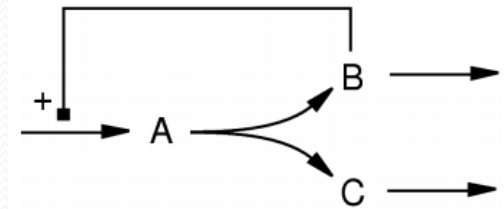


Figure 1: Examples of different renderings of the same layout.

-from SBML Level 3 Package: Layout specification

Project Motivation

- Why improve CellDesigner/JSBML plugin interface?
 1. JSBML users: harness model layout power of CellDesigner.
 2. CellDesigner users: access diverse suite of JSBML tools.
 3. Plugin developers: Interface can ease plugin development.

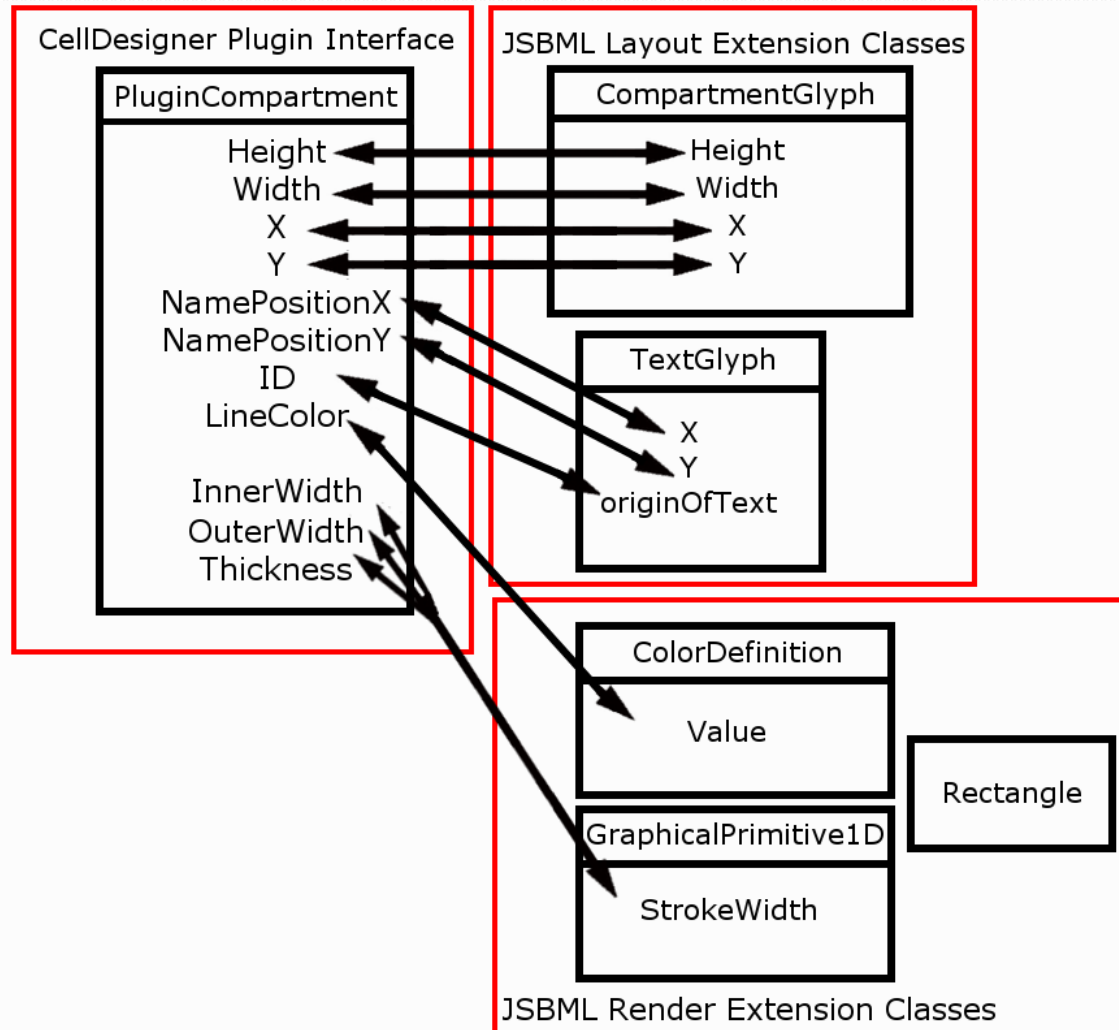
Project Overview

1. CellDesigner provides non-accessed layout/render information.
 - Want to import these attributes via Layout/Render.
2. CellDesigner/JSBML model synchronization was incomplete.
 1. JSBML notified CellDesigner of core Model changes.
 2. CellDesigner → JSBML unimplemented.
- When CellDesigner sends events, need to update JSBML Model.

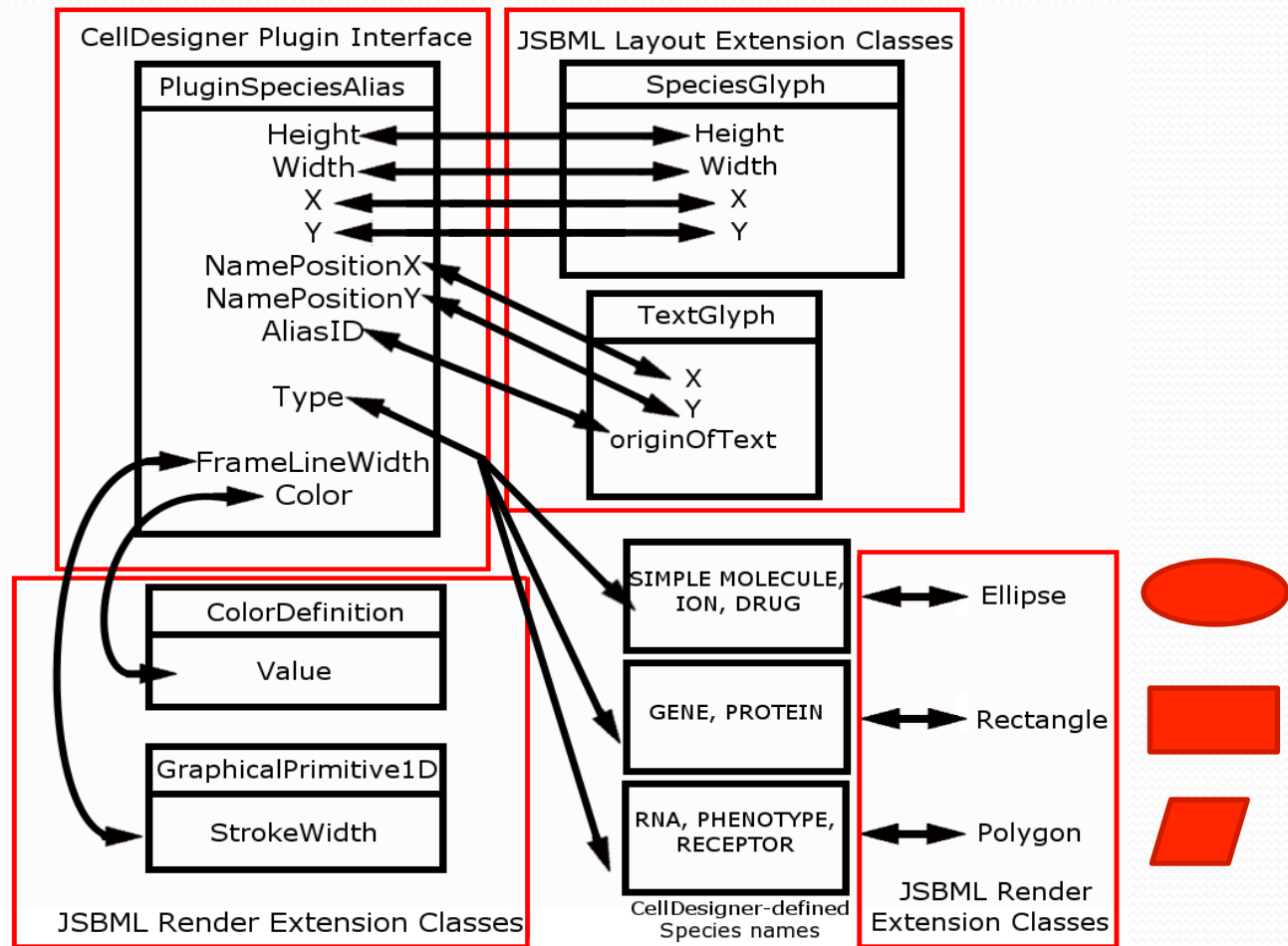
Project Goals

- Develop small collection of CellDesigner plugins.
 1. Visualize Layout/Render data additions from CellDesigner.
 2. Visualize synchronization from CellDesigner to JSBML.
- 1st step: map CellDesigner plugin interface to Layout/ Render.
 - PluginCompartment, PluginSpeciesAlias, PluginReaction

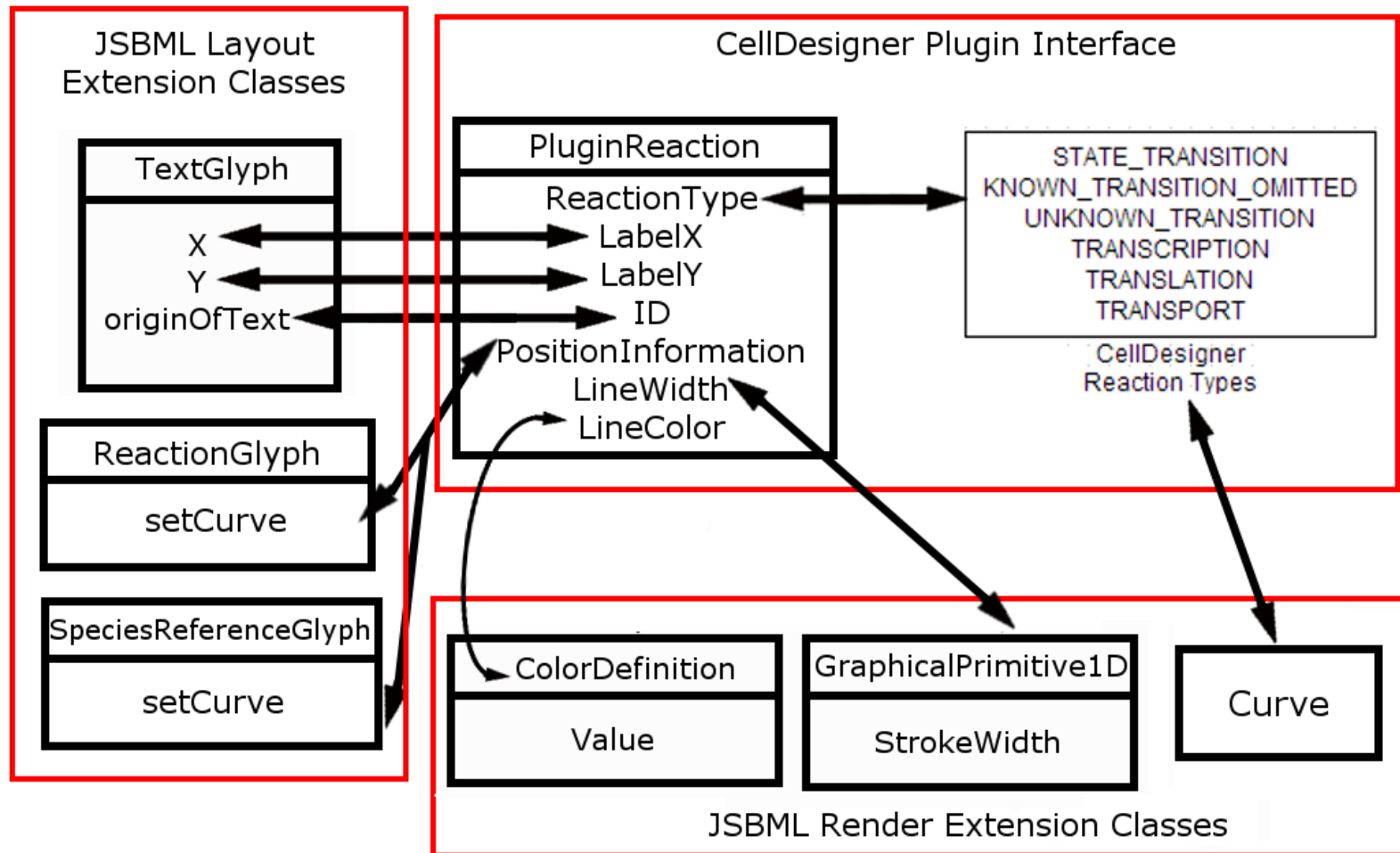
PluginCompartment-JSBML Mapping



PluginSpeciesAlias-JSBML Mapping

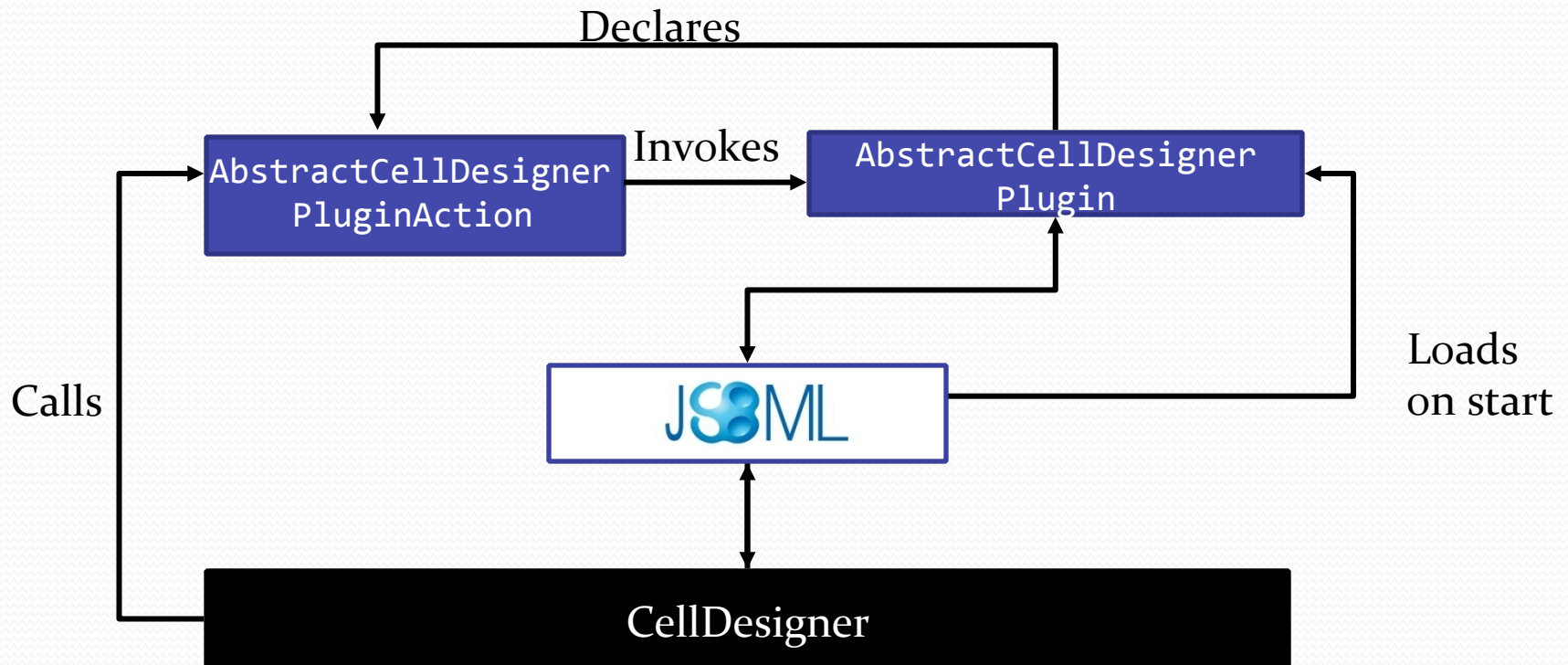


PluginReaction-JSBML Mapping



CellDesigner Plugin Creation

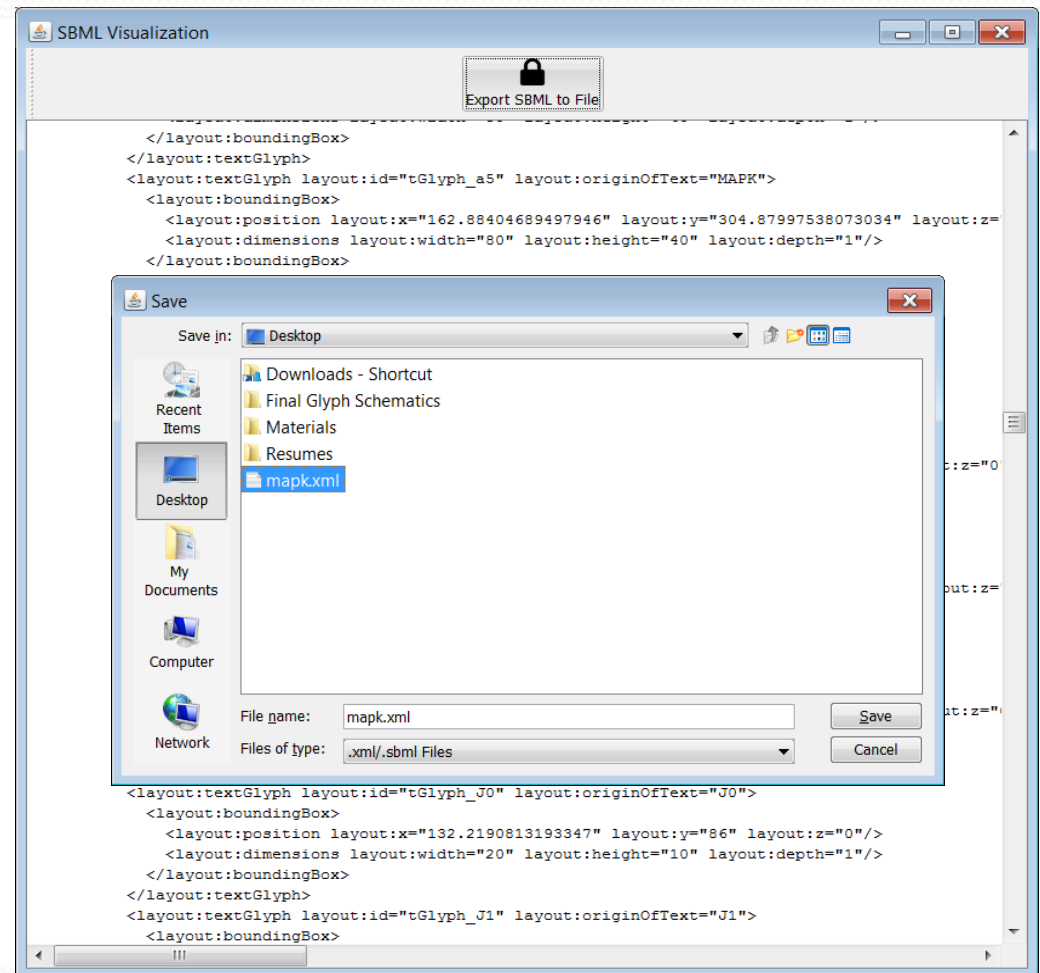
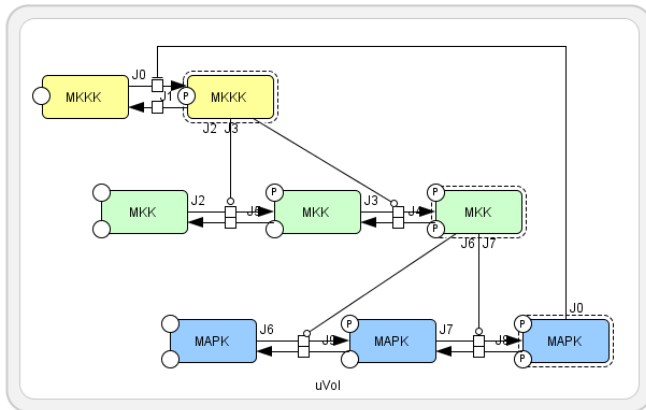
- To create plugin, implementation of only two abstract classes required.
- Once implemented, link formed between JSBML and CellDesigner.
- Easily make JSBML application into CellDesigner plugin.



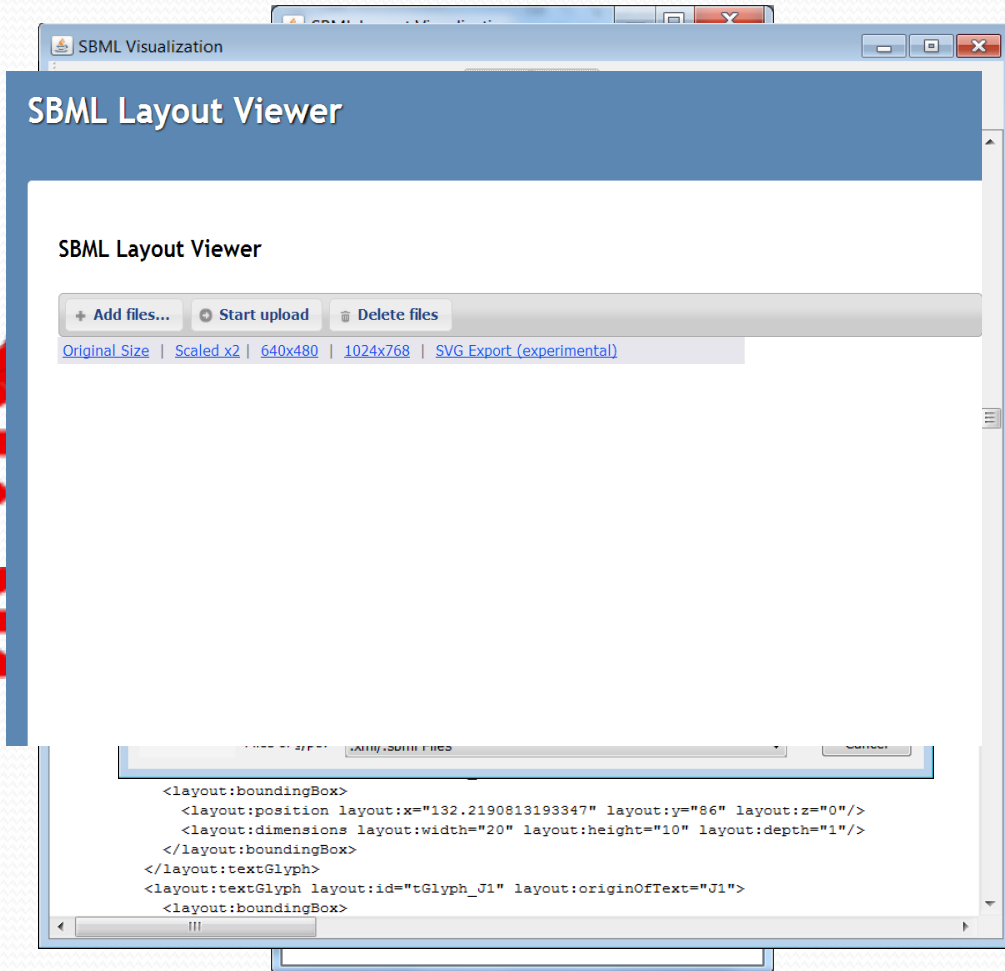
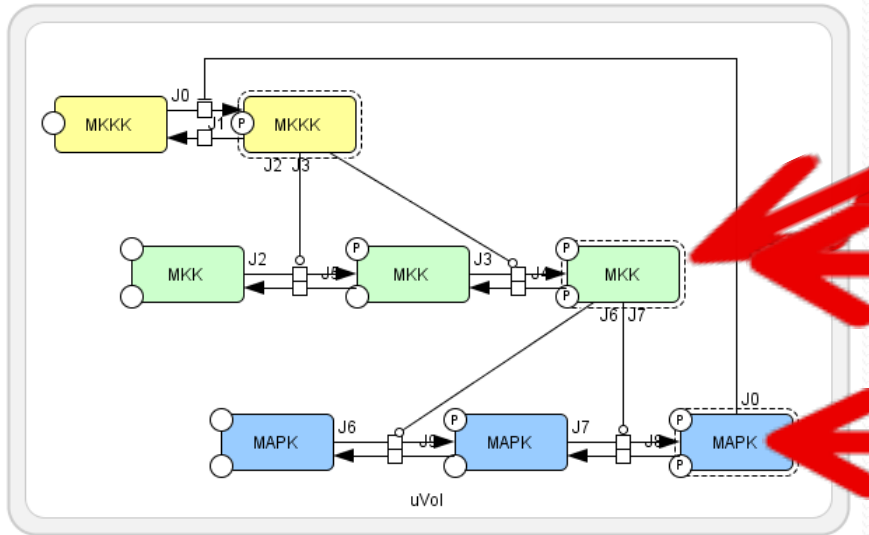
Project Progress

1. Imported CellDesigner Layout data into JSBML.
 1. SBML file export plugin.
 2. Tree-based Layout visualization plugin.
2. CellDesigner→JSBML synchronization completed.
 - Tree-based SBML visualization plugin.

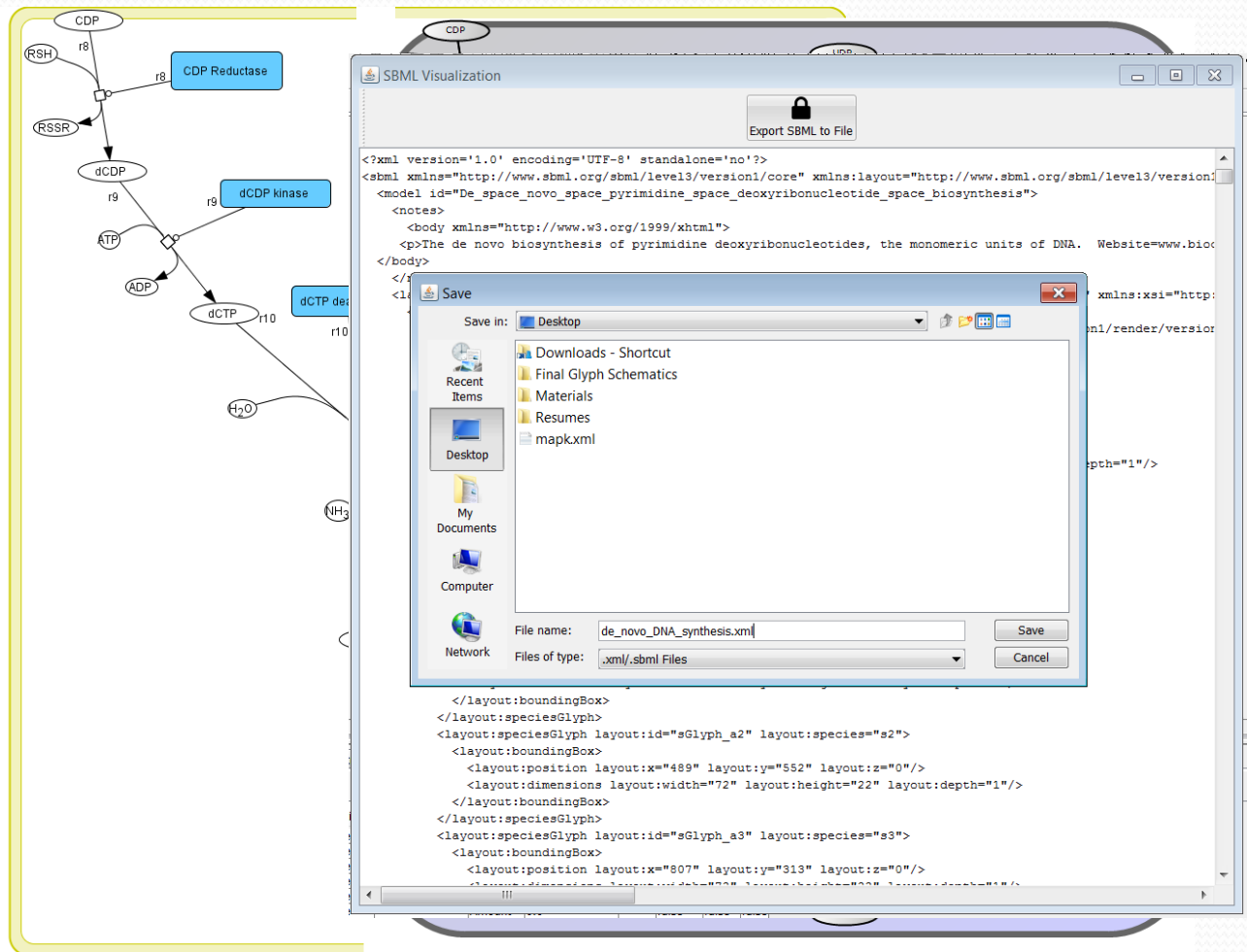
SBML File Export Plugin



Layout Visualization: MAPK Pathway

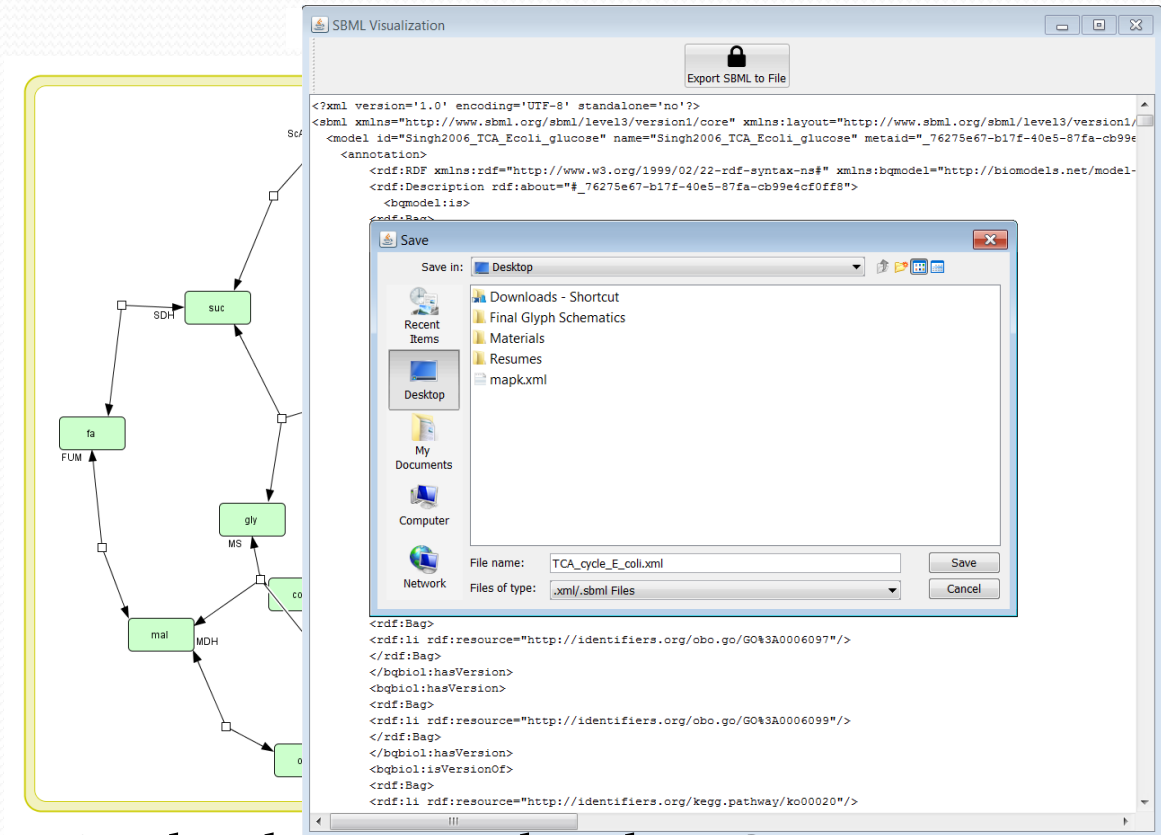


Layout Visualization: *De novo* nucleotide biosynthesis



-PANTHER Model: *De novo* pyrimidine deoxyribonucleotide biosynthesis
Visualized by SBML Layout Viewer
URL: sysbioapps.dyndns.org/Layout

Layout Visualization: *E. coli* TCA Cycle



Layout information has been translated to JSBML.

CellDesigner/JSBML Synchronization

- CellDesigner plugins sends notifications when:
 1. `SBaseAdded()`
 2. `SBaseChanged()`
 3. `SBaseDeleted()`
 4. `modelOpened()`
 5. `modelSelectChanged()`
 6. `modelClosed()`
- To understand further, created plugin that receives/prints events.

CellDesigner/JSBML Synchronization

SBML/CD Test

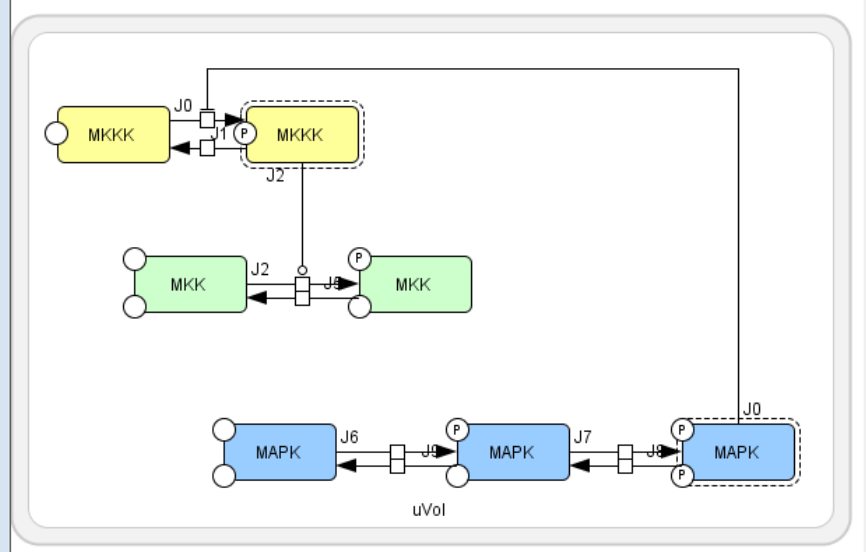
Model_SelectChanged: oscillating_MAPK
model_Opened: oscillating_MAPK
Model_SelectChanged: oscillating_MAPK
SBase_Changed: PluginSpeciesAlias: a4
SBase_Changed: PluginReaction: J3
SBase_Changed: PluginReaction: J4
SBase_Changed: PluginReaction: J6
SBase_Changed: PluginReaction: J7
SBase_Changed: PluginReaction: J7
SBase_Changed: PluginReaction: J6
SBase_Deleted: PluginReaction: J4
SBase_Deleted: PluginReaction: J3
SBase_Deleted: PluginReaction: J3
SBase_Deleted: PluginSpeciesAlias: a4
SBase_Deleted: PluginSpecies: MKK PP

Four reactions, one species.

J6, J7 changed, J3, J4 deleted in addition.

One user action, many events sent.
Update JSBML Model accordingly.

Clear Display



The diagram illustrates a signaling pathway with three levels of activation: MKKK (yellow), MKK (green), and MAPK (blue). Each level has an active state (represented by two circles) and an inactive state (represented by a circle with a 'P').

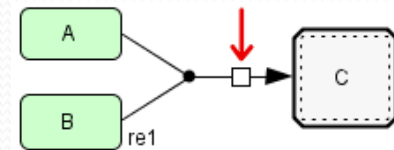
- MKKK Level (Yellow):** The inactive state (MKKK-P) is shown in a dashed box. It is activated from the inactive state of MKK (J2) and from the inactive state of MAPK (J0). The active state (MKKK) is activated from the inactive state (MKKK-P) via J1.
- MKK Level (Green):** The inactive state (MKK-P) is shown in a dashed box. It is activated from the inactive state of MAPK (J6) and from the inactive state of MKKK (J3). The active state (MKK) is activated from the inactive state (MKK-P) via J2.
- MAPK Level (Blue):** The inactive state (MAPK-P) is shown in a dashed box. It is activated from the inactive state of MKK (J8) and from the inactive state of MKKK (J3). The active state (MAPK) is activated from the inactive state (MAPK-P) via J7.

Activation is shown by arrows with labels J0, J1, J2, J3, J6, J7, J8. Inactive states are shown in dashed boxes. A uVol input is shown at the bottom.

The screenshot displays the 'SBML Structure Visualization' application window. The main area shows a hierarchical tree structure of an SBML document titled 'oscillating_MAPK'. The tree starts with 'XMLNode [childrenElements=[XMLNode [...]]]' at the root, followed by 'annotation', 'listOfLayouts', 'layout', 'listLocalRenderInformation' (containing 'Layout_Size'), 'listOfCompartmentGlyphs', 'listSpeciesGlyphs', and a series of 'speciesGlyph' entries. Below these are 'listReactionGlyphs' containing multiple 'reactionGlyph' entries, followed by 'listTextGlyphs', 'listUnitDefinitions', 'listCompartments', 'listSpecies' (listing MKK, MKKK, MAPK), and finally 'listReactions' (listing J0 through J9). Red arrows highlight specific parts of the tree: one points to the right side of the 'listReactionGlyphs' section, another points to the 'listReactions' folder, and a third points to the 'J4' reaction entry.

Next Steps/Further Work

1. Render Extension additions are not implemented.
 - a) JSBML bug preventing Render Extension from being implemented.
 - b) On Monday, bug report accepted.
2. Reaction positioning algorithm issues:
 1. Guesswork involved in finding the process node.
 2. Updated CellDesigner plugin interface released on Sunday.
3. Synchronize Layout Extension changes from JSBML to CellDesigner.



Project Discussion

- Once completed, hope that it will make the interface between CellDesigner and JSBML very fluid.
- Will provide a way to communicate between generic functionality of JSBML and visualization function of CellDesigner.
- With improvements, it will be a polished addition to current JSBML-CellDesigner bridge.

Acknowledgements

Mentors:



Dr. Andreas Dräger
Alex Thomas



Dr. Akira Funahashi



Funding: **Google**

Thank you!
Questions?