Notes from SED-ML related discussion at HARMONY 2017

SED-ML Current Status

Mostly carrying on the discussions from COMBINE 2016, see the SED-ML discussions @ COMBINE 2016 document for reference.
PDF of L1V3 draft

Where is the "specification"?

- Currently the spec is LaTeX in a subversion repository hosted on Sourceforge.
- Suggested to move to an Overleaf project to enable easier editing, everyone thinks thats a good idea.
- Sid: Here is a not too complex template overleaf document:
  - https://www.overleaf.com/read/jfhrrdppqjtr
  - note that if you have 100+ of PDF figures you will hit the freemium limit on overleaf, just FYI you should probably use all PNG images

Styling changes from COMBINE discussion

- Implemented in Frank's SED-ML C# tools
- Not yet "proposed" for addition to SED-ML specification
- Discussion about Deviser and how that could be used to define the specification and libSEDML (C++)
  - Developed to support rapid creation of SBML packages, but starting to support non-package specification

Propose to make changes for plot styling to the specification (L1V3 draft) and send that round for approval.

Set date and time during Harmony 2017 to edit the spec.

- **Tuesday afternoon, 9am Wednesday (if required): room N230**

Legends

- Decided to include a boolean flag that simply indicates whether a legend should be shown for a plot.
- Defer full specification of location, styling, etc. until future version.

Test suite

- https://github.com/luciansmith/sedml-test-suite/
- Move to a system based on COMBINE archives, include modes, SED-ML, “correct” outputs.
- Use SED-ML that only uses reports, and then compare generated reports to “correct” results
- Could define SED-ML which describes simulation experiments which perform the comparison (tolerance on comparison), or some kind of settings file as per the SBML test suite.
- Move test suite under the SED-ML github organisation
    - A “source” repository that has all the raw files in a folder-based system
    - Then generate “releases” that create COMBINE archives for each test and make available for distribution.
- Make use of SBML test suite methods for generating tags, components, so that tests can be filtered. Generated “info” file.

Future Version

Set values from distributions

Would be good in future SED-ML to be able to set values using a statistical distribution. Could look to SBML-distrib for guidance.

- Need to be able to measure stochastic reproducibility - mean, variance, etc…

Legends

- Allow the legend position to be specified, within and outside the axis bounding box

Additional Graphic Decorations

- Ability to draw shading between two curves, eg for envelopes
- A new type of curve/data generator
- Ability to specify subplots, eg grid of smaller plots.

Subplots

- Need to specify the layout of plots
- Can define all the plots as per normal, but need the layout in order to reproduce figures from papers.

Action Items

- ~~Frank: move spec temporarily to L1V3 to overleaf~~

    - Overleaf link: https://www.overleaf.com/10141340wptfdtmtcmgp#/37437737/
- Herbert: update spec with curve styling

Older Google Docs

Parameter estimation:
https://docs.google.com/document/d/1rrs0fYuKFr4fgL1b7eGwSnaLhRPW6NdXwAaJY0ZN_WY/edit#heading=h.61sixvmnpg15

Harmony 2012:
https://docs.google.com/document/d/1ifqQy1QJoAYXGtxI7LElHFie6_kP8sXwSIBGTiG5_as/edit

Harmony 2015 / 2016:
https://docs.google.com/document/d/1dgm6hGCK8MbxlrkXY-fdU1Y9UTph4G7WQ_Xjhq6HipU/edit#heading=h.pio9gf5mv6uo

COMBINE 2016:
https://docs.google.com/document/d/1uNNkjATFUQ78FspKkBTwJ2UmevmCGfovaKI22sr3VF8/edit#

Data

SED-ML L1V3 draft: http://co.mbine.org/standards/sed-ml/level-1/version-3/draft-1

Current tool support:

- C# libSEDML has full support, including accessing data values/slices, for data in NuML documents and CSV.
- C++ libSEDML includes the SED-ML elements for data, but doesn't provide access to the data values.

Discussion regarding use of NuML as a mapping/wrapping layer around other data file formats vs further extension of SED-ML to include that support natively. Thought that getting NuML and/or CSV support is sufficient for now, can look at further formats as the need arises. Currently L1V3 draft should work with other data file formats (including HDF5 or Excel).

- Possible recommendation: A tool supporting SED-ML L1V3 should support at least CSV files, other formats are optional
- Character encoding can be an issue, especially when generating CSV from Excel files. As could decimal separator.
- How much to specify in the specification?

Use cases we want to support with data in SED-ML L1V3

- Setting parameter values/initial conditions from a data source
- Plotting data alongside simulation results
- Source for comparison simulations (e.g., testing simulations give the correct results in the test suite).

Future versions

- Parameter estimation, fitting

Notes from looking at the specification changes.

- Marker and Line shouldn't be primitive types?
    - Yep, they'll be children on the curve class.
- listOfMarkers/Lines and reference them from curves?
    - No, just use marker and line directly as children of curves

Further spec changes discussed and added to the Overleaf specification

- Introduce abstract plot class to hold common plot properties (axes, ranges, aspect ratio, etc)
- Introduce axis class to hold properties for axis instead of many many new attributes on plot class
- Defining primitive types and referring to them where needed.
- Add id and name to SEDBase
- Error bars and envelope plots
    - Only for curves, not surfaces
    - New ShadedArea class derived from curve which adds two new data generators for the two y values, then shade the area between the two curves. Also need a fill child to define the shading.
    - New ErrorBars class derived from curve which adds four new data generators for x and y upper/lower. And booleans for whether the error bars have the T or just a I
    - Now ErrorBars merged into Curve as optional attributes.
- Create new AbstractCurve class from which Curve and ShadedArea are derived (to avoid overloading the yDataReference with a new meaning in ShadedArea). ErrorBars derived from Curve.
- AbstractCurve has optional Marker and Line children, if they are not present then behaviour is undefined. Markers and Lines with type "none" should be used to explicitly specify no marker/line for a curve.

Lucian's new figures, with proposed changes are in:
https://drive.google.com/drive/folders/0B71r5kLs3FlibnZMUEpIU1RKR1k

Back to L1V3

- Given the large number of changes coming out of HARMONY for the SED-ML specification, we talked about maybe keeping the L1V3 specification as it is and getting that released once we have two implementations.
- Tellurium almost has support for data, so if that gets finished we can test interoperability with that and C# libSEDML.
- We can then release L1V3 without all the changes discussed above, going back to only the addition of the data constructs.
- The changes discussed above then move into L1V4, which potentially comes “soon”

New tasks - eigenvalues, jacobians, sensitivities...

- Talked about how to get things like eigenvalues, jacobian, etc.
- Decided that adding new tasks for these is the way to go: https://docs.google.com/drawings/d/1AvL5kGmCjYqw757a07A2Osc_9vZNVd7ySG3KqhhCX1I
- These tasks will make various vectors and matrices available to the data generators that which have variables which reference the tasks.
- Will probably need new symbols and/or MathML in order to be able to pull out and use the appropriate data. Will leave this until such time as implementers start doing things with these tasks.