

XMILE's

Display and Interface: Gaining insights from models



Billy Schoenberg

Software Engineer

isee systems



Gary Baxter

MEMIC



Today's Agenda

- Introduce Display and Interface objects in two popular modeling tools.
- Highlight significant benefits to modelers of having standard Display and Interface capabilities
- Review and Introduction to the XMILE standard
- Technical discussion of the Display and Interface portions of the XMILE standard
 - XMILE Views
 - Linking model objects and display objects
 - The cascading style system
 - Representation of stock and flow diagrams
 - Review available interface-type objects



Display Objects

- All visual objects representing model equations in a stock and flow diagram.
- Stocks, Flows, Auxiliaries, and Connectors.
- Other visual objects – groups, ghosts, etc.



Display Objects

- Nothing unique about how they are implemented – basically the same look and feel.
- Non-standard set of attributes available for customization.



Interface Objects

- All those visual objects that allow the modeler or customer to interact with the model.
- Buttons, graphs, and input sliders.
- Other visual objects – tables, switches, etc.



Interface Objects

- Nothing unique about how they are implemented – basically the same look and feel.
- Non-standard set of attributes available for customization.



Benefit to Modelers

- Current situation
 - Display and Interface Objects are similar but do not share a standard set of attributes.
 - Two separate models in two separate file formats.
 - One modeling package for duration of the modeling project.
 - Limited to that package's features and functionality.



Benefit to Modelers

- Future opportunity
 - Common file format and a basic “core” set of capabilities.
 - Have a truly portable model.
 - All the features and functionality of all the packages are now available.
 - An opportunity to produce better models.
 - Easier consumption of models by other applications.



Do Your Part

- Ask your vendors when they plan to adopt XMILE.
- Here's the answer I'm looking for:

“We are committed to implement XMILE in our first major release after formal adoption of the standard.”



Today's Agenda

- ✔ • Introduce Display and Interface objects in two popular modeling tools.
- ✔ • Highlight significant benefits to modelers of having standard Display and Interface capabilities
- Review and Introduction to the XMILE standard
- Technical discussion of the Display and Interface portions of the XMILE standard
 - XMILE Views
 - Linking model objects and display objects
 - The cascading style system
 - Representation of stock and flow diagrams
 - Review available interface-type objects



What is the XMILE standard?

- Open standard and file format for system dynamics models.
- Developed by OASIS Technical Committee (formed in June 2013) composed of software vendors and individual practitioners.
- Builds on more than a decade of discussion in the SD community, and specifically a draft standard published by Karim Chichakly in 2007.
- Draft to be released July 2014, with final release Fall 2014.

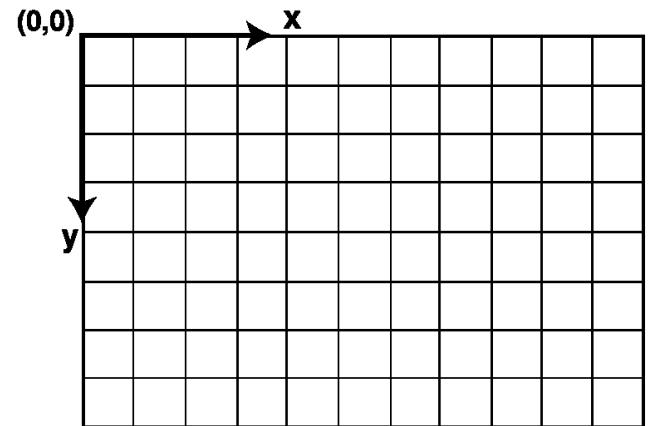


Technical Discussion of the Specification



The XMILE View

- Container for all XMILE display objects
- Optional types of views – stock and flow, interface, popup and vendor specific
 - Only stock and flow views can contain the canonical representation of XMILE variables
- Size (width, height) <double>
- Zoom <double>
- Background color or image



XMILE view assumptions and attributes

- All objects in a view must have
 - Position (x,y) <double>
 - Size (width,height) <double> or a child Shape tag
- Stocks cannot be represented using a circle
- Auxiliaries and Flows cannot be represented using a rectangle **UNLESS** they contain a macro/function with an implicit stock
- The shape tag allows for the specification of arbitrary shapes for any display object (pursuant to the above 2 rules)
- Shapes have a required type of rectangle, circle, name_only or vendor specific.
- Rectangle shapes require width, height <double> and optional corner-radius <double> (default: 0)
- Circle shapes require radius <double>
- Name only shape is a marker – no required attributes



Referencing variables in an XMILE view

- Any object appearing in the <variables> tag should have a related <stock|flow|aux|module|group> tag in a <view> tag
- A tag within a <view> is linked to its corresponding tag in the <variables> tag via its name

```
<xmile>
  <model>
    <variables>
      <stock name="My_stock".../>
    </variables>
    <views>
      <view>
        <stock name="My_stock" x="100" y="100"
          width="35" height="25"/>
      </view>
    </views>
  </model>
</xmile>
```

Important: The first encountered instance of a variable's view is its canonical representation



What about objects without a name?

- Graphs, Tables, Sliders, Buttons etc don't have a name
- Their 'name' is their uid (unique identifier) <int>
 - Linearly increasing
 - Not stable across model IO operations
 - Unique per XMILE Model
- uid's allow us to refer to objects without a name



Ghosts, Shadow Variables, etc. are XMILE Aliases

- An alias is a symbol representing a “portal” to the display of another XMILE model object in the same view.
 - Keeps diagrams neat and prevents connectors from crossing
 - Aliases are allowed to have connectors leaving them, but never pointing to them

```
<alias uid="1" x="50" y="100">  
  <of>My_stock</of>  
</alias>
```

Doesn't have a name needs a UID

- uid - The ‘name’ for the alias
- x,y - Its position (all of its size/shape properties are inherited from the variable it represents by default)
- of - The name of the model entity which this alias represents. The model entity must be in the same model as the alias.



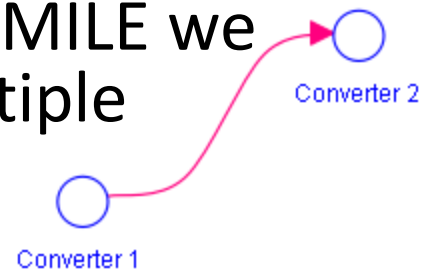
How do we describe the visual relationships between variables?

- The <connector> tag is used to describe the visual appearance of the relationships between XMILE model objects.
- A connector is an arrow which only ever appears between two display objects.

```
<connector uid="1" x="50" y="100" angle="32"/>  
  <from>  
    <alias uid="1">  
  </from>  
  <to>My_flow</to>  
</connector>
```

That's how we use a UID

- The typical connector is an arc, but with XMILE we can represent Bezier connectors with multiple points using the <pts> tag.



Commonly seen XMILE display object attributes

- **border-width**="thick | thin | <double>" – default=1px –thick=3px –thin=1px
- **border-color**="<hex code> | predefined color*"
- **border-style**="none | solid" –default=none
- **font-family**="<string>"
- **font-style**="normal | italic" – default=normal
- **font-weight**="normal | bold" – default=normal
- **text-decoration**="normal | underline" – default=normal
- **text-align**="left | right | center"
- **vertical-text-align**="top | bottom | center"
- **font-color**="<hex code> | predefined color*"
- **text-background**="<hex code> | predefined color*"
- **font-size**="<double>pt"
- **padding**="<comma separated list of no more than 4 doubles and no fewer than 1 double>"**
- **color**="<hex code> | predefined color*"
- **background**="<hex code> | predefined color*"
- **z-index**="<int>" –default=-1 (-1 is bottom-most, top-most is INT32_MAX ((1 << 31) – 1))



The cascading style system

- Styles refer to groups of XMILE objects depending on their location
- Styles affect the display of all related XMILE display objects

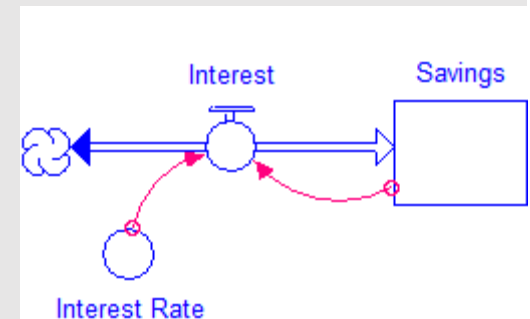
```
<style font-color="blue" font-size="10">  
  <stock font-weight="bold" font-color="black"/>  
  <flow font-size="12"/>  
</style>
```

- Any related stock object would have a bold black size 10 font
- Any related flow object would have a normal weight blue size 12 font
- A <style> tag may be placed in any of the following locations which changes the display objects it affects:
 - As a child of the root (<xmile>) tag
 - As a child of a <views> tag
 - As a child of a <view> tag



The representation of a stock and flow diagram

```
<model>
  <variables>
    <stock name="Savings"/>...
    <flow name="Interest"/>...
    <aux name="Interest_Rate"/>...
  </variables>
  <views>
    <style font-family="Arial" font-size="9" color="blue" font-color="blue"/>
    <view type="stock-flow" width="1152" height="1502" zoom="100">
      <stock name="Savings" x="237" y="109" width="35" height="25"/>
      <flow name="Interest" x="180" y="108" width="18" height="18">
        <pts>
          <pt x="146" y="108" />
          <pt x="215" y="108" />
        </pts>
      </flow>
      <aux name="Interest_Rate" x="106" y="134" width="18" height="18">
      <connector x="239" y="126" uid="0">
        <from>Savings</from>
        <to>Interest</to>
      </connector>
      <connector x="115" y="134" uid="1">
        <from>Interest_Rate</from>
        <to>Interest</to>
      </connector>
    </view>
  </views>
</model>
```

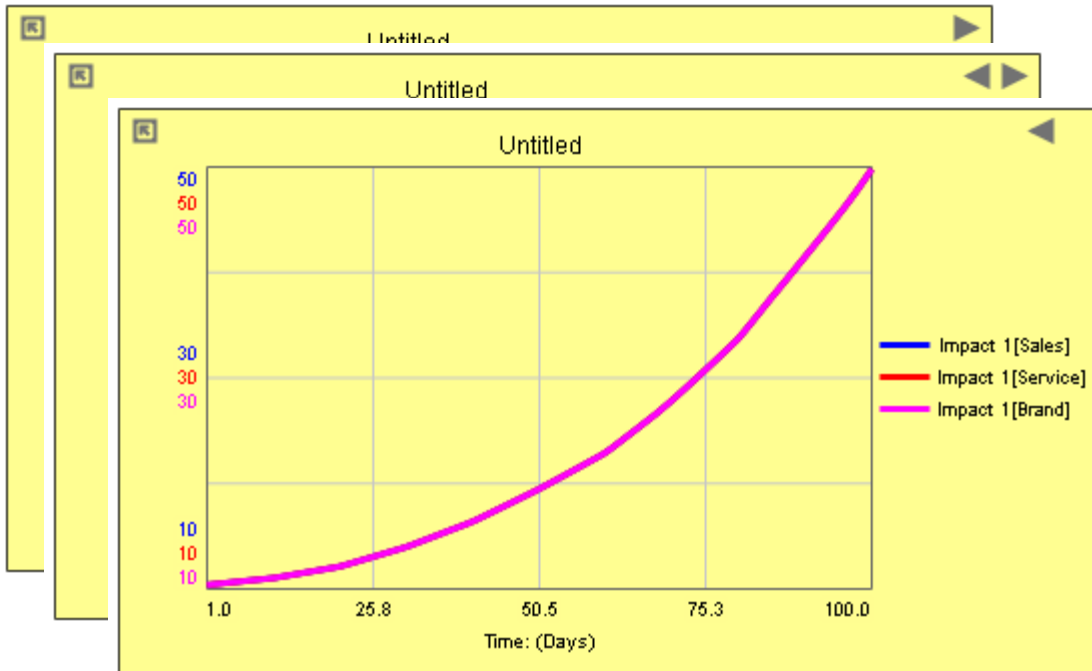


Available interface-type objects

- Stacked Containers
- Input Objects
 - Sliders/Knobs
 - Switches/Radio Buttons
 - Numeric Input/List Input Device
 - Graphical Input
- Output Objects
 - Numeric Displays
 - Lamps/Gauges
 - Graphs
 - Tables
- Annotations
 - Text Boxes
 - Graphics Frames (images video)
 - Buttons



Stacked Containers



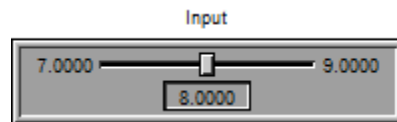
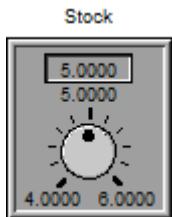
```
<stacked_container
visible-index="0"
...>
  <graph .../>
  <graph .../>
  <graph .../>
</stacked_container>
```

- Tables and Graphs can be placed in stacked containers to allow users to flip through results
- Tables and Graphs can also show comparative results from multiple runs

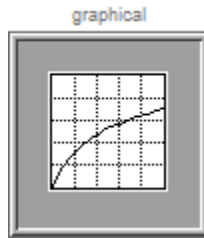


Input Objects

- Allow users to enter simulation parameters



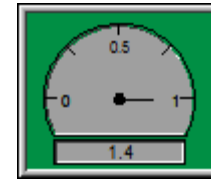
```
<slider ...>  
  <entity name="Input"/>  
  <reset_to>8</reset_to>  
</slider>
```



List Input Device ▼	
List Input	55



Output Objects



- These show simulation results, both instantaneous and over time
- Numeric Display, Lamp, Gauge are instantaneous
- Table, Graph* are over time

```
<table...>
  <item type="time" .../>
  <item ...>
    <entity name="Stock_1"/>
  </item>
  ...
</table>
```

	Stock 1	Flow 1	Converter 1
Feb	147.75	26.19	0.20
Mar	179.59	31.84	0.20
Apr	218.29	38.70	0.20
May	265.33	47.04	0.20
Jun	322.51	57.18	0.20
July	392.01	69.50	0.20
Aug	476.49	84.48	0.20
Sep	579.18	102.69	0.20
Oct	704.00	124.82	0.20
Nov	855.72	151.72	0.20
Dec	1,040.13	184.41	0.20

*Certain graphs are not over time



Annotations

- Used for putting pictures, videos, static text and buttons into simulations
- Buttons can perform a wide variety of simulation related actions
 - Navigation
 - Simulation actions (Run, Pause, Stop, Restore...)
 - Export/Import Data
 - Display information popup
 - Text
 - Image
 - Video



To learn more about XMILE...

XMILE overview webinar schedule:

- April 29: Introduction to XMILE
- May 20: Simulation Capabilities
- June 3: Display and Interface
- **June 24: Panel Discussion**
- July 21-23: Delft Conference
 - Round table discussion and ballot

Technical Committee information: www.oasis-open.org/committees/xmile/

Series videos are available at: www.youtube.com/user/XMILEtc

