XMILE's Simulation Capabilities: A Look Under the Hood

Will Glass-Husain

Co-Founder and Chief Software Architect

Forio Online Simulations

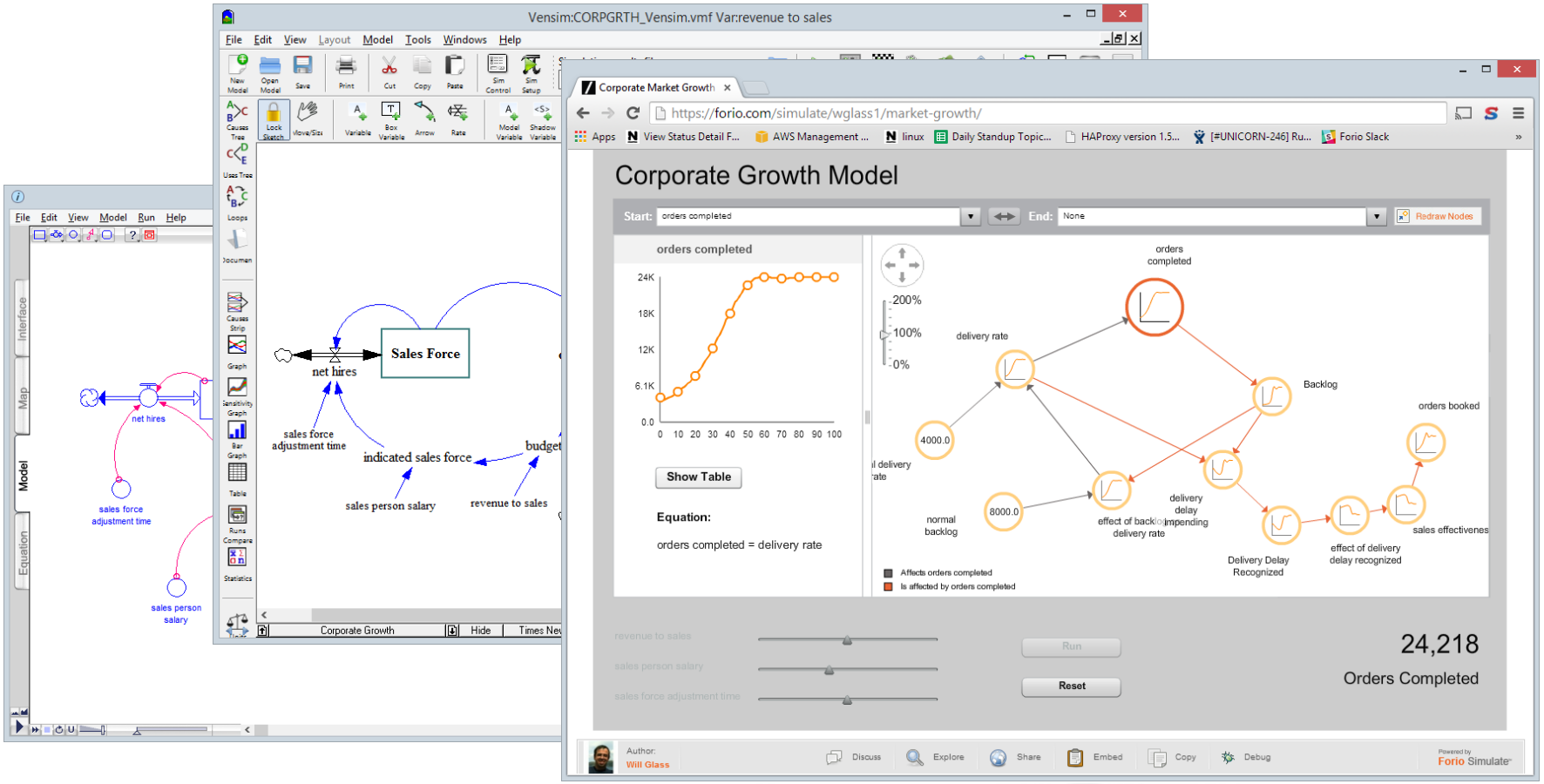
What is the XMILE standard?

- 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.

XMILE Technical Goals

- Core subset of common SD simulation software functionality
- Specify stock-flow diagrams (optional)
- Include interactive components (optional)
- Extensible in both representation and simulation behavior
- Small file size - human readable and editable
- Focused on system dynamics models. (floating point numbers only).

XMILE will promote innovation among users and software vendors.



isee systems OASIS IBM /forio System Dynamics Society VENTANA systems, inc. Vensim

In designing XMILE, we looked for the common features of SD simulation software.

- System dynamics constructs: stock/flow/auxiliaries
- Model diagrams: Display variables with connectors
- Computation: Euler's Method, RK4
- Functions: arithmetic, logical, financial, array
- Inputs: text boxes, slide bars, etc.
- Outputs: graphs, tables

We also considered the differences between SD software capabilities.

- Some major features may be present/absent
  - Text-only models
  - Pages with input and output widgets
  - Array capability
  - Units
- Software may contain specialized type of building blocks
  - example: iThink conveyors
- Differences in function definitions
  - int(1.5) returns 1, but does int(-1.5) return -1 or -2?
- Support for single or multiple models and single or multiple diagrams

XMILE contains these standard language components.

- Stocks, Flows, Auxiliaries
- Graphical Functions
- Groups
- Units
- Builtin Functions

XMILE contains these optional language components.

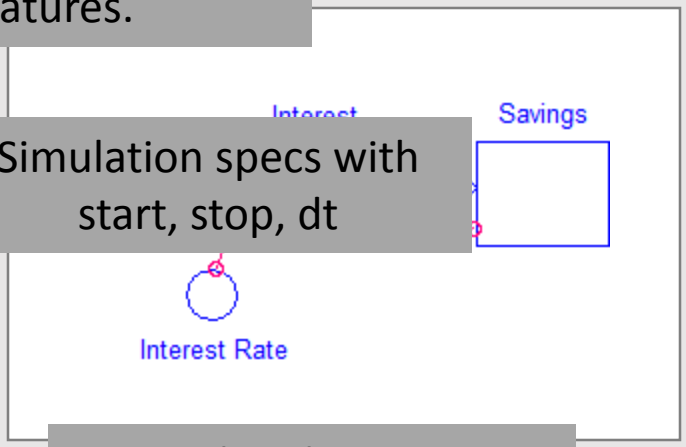
- Model diagrams
- Interface objects
- Arrays
- Submodels
- Event triggers
- Macros
- Conveyors
- Queues

Here's a "minimum" model in XMILE.

```
<xmile version="1.0" xmlns="http://www.s

  <header>
    <name>Bank Balance</name>
  </header>

  <sim_specs method="Euler">
    <start>0</start>
    <stop>36</stop>
    <dt>0.25</dt>
  </sim_specs>

  <model>
    <variables>
      <stock name="Savings">
        <eqn>1000</eqn>
        <inflow>Interest</inflow>
      </stock>
      <flow name="Interest">
        <eqn>Savings * Interest_Rate</eqn>
      </flow>
      <aux name="Interest_Rate">
        <eqn>.03</eqn>
      </aux>
    </variables>
  </model>
</xmile>
```

Header with model meta information and required software features.

Simulation specs with start, stop, dt



Stocks, Flows, Aux variables with equations

XMILE allows these top-level elements.

<header>	Required	information about the origin of the model and required capabilities.
<sim_specs>	Optional	default simulation specifications for this model
<model_units>	Optional	definitions of units used in this model.
<dimensions>	Optional	definitions of array dimensions specific to this model.
<behavior>	Optional	behaviors inherited/cascaded through all variables and models
<style>	Optional	Styles that are inherited/cascaded through all variables and models
<data>	Optional	persistent data import/export connections
<model>	1 or more	model equations and (optionally) diagrams.
<macro>	0 or more	macros that can be used in model equations

<header> and <options> specify the required capabilities to load the model. Not all software will support all features.

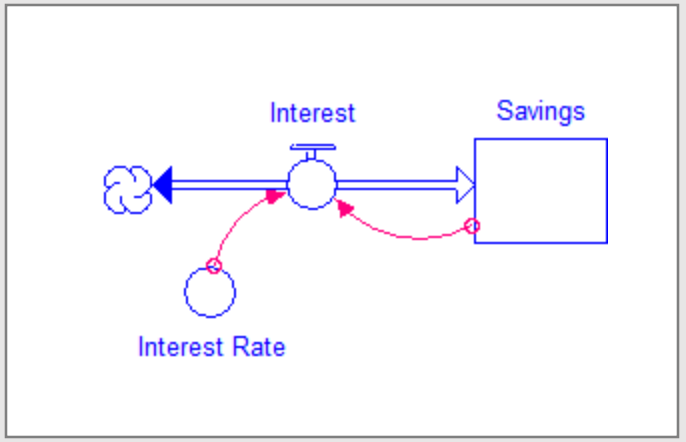
```
<header>
    <name>Bank Balance</name>
    <vendor>Forio</vendor>
    <options>
        <uses_conveyor/>
        <uses_queue/>
        <uses_arrays>N</uses_arrays>
        <uses_submodels/>
        <uses_macros/>
        <uses_event_posters/>
        <has_model_views/>
        <uses_outputs/>
        <uses_inputs/>
        <uses_annotation/>
  </options>
</header>
```

<model> splits up the equations and diagram with elements <variables> and <views>.

```
<model>
  <variables>
     <stock name="…">
       <eqn>…</eqn>
     </stock>
     <flow name="…">
       <eqn>…</eqn>
     </flow>
     <aux name="…">
       <eqn>…</eqn>
     </aux>
     …
  </variables>

  <views>
     <view width="…" height="…">
       <stock name="…" x="…" y="…" width="…" height="…"/>
       <aux name="…" x="…" y="…" width="…" height="…"/>
       …
    </view>
  </views>
</model>
```

XMILE's core building blocks are <stock>, <flow> and <aux>.

```
<model>
  <variables>

    <stock name="Savings">
      <eqn>1000</eqn>
      <inflow>Interest</inflow>
    </stock>

    <flow name="Interest">
      <eqn>Savings * Interest_Rate</eqn>
    </flow>

    <aux name="Interest_Rate">
      <eqn>.03</eqn>
    </aux>

  </variables>
</model>
```



Model conventions: Identifier syntax

- Roman letters (A-Z or a-z), underscore (_), dollar sign ($), digits (0-9), and Unicode characters above 127. Identifiers cannot begin with a digit or a dollar sign, and cannot begin or end with an underscore.
- Arbitrary UTF8 strings can be included by surrounding the names in double quotes (") and appropriately escaping certain special characters.
- Letters are not case sensitive, whitespace is all treated the same, and underscore is the same as whitespace. *interest_rate*, *Interest_Rate*, *“interest_rate"*, *“interest rate“, “interest\nrate"* are all considered to be the same identifier.

Other model conventions

- Reserved identifiers: AND, OR, NOT, IF, THEN, ELSE, std, and the names of all builtin functions.
- Enter inline comments with curly braces.

```
a*b { take product of a and b } + c { then add c }
```

- Documentation (plain text or HTML) may also be store as part of building block using <doc> element.
- Data types – floating point numbers only.

isee systems OASIS IBM /forio System Dynamics Society VENTANA systems inc Vensim

Many builtin functions are included.

Example Specification

INT:	next integer less than or equal to the given number
Parameters:	1: the number to find next lowest integer of
Range:	(-∞, ∞); note negative fractional numbers increase in magnitude
Example:	INT(x)

List of Functions

- ABS
- ARCCOS
- ARCSIN
- ARCTAN
- COS
- EXP
- INF
- INT
- LN
- LOG10
- MAX
- MIN
- PI
- SIN
- SQRT
- TAN
- DELAY
- DELAY1
- DELAY3
- DELAYN
- FORCST
- SMTH1
- SMTH3
- SMTHN
- TREND
- PULSE
- RAMP
- STEP
- EXPRND
- LOGNORMAL
- NORMAL
- POISSON
- RANDOM
- DT
- STARTTIME
- STOPTIME
- TIME
- IF_THEN_ELSE
- INIT

Function calls and operators work as you'd expect, with standard infix expressions.

```
<eqn>IF TIME &gt; 2010 THEN Book_Revenue + Toy_Revenue ELSE
Book_Revenue</eqn>
```

Function Calls

- ABS(-3.1)
- TIME

Control Structure

IF *condition* THEN *expression* ELSE *expression*

IF_THEN_ELSE (*then-expression*, *else-expression*)

Array subscripting

- Sales[2]

Operator precedence

- [ ]
- ( )
- ^
- + – NOT
- * / MOD
- + –
- < <= > >=
- = <>
- AND
- OR

By default, the integration method is Euler's, but other methods are supported.

XMILE Name	Integration Method
Euler	Euler's method (default)
RK4	Runge-Kutta 4
RK2	Runge-Kutta 2
RK45	Runge-Kutta mixed 4th- 5th-order
Gear	Gear algorithm

- The last three integration methods are optional. In these cases, a supported fallback method should be also provided, for example, "Gear, RK4". This means that Gear should be used if the product supports it. Otherwise, use RK4.
- Some vendors do not offer RK2, as it is less useful than it once was when computing power was expensive. XMILE defines RK4 to always be the fallback for RK2, i.e., RK2 implies "RK2, RK4."

<model_units> specifies the units used throughout the model.

```
<model_units>
    <units>
        <eqn>dollars</eqn>
    </units>

    <units name="dollars per person per year">
        <eqn>dollars/(person-yr)</eqn>
        <alias>dpy</alias>
    </units>

    <units name="dollars per year">
        <eqn>dollars/yr</eqn>
</units>
</model_units>
```

```
<aux name="Salary">
   <eqn>80000</eqn>
   <units>dpy</units>
</aux>
```

<dimensions> specifies the array dimensions used throughout the model.

```
<dimensions>
  <dim name="Product"
                size="3"/>

  <dim name="Location">
    <elem name="Boston"/>
    <elem name="Chicago"/>
    <elem name="LA"/>
  </dim>
</dimensions>
```

```
<aux name="Sales">
  <dimensions>
    <dim name="Location"/>
    <dim name="Product"/>
  </dimensions>

  <eqn>1000</eqn>
</aux>
```

```
<array name="Sales">
  <dimensions>
    <dim name="Location"/>
    <dim name="Product"/>
  </dimensions>

  <aux subscript="Boston,1">
    <eqn>2000</eqn>
  </aux>
  <aux subscript="Boston,2">
    <eqn>2000</eqn>
  </aux>
  …
</array>
```

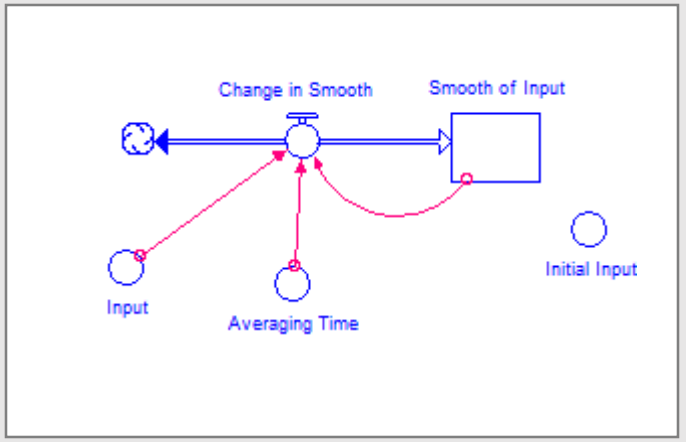
The model language can be extended with macros.

```
<macro name="LOG">
  <parm>x</parm>
  <parm>base</parm>

  <eqn>LN(x)/LN(base)</eqn>

  <format>LOG(<value>, <base>)</format>
  <doc>Finds the base-<base> logarithm of <value>.</doc>
</macro>
```

isee systems OASIS IBM /forio System Dynamics Society VENTANA systems inc Vensim

Macros may contain their own variables.

```
<macro name="SMOOTH1">
  <parm>input</parm>
  <parm>averaging_time</parm>
  <parm default="input">initial</parm>

  <eqn>Smooth_of_Input</eqn>

  <variables>
    <stock name="Smooth_of_Input">
      <eqn>initial</eqn>
      <inflow>change_in_smooth</inflow>
    </stock>
    <flow name="change_in_smooth">
      <eqn>(input - Smooth_of_Input)/averaging_time</eqn>
    </flow>
  </variables>
</macro>
```



Other features of XMILE include the ability to…

- Split simulations over multiple files, with a master file including a list of other files.
  - Modelers may wish to split files for modularity
  - Vendors may wish to publish vendor-specific macro libraries.
- Create new building blocks with macros, e.g. a non-negative stock.
- Include data from CSV or other sources.
- Specify events that are triggered when variables hit particular threshold.

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