It's about Time: The Why and How of Using XML for Developing an Interchange Standard for System Dynamics Models

Vedat G. Diker[1]
College of Information Studies
University of Maryland
4105 Hornbake Bldg.
College Park MD 20742
v d i k e r@wam.umd.edu

Robert B. Allen
College of Information Science and Tech.
Drexel University
3141 Chestnut Street
Philadelphia, PA 19104-2875
r b a@drexel.edu

Abstract

This paper discusses the benefits of having an interchange standard for system dynamics models, why XML is a good candidate on which to build such as standard, and how the development process may take place through community-wide participation. The paper also presents XMILE, a prototype model interchange standard, as a proof of the concept.

Introduction

As system dynamics modelers, we use modeling software to build, improve, analyze, test and review system dynamics models on a regular basis. Although system dynamics modeling software are generally user-friendly and provide a considerably high level of functionality to help us with our tasks as modelers, they do involve slight annoyances and give us moderate levels of trouble every now and then. Besides these "intra-application" annoyances, which we experience due to our choice of using "this" or "that" software, there is a different, "inter-application" type of annoyance, which surfaces whenever we venture to use "this" and "that" modeling software together, or use "this" software on a

[1] Contact author.

model built on "that" software. Although converting some native file types into other types are considerably less hard than vice-versa, it is a well established fact that there is no general, quick, simple, and painless solution to the problem of model file conversion. Just like the quintessential system dynamics wisdom of "All models are wrong, but some are less wrong than others," we can say "all conversions are painful, but some are much more painful than others." In fact, some conversions require building a replication of an entire model on the destination platform. However, the person-level frustration and toil is the least of the problems caused by complicated conversion processes. As we will discuss in this paper, the community-level losses may be much higher.

The system dynamics community has discussed about developing a model interchange standard in order to remedy the problem of model conversion between software for quite some time. We are not going to try and summarize all the discussion carried out over these many years, since doing justice to this task would require an ethnographic study of system dynamics folklore rather than a regular literature review. It should suffice to refer the readers to the system dynamics discussion list archives at http://www.ventanasystems.co.uk/forum/ to get an idea about the recent discussions regarding the topic.

Although we propose an XML-based standard as a viable and natural solution for the system dynamics model interchange problem, we are not the first people to think about this. There have been many others who thought XML can be a viable solution for this problem[2]. What we believe we add to the discussion is laying out the reasons why

[2] One of the open discussions on using XML as a basis for developing a model interchange standard took place on the System Dynamics Listserv over the months of February and March in 2003. The discussion started with a query from Carolus Grütters, who coined the term

developing such an interchange standard would be beneficial to our field and to the system dynamics community, and the way we can approach the problem of such a community-wide development endeavor. We also present a "proof-of-concept" prototype for the interchange standard, and the results of our first pass implementation of tools that would translate to and from that prototype.

Why Do We Need a Model Interchange Standard?

There are several highly critical reasons for developing a model interchange standard for system dynamics models. One of these has to do with encouraging development and use of new niche modeling and analysis software. The story of Digest (Mojtahedzadeh, 2001), a model analysis tool developed by Mohammad Mojtahedzadeh, is a good example of how the lack of seamless conversion between model file formats based on a common interchange standard can impede wide-scale use of niche software in our field. Digest can help modelers and analysts in identifying dominant loop structures during a

---

"Simulation Model Interchange Language Entity (SMILE)." Will Glass-Husain, Jim Hines, Raymond Joseph, Leonard Malczynski, Michael McDevitt, Magne Myrtveit, Anastássios Perdicoúlis, George Richardson, George Simpson and Jim Thompson contributed to the discussion. We build on and add to their perspectives. Readers are referred to the System Dynamics Listserv Archives at http://www.ventanasystems.co.uk/forum/ for details of that discussion. Another discussion took place on July 22, 2003 at the 21st International Conference of the System Dynamics Society in New York City. The session titled "A Big SMILE Discussion" was facilitated by Jim Hines, and was included in the formal conference program. More recently, Anastassios Perdicoulis used an XML-based representation for files native to a software application named Interact, which he is building with the help of other contributors. His announcement on the System Dynamics Listserv received replies from John Gunkler, Magne Myrtveit, and Bill Harris. Vedat Diker thanks Luis Luna-Reyes for valuable discussions and comments on the subject.

given phase of a simulation, thus providing insights about how the model structure drives model behavior. Given the promise Digest holds, and the relatively long period it has been around, one would expect it to be used widely by now. However, Digest is not being used. It is not being used at all, virtually[3] , let alone being used widely. There may be many reasons for this non-use, but we believe a very critical reason is the challenge of converting model files into a format that can be handled by Digest. As an example, a model built in Vensim has to be rebuilt in Stella/iThink, and then copied and saved as text before it can be read by Digest. Who knows; maybe Digest will not prove to be a very useful tool after all, but we will never know before it is applied to a considerably large number of studies. A model interchange standard, and related conversion tools, which would make it easier and simpler to convert model files could help increase use of Digest and other niche software in our field.

Easier and simpler model conversions provided by a common model interchange standard and related conversion tools would also improve the rigor of peer-review in our field. Wide-spread practice of model reviews along with the reviews of written work has been recognized as a goal for improving the general quality of research in our field. An important obstacle along the way is the challenge of converting model files from the application used by the author to the application used by the reviewer.

A model interchange standard would also help bridging the distance between different segments of researchers within our field. Although we do not have any empirical

[3] A search for the keyword "digest" over the abstracts published in the International System Dynamics Conference proceedings for the last five years (2000-2004) returns six abstracts, four of which are about workshops and presentations that introduce Digest itself. This leaves two pieces of work making explicit use of Digest, out of 1300 or so works published in the proceedings over the last five years.

findings, it is our observation that specific segments of the system dynamics community tend to prefer different modeling software. A good example is the K-12 segment, who seem to prefer Stella/iThink. The barriers in front of model file conversions prevent individuals who use other applications from studying deeper the work generated by that segment.

Furthermore, if modelers can convert model files easily, they may choose to use two or more modeling or analysis applications in parallel. That would allow modelers to mix-and-match the best features of each application, thus using one for modeling, one for analyzing, and yet another one for presenting their models. Also, using application-specific analysis tools from two or more applications would arguably help building more rigorous models, since one problem missed by a tool may be caught by the other. These are only some of the most obvious benefits of having a common model interchange standard. We are sure that other colleagues in our field do and will think of many other such benefits.

Finally, we think XML is a good candidate as a basis for a model interchange standard for several reasons. It is a widely accepted metadata standard. Since it is text based, it is easier to transfer and edit. There are many tools to support XML use.

How Can We Develop an XML-based Model Interchange Standard?

We believe an endeavor as comprehensive as developing a model interchange standard requires a wide participation across the system dynamics community. The ideal approach would be forming a consortium under the endorsement of the System Dynamics Society, where a substantial number of modelers and software developers would share their ideas

in order to build a robust and widely agreed-upon standard. We envision that as the standard emerges, so will small projects for developing translators to and from XMILE. These can be individual or small group projects, as well as those that involve tens of individuals. Along with such tools, new uses will occur, which will render the standard more and more relevant. This will hopefully be followed by efforts by major software developers for including built-in translators in leading system dynamics software. Ultimately, the real need for such an interchange standard, and the success of the implementation will determine whether the standard will gain acceptance in the system dynamics community. It is our belief that the need is there, and the community is apt to agree on a robust standard.

XMILE - A "Proof-of-concept" Prototype for an XML-based Model Interchange Standard.

We developed a prototype XML-based model interchange scheme we call XMILE, named after Carolus Grütters' hypothetical SMILE. We also developed parsers (translators) that convert Vensim and Stella/iThink text representations into XMILE, and from XMILE into Vensim and Digest text representations.

The basic structure of XMILE is as follows:

```
<XML>
   <FILESPACE>
      <MODEL>
         <VARIABLEBLOCK>
            <Variable>
               <VariableName> </VariableName>
               <VariableType> </VariableType>
               <VariableValue> </VariableValue>
               <VariableUnits> </VariableUnits>
               <VariableComment> </VariableComment>
            </Variable>
```

```
            <Variable>
            </Variable>
        </VARIABLEBLOCK>
        <CONTROLBLOCK>
        </CONTROLBLOCK>
        <DISPLAYBLOCK>
        </DISPLAYBLOCK>
        <UNITEQUIV>
        </UNITEQUIV>
    </MODEL>
    <DATA>
    </DATA>
  </FILESPACE>
</XML>
```

We applied XMILE to a simple population model in order to tests the concept and the parsers. Two example variables (one flow and one stock) from the model are represented in XMILE as follows:

```
<Variable>
   <VariableName>Births</VariableName>
   <VariableType>flow</VariableType>
   <VariableValue>Birth Coefficient*Population  </VariableValue>
   <VariableUnits>People/Year</VariableUnits>
   <VariableComment></VariableComment>
</Variable>

<Variable>
   <VariableName>Population</VariableName>
   <VariableType>stock</VariableType>
   <VariableValue>INTEG (  Births-Deaths,   Initial Population)  </VariableValue>
   <VariableStock>
      <StockFlow>Births-Deaths</StockFlow>
      <StockInitialValue>Initial Population</StockInitialValue>
   </VariableStock>
   <VariableUnits>People</VariableUnits>
   <VariableComment>This is the total population at a given time.</VariableComment>
</Variable>
```

Conclusion

In this paper, we discussed why we urgently need a model interchange standard, why XML is a good candidate for the job, and how we envision the process of developing the

standard with a community-wide participation. We also summarized our efforts for developing a prototype model interchange standard we named XMILE. We hope this paper starts a productive discussion within the system dynamics community toward the development of the standard we have been waiting for quite a while now.

Reference

Mojtahedzadeh, M., Andersen D., Richardson, G. 2004, Using Digest to Implement the Pathway Participation Method for Detecting Influential System Structure, *System Dynamics Review*, 20 (1), pp. 1-87.