FULLY EMBEDDED SIMULATION MODELS IN MANUFACTURING PROCESSES

Shabnam Tauböck[1], Michael Steiner[1], Markus Klug[1]

[1]Profactor Research & Solutions
Research Centers Seibersdorf
2444 Seibersdorf
AUSTRIA

Shabnam.Tauboeck@profactor.at

Abstract

This paper presents the concept of using simulation as a fully embedded tool.

Simulation is an often used tool for planning and optimizing. The possibility to pursue different scenarios without actually risking a lot of money and time is one of the biggest advantages of modeling and simulation. But simulation still holds the impression of being usable to only those people that possess a vast knowledge of its basics and dynamics. The results given by a simulation can be easily presented in a way that everyone familiar with the topic can interpret them. But to create a new simulation model even using a specialized simulator usually holds no appeal for those who are manly interested in the results. One step towards enhancing user friendliness was developing database driven models that are automatically created according to the information stored in the external database.

But still it is time consuming to prepare this data; Simulation as a fully embedded tool is designed to do all this automatically, using the fact that the data needed for a simulation run could often be provided by other software tools that are already in use in most companies: An Enterprise Resource Planning System (ERP) contains a huge amount of information already structured and always up to date. An interface with such a software tool can provide the information needed to create a model of the system, thus reducing the complexity to a one time expense to develop the interface between such a system and the model database. Of course one must not underestimate the complexity creating such an interface but the result is a simulation tool that can be used at any time to analyze the current situation; the potential of benefits that can be achieved here easily outbalances the effort.

Keywords: Enterprise Dynamics, Interactive, Embedded Simulation.

Presenting Author's biography

Shabnam Michèle Tauböck works for Profactor Research and Solutions since March 2007. From Dezember 1999 until March 2007 she worked for the business area Process Optimization, Division Environment and Life Science at ARC Seibersdorf research. From June 1998 until April 2000 she worked for PSE SIEMENS in Vienna.

1 General

Simulation is an often used tool for planning and optimizing. The possibility to pursue different scenarios without actually risking a lot of money and time is one of the biggest advantages of modeling and simulation. This paper gives a summary over the different approaches to minimize the effort necessary to work with a simulation model, as well as an outlook of how to combine different approaches to make simulation an easy to handle tool for everyday use.

This would make simulation a support tool for decision making as it offers a quick comparison of different approaches to solve a current problem;

It could be used as a control tool with the additional feature of analyzing the nearest future based on the results of simulation runs.

Only two aspects make it difficult to use simulation:

Knowledge: Simulation still holds the impression of being usable to only those people that possess a vast knowledge of its basics and dynamics. The results given by a simulation can be easily presented in a way that everyone familiar with the topic can interpret these. But to create a new simulation model even using a specialized simulator usually holds no appeal for those who are manly interested in the results.

Time: Creating a new simulation model takes time, even for the most experienced simulation expert.

2 Enterprise Dynamics

Enterprise Dynamics is a tool for simulating discrete event processes. The chosen simulation tool - Enterprise Dynamics - offers a huge flexibility due to its internal programming language *4D Script*, which enables adding further functionality to the provided elements as well as creating completely new ones.

At present there exist a lot of software tools for modelling and simulation of discrete dynamic systems. Almost all tools are based on a modern object-oriented DEVS world view regarding active objects (entities) passing passive objects (stations) along given path. A time event mechanism updates movement of entities, being capable also of collisions and other state-dependent phenomena. Some of these tools can look back at a long tradition, now offering also any kind of object-oriented features, etc. Some of the tools are results of new research projects, starting with object-oriented approach from the first release.

All these tools are more or less similar at the basic modelling level where general entities are passing general stations. But big differences are met in application libraries: some tools have specialised for a certain application, like network modelling, offering preconfigured stations like network routers. Other tools offer different modelling levels, from a library with basic elements to high specialised modelling libraries.

In these investigations the software tool Enterprise Dynamics is used, an object-oriented dynamic analysis and control system. The system consists of a Enterprise Dynamics (e.D.)-engine® and many building blocks grouped into e.D.-Suites®. An e.D.-Suite is configured for a specific field of expertise, to assist the modelling of a specific problem, branch or area.

Animations can range from 2D flowcharts to true 3D Virtual reality models that empower imagination and creativity. Building blocks can be easily created, customized and added.

3 Interface to External Data Source

To improve the speed of simulation runs it is often recommendable to keep as much data as possible out of the simulation environment and in external databases. This also enhances the user friendliness as these databases are much more convenient to handle.

Several projects done during the last years have shown the tendency to hide the simulation environment itself from the user behind an easy to manage General User Interface (GUI) to offer a more comfortable way to parameterize the simulation model.

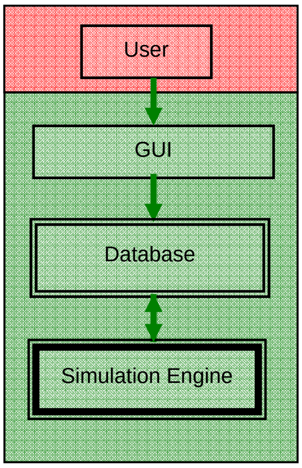

Fig. 1: Data Exchange between user and Simulation Environment

Enterprise Dynamics offers the possibility to use ODBC connections to an external data source or ADO to communicate with a database as ACCESS or ORACLE.

The more complex a simulation model becomes and the more complex therefore the parameterization becomes the more complex the user interface becomes and entering information is a task that is most of the time more time demanding than the simulation run itself.

ISBN 978-3-901608-32-2

4 Database driven Model Generation

To offer a possibility of quickly creating and modifying simulation models the approach of database driven model generation was developed [1]. Basically this is done by splitting the simulation model specific data and the basic model to enhance the flexibility of the simulation model:

To minimize the effort of redesigning the simulation model the approach of self constructing simulation models proves itself to be the most convenient. It enables even users with no knowledge of the simulation tool to change the simulation model by altering the according parameters. Such self constructing models need longer to develop but are a lot more convenient in usage.

The basic idea is to separate the basic model from the scenario specific data. Keeping the data outside of the database e. g. in a database offers several advantages:

- the data is much easier to manage
- the data is much easier to modify
- the simulation model does not store any more data during the simulation run than strictly necessary; this is especially noticeable with result data.

The input data needs to be structured to represent a model of the simulated system that contains all possible combinations of parameters.

The interaction with the database can be easily done via a GUI, hiding the simulation environment as well. Additionally this GUI can be used to start a simulation run and to analyze the results.

This approach implies the split between a simulation environment and a data model: The simulation environment contains the basic model elements as well as the functions to create the simulation model according to the data in the external database. The data model contains all additional information needed.

5 Real Data

Simulation can be used for experimenting with future developments as well as for analyzing the current status of an existing system, i.e. a production plant. This can provide a deeper insight to effects decisions and changes may have on the whole system, something that may not be easily possible to see without the help of a simulation, especially for more complex systems with many interactions and cross influences.

To analyze the current status would mean to use the current state of the system as an initial setting for the simulation to observe its behavior in the nearest future.

Using the current status of a production plant to simulate the next 24 hours can show if any bottlenecks will occur or if the production can be done as planned. Of course it is a precondition that really all data about the current status is available and entered as initial values to the simulation system. That includes all stock levels, all recourses available, planned downtimes, available personnel and the orders that need to be filled.

Additionally all basic system parameters must be given to the system as machine cycle times, shift times, transport times etc.

This implies the need for entering the current status to the simulation tool to create a model that mirrors the real system.

A problem very quickly arising here is the huge amount of data needed to create and initialize the model.

To ensure the transfer of all data a direct interface to an existing system, for example an Enterprise Resource Planning System (ERP systems) is the most convenient way as it offers several advantages:

- To any given time real system data can be imported to the simulation tool.
- Data already structured
- The current status can be analyzed
- Problems arising during the productions identified beforehand
- Solutions can be tested to compare the effect

This ensures the interest of developing interfaces to already existing systems for easy data transfer.

Problems: interfaces to develop – depending on tool used.

Completeness of data

6 Crossing System Borders

Usually a simulation model is a closed system where a simulation run is only influenced by the system parameters and the input data.

Complex decisions within the system may depend not only on the current status of the simulated system but may be influenced by additional data that can only be received from outside the simulated system. This forces an interaction with the outside, to receive the data critical for the decision.

One simulation model developed in the last year was designed to ask for the input of the user for certain decisions. The system itself was not able to choose a strategy but stopped the simulation run at this point and required an input from the user; the simulation run continued as soon as this input was given, using the according strategy.

The reason for such an interaction can be that the rules used to come to a decision are based on information that is not available in the simulated system.

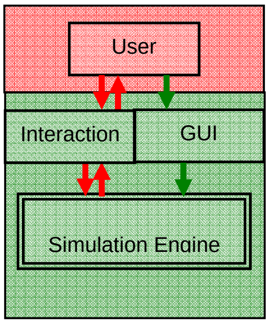

Fig. 2: Data Exchange and Interaction between User and Simulation Environment

This interaction may be directly with the user – the simulation run stops and waits for a decision made by the user before it continues the simulation run accordingly. This requires a GUI for interacting with the simulation environment.

The interaction may also be with another software program, i.e. the ERP system to get additional data that may influence the decision – i.e. availability of staff or additional resources.

7 GUI for Simulation Environment

A GUI to communicate with the Simulation Environment is needed to enter certain simulation run specific data as well as receive information during and after the run.

- Status of the runs
- Indicate problems
- Compare different solutions

It can also be used to interact with the simulation environment during the simulation run if needed and to process and analyze the results of the simulation runs.

8 The Future: Fully Embedded Simulation

All this features have been developed for different simulation projects in the past. The task to combine them is only a matter of defining the interfaces between the single modules:

8.1 Database

8.1.1 Data Model

The data model contains the information needed to create all possible scenarios. This data model is implemented in an external data source, as a database or Microsoft Excel. This data source is then used to specify the scenario to be simulated.

8.1.2 Interface

It also needs to interact with the external data source for preprocessing real data to create a scenario based on the current state of the system.

8.2 Simulation Environment:

8.2.1 Model template

The model template contains several different functionalities:

8.2.1.1 Interface to Database

An interface to the database enables the simulation environment to import the model specific data. This includes the

- structural parameters of the system
- the parameters of the model elements
- global parameters to define the state of the system

8.2.1.2 Functions for Model Buildup

Before the simulation model is created several actions have to be taken. These are executed by functions embedded in the simulation model template.

Such actions are the import of the data from the data model; the creation of the simulation model, the start of a simulation run, the processing of the result data. All these actions can be automatically triggered if a simulation run is started.

8.2.1.3 Interactivity with the User

The interaction with the user can be used to report any problems detected during the simulation run:

- Machine down
- Lack of available personnel
- Production behind plan
- Bottlenecks
- Shortage of material

The user is able to make a decision how the simulation model should proceed: the run can be completed with the current setting, or several actions can be triggered to be executed in the simulation.

While the simulation run is initialized with the current status of the system, it could be possible to change some of the parameters at this point to watch the effect of different measures taken.

It also would be of interest to set certain measures at a point in time before the problems arise, so it would make sense to offer a possibility to define runs with parameters that are based on real data but modified to catch the problems detected before they arise. This should be done using the GUI.

8.2.2 Model library

The simulation Library contains all elements that are defined in the data model. Each has its own behavior that can be influenced by the parameters given in the data model. The functions creating the simulation model will use these elements and set their individual parameters after the model buildup.

8.3 Real Data (ERP)

External tool that contains the information of the current state of the system

8.4 GUI

8.4.1 Run Control

The GUI contains functionality to start and control simulation runs.

8.4.2 Scenario definition

It also would be of interest to set certain measures at a point in time before the problems arise, so it would make sense to offer a possibility to define runs with parameters that are based on real data but modified to catch the problems detected before they arise. This should be done using the GUI

8.4.3 Database parameterization

The database contains the whole model structure. Basic parameters that can not be retrieved by the external data source need to be entered directly to the database. The GUI should offer the possibility to edit and enter this basic data.

8.4.4 Interface to external tool

The database should be able to retrieve the current status of the system from the external tool. The exact definition of this interface is depending on the external tool used.

8.4.5 Editing data in the database

Additionally to the data import from an external data source the creation of further scenarios is essential for using simulation to assess the effect of different measures taken. Therefore the GUI needs to offer an easy way to edit certain parameters and values to minimize the time and effort of creating new scenarios.

8.5 Results

All results are stored in the database; for analysis and comparison of simulation runs special programs might be used if the functionality provided by the database is not sufficient.

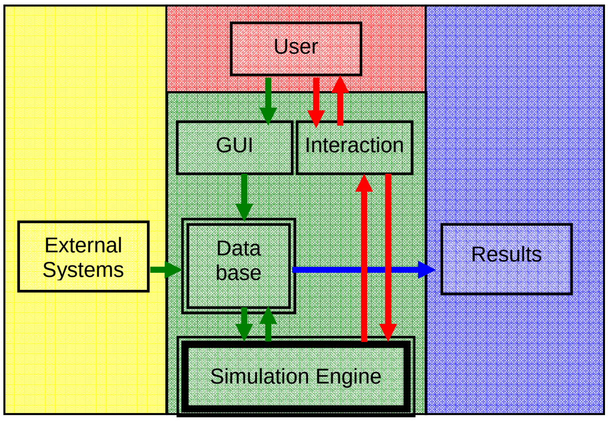

Fig. 3: Interactions of the Simulation Environment

The combination of all the modules listed here leads to a simulation tool that acts as a fully embedded software component interacting with already in use software for data exchange. It is able to interact with other systems to get data and information relevant for simulating the current system status and creating a forecast for the nearest future. It is able to detect problems arising in the next future and offers the possibility to create scenarios via the GUI to find the best solution by testing different approaches.

It can interact directly with either the user or another external system to find a solution not only based on the state of the simulated system but also on data from the external world.

9 Results

Simulation as fully embedded tool is certainly a powerful instrument that can be used as a decision support tool as well as easily extended to help optimize the workflow for manufacturing and logistic processes. The possibility to interact with the system offers a way to add human decisions whenever needed or wanted.

Keeping the effort of data input to a minimum surely makes simulation a much easier and more flexible tool to use even without expert knowledge.

10 References

[1] Bernhard P. Zeigler, Herbert Praehofer, Tag Gon Kim (2000). *Theory of Modeling and Simulation - Integrating Discrete Event and Continious Complex Dynamic Systems*, Academic Press.

