Proceedings of the 1996 Winter Simulation Conference
ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain

MODELING WITH EXTEND™

David Krahl

Imagine That, Inc.
6830 Via Del Oro, Suite 230
San Jose, CA 95119, USA.

ABSTRACT

This document presents an overview of the Extend modeling environment, emphasizing Extend's ability to model a wide range of systems. Extend is a general purpose graphically oriented discrete event and continuous simulation application with an integrated authoring environment and development system.

1 INTRODUCTION

For many years there has been a perceived dichotomy in simulation software between simulation languages and simulators. The languages were viewed as more powerful and general purpose, while simulators focused on ease of use and were generally limited to a specific industry. Extend bridges these two types of programs in one easy-to-use, yet flexible software program. It exists as:

- A stand-alone simulation tool which can be used to create complex discrete event and continuous models without programming.
- A simulation authoring package where model interfaces can be easily created, without programming, to enhance productivity and ease of use.
- A development environment for building customized models of unique types of systems. The programming environment allows the modeler to create a simulator for a specific industry.

2 EXTEND'S MODELING ENVIRONMENT

Before looking into how Extend can be used to build models, it is helpful to understand the Extend modeling environment.

Extend models are constructed with library-based iconic blocks. Each block describes a step in a process or a calculation. Blocks reside in libraries. Each library represents a grouping of blocks with similar characteristics such as Discrete Event, Plotters, Electronics, or Business Process Reengineering. Blocks are placed on the model worksheet by dragging them from the library window onto the worksheet. The flow is then established between the blocks.

There are two types of logical flows between the Extend blocks. The first type of flow is "items," which represent the objects that move through the system. Items can have attributes and priorities. Examples of items include parts, patients, or a packet of information. The second type of logical flow is "values," which will change over time during the simulation run. Values represent a single number. Examples of values include the number of items in queue, the result of a random sample, and the level of fluid in a tank.

Each block has connectors that are the interface points of the block. Figure 1 shows the connector symbols for the value and item connectors.

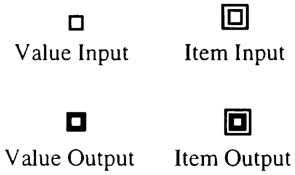


Figure 1: Value and Item Connectors

Connections are lines used to specify the logical flow from one connector to another. Double lines represent item connections and single lines represent value connections.

A model of a single server, single queue system would have the following form:

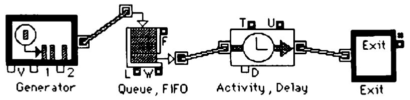


Figure 2: A Single Server Single Queue Model

The block on the far left represents a Generator which periodically creates items. Following this is a Queue block that holds items until requested by the next block. The Activity Delay represents a limited capacity of one processing unit and delays an item for a fixed amount of time. The last block in the model is an Exit block that removes items from the system.

An enhancement to this model would be to specify that the delay in the Activity Delay is determined by a specific random distribution. This can be done by connecting the output of an Input Random Number block to the delay connector (labeled "D") on the Activity Delay block as in Figure 3:

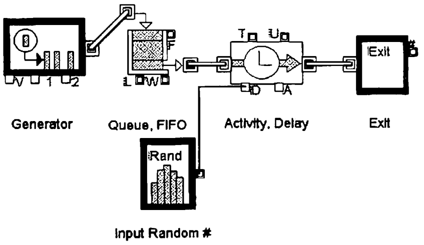


Figure 3: A Model With Random Process Times

Another feature that can be added to the model is a Discrete Event Plotter which graphically displays, in this example (Figure 4), the contents of the Queue. The Discrete Event Plotter value input connector will be connected to the Queue's length (labeled "L") value output connector as follows:

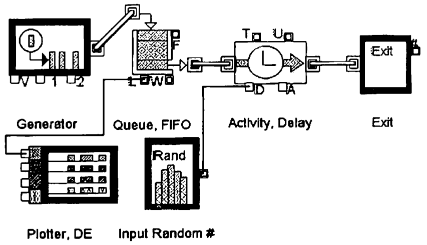
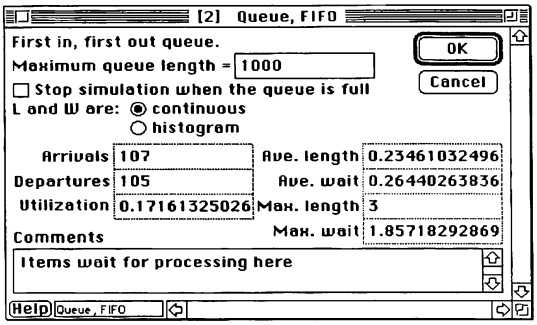


Figure 4: Discrete Event Plotter Added to Model

Simulation parameters such as the number of runs and simulation end time can be specified in the Simulation Setup dialog item under the Run menu. The simulation can then be run by selecting the Run Simulation menu item from the Run menu.

During the run, the current simulation status is displayed in a bar near the bottom of the monitor screen. This displays the estimated time before the run will be completed, the current simulation run time, the number of simulation steps completed so far, and the current simulation run number.

Once the simulation run has completed, the results of the simulation are reported within the blocks. Double clicking on each block reveals the information collected from the simulation run. For example, double clicking on the Queue FIFO block opens a dialog which shows the following information about the state of the Queue FIFO block:



First in, first out queue.

Maximum queue length = 1000

□ Stop simulation when the queue is full

L and W are: ● continuous ○ histogram

Arrivals	107	Ave. length	0.23461032496
Departures	105	Ave. wait	0.26440263836
Utilization	0.17161325026	Max. length	3
		Max. wait	1.85718292869

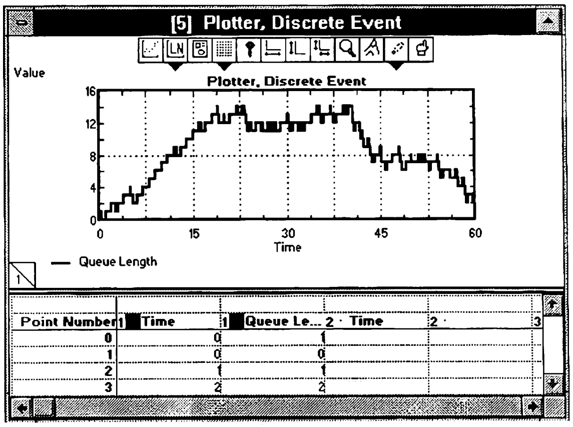
Comments

Items wait for processing here

Queue, FIFO

Figure 5: Dialog of Queue FIFO

The Plotter block shows the number of items stored in the Queue FIFO over time in both graphical and tabular format:



Point Number	1 Time	1 Queue Le...	2 · Time	2 ·	3
0	0	1			
1	0	0			
2	1	1			
3	2	2			

Figure 6: Plot of Queue Length

Simulation results may be stored in a table, plotted, cloned to a different area of the worksheet, exported to another program such as a spreadsheet or database, displayed in an animation, or even used to control some aspect of the outside world through external device drivers.

3 STANDARD EXTEND LIBRARIES

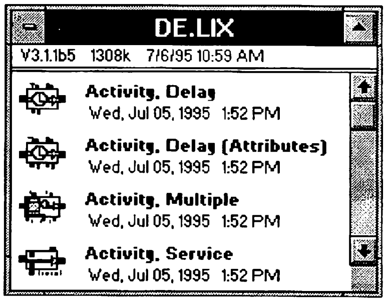
The standard Extend libraries include constructs for discrete event modeling, results plotting, generic calculations, electronics design, interprocess communication, and utilities. For discrete event modeling, the most commonly used standard libraries are the Discrete Event, Generic, and Plotter. Additional, optional, discrete event libraries include the Business Process Reengineering and Manufacturing libraries.

Extend supports the following general modeling functionality for discrete event modeling:

- Attributes - Unique variables which are local to the items moving through the simulated system.
- Priorities - A unique value, local to a given item, which can be used to rank items in a queue or interrupt items in process.
- Values - The number of items represented by a single item. Setting a value will create clones of an item when that item arrives to a queue, resource, or exit block.

3.1 Discrete Event Library

The Discrete Event library contains blocks which are specific to modeling discrete event systems. In a discrete event model, the clock will update at intervals dictated by the individual events in the system. The discrete event blocks pass items to one another through their item connectors. If a discrete event block is unable to receive an item it rejects it and the item waits until it receives a downstream request.



DE.LIX

V3.1.1b5 1308k 7/6/95 10:59 AM

Activity. Delay
Wed, Jul 05, 1995 1:52 PM

Activity. Delay (Attributes)
Wed, Jul 05, 1995 1:52 PM

Activity. Multiple
Wed, Jul 05, 1995 1:52 PM

Activity. Service
Wed, Jul 05, 1995 1:52 PM

Figure 7: Discrete Event Library Window

The most commonly used block types in the Discrete Event library are as follows:

- Activities - Time delays
- Batching - Combining of items
- Resources - Limits capacities
- Decisions - Chooses alternate paths

3.2 The Generic Library

The Generic library is used for both continuous and discrete event modeling. In the continuous mode, calculations are performed at each evenly spaced clock step. In the discrete event mode, calculations are made in response to a request (message) from a discrete event block.

When used with the Discrete Event library, the Generic library is typically used to provide values for inputs or operate on the value outputs of the Discrete Event blocks. Typical examples of using the Generic library in this mode include using a Decision block to compare the length of two queues or using an Input Random Number block to generate a random time delay for an Activity.

There are a number of classes of Generic blocks. These include: mathematical calculation, integration, file operations, logical calculations, integration, statistical calculations, error reporting, simulation events, accumulation, and threshold detection.

3.3 Other Libraries

In addition to the above libraries, Extend also includes libraries for statistics, animation, plotters, utilities, electronics, filters, digital circuits, controls, dynamic data exchange (DDE), and DLLs or XCMDs. Libraries are available from third party developers for control systems, paper manufacturing, neural networks, biology, and signal processing.

4 CUSTOMIZING EXTEND

The above discussion illustrates the highly graphical and interactive nature of Extend. However, Extend can also take the shape of the model application. Interfaces, components, and graphics can be used which tailor the model to a specific application area.

The most visible aspect of a custom model is the user interface. By modifying an existing interface or creating a new one, the simulation modeler is able to create a model which can be exercised by someone more familiar with the system than with the simulation tool. Models can be built that fit naturally into the conceptual framework of the person using the model. In Extend, this can be done with a range of tools, including:

- Hierarchical modeling - Models can be subdivided into logical components.
- Dialog cloning and the Notebook - Consolidate critical parameters and results to a central location. This can be used to create a custom user interface.
- Interactive controls - Add sliders and switches which provide interactive control of the model

- Pictures and animation - The model is the interface.
- User messages - Users can be prompted for information at any point in the simulation.
- Block development environment - Create modeling components for a specific environment.

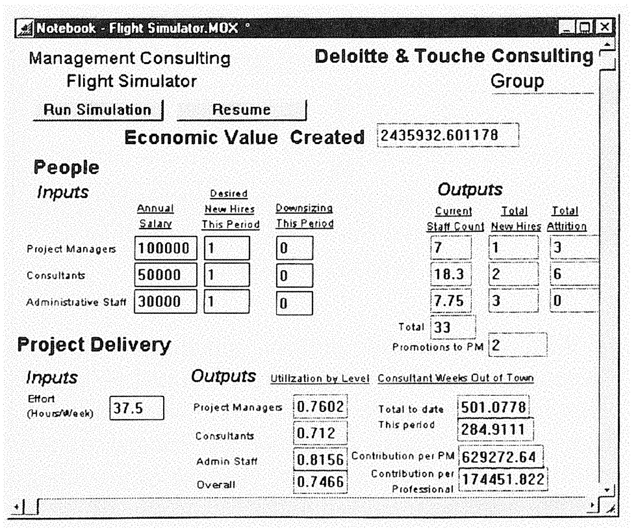
5 SAMPLE EXTEND MODELS

The following sections will present a number of applications which utilize Extend's features to create models in different areas. These models come from users with a variety of backgrounds and simulation experience levels. What is presented here is only a small subset of the areas where Extend is currently being applied.

5.1 Management Flight Simulator

The Deloitte & Touche Consulting Group Management Flight Simulator (a model for experimenting with different management strategies) is used internally for training sessions. This model, developed by Rick Blaiklock of their Ottawa office, focuses on the relationship between the economic factors of: staffing levels, sales effort, competition, employee training, and market research.

In the training session, participants run a consulting operation for a total of ten years of simulated time. At the end of each six month period, operational decisions are entered into the model and the economic results are plotted over time. Figure 8 illustrates the input panel for the decisions regarding the staffing levels of the model. Similar panels are used for the other business factors.



Management Consulting Flight Simulator

Deloitte & Touche Consulting Group

Economic Value Created 2435932.601178

People

Inputs

	Annual Salary	Desired New Hires This Period	Downsizing This Period
Project Managers	100000	1	0
Consultants	50000	1	0
Administrative Staff	30000	1	0

Outputs

Current Staff Count	Total New Hires	Total Attrition
7	1	3
18.3	2	6
7.75	3	0

Total 33

Promotions to PM 2

Project Delivery

Inputs

Effort (Hours/Week) 37.5

Outputs

Utilization by Level

Project Managers	0.7602
Consultants	0.712
Admin Staff	0.8156
Overall	0.7466

Consultant Weeks Out of Town

Total to date	501.0778
This period	284.9111
Contribution per PM	629272.64
Contribution per Professional	174451.822

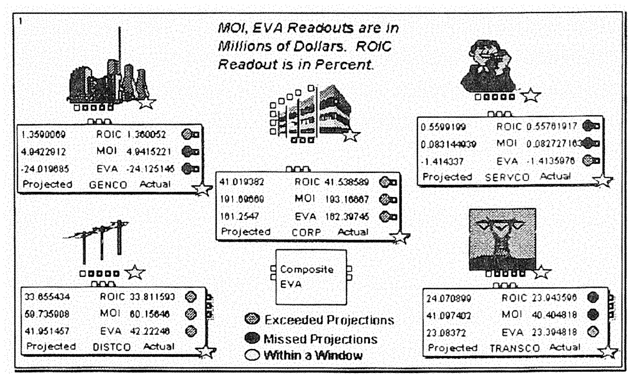
Figure 8: Management Flight Simulator

Here, dialog cloning is used extensively to create a user interface for model users who are not necessarily proficient in simulation. Because of this, the class participants are able to use a sophisticated model with only a brief introduction to the inner-workings of the model.

5.2 Economic Value Added Model

The Computer Aided Process Improvement (CAPI) Balanced Scorecard Economic Value Added Model (EVA) is an analysis tool designed to provide the backbone and baseline for reengineering and utility operations improvements. This allows executives to see the impact of decisions they have made on the balance sheet. The model is designed around the organizational framework of a large utility company operating via Strategic Business Units (SBUs). Of the four SBUs, three are revenue-generating and one is dedicated to cross-functional overhead processes. The revenue-generating SBUs are: Generation, Transmission, and Distribution, while the overhead processes are grouped into Services. The CAPI EVA model is capable of calculating the following:

- Economic Value Added (EVA) for each SBU and for the Corporation
- Return on Invested Capital (ROIC) for each SBU and for the Corporation
- Measured Operating Income (MOI) for each SBU and for the Corporation
- Average Capital for each SBU and for the Corporation
- Total Revenue for each SBU -- monthly and accumulated YTD
- KWH (kiloWatt hours) System Sales -- monthly and accumulated YTD
- Off System Sales -- monthly and accumulated YTD
- By Product Sales -- monthly and accumulated YTD
- Physical Plant and Equipment (PP&E) Maintenance -- monthly and YTD
- Total Cost to Generate Power -- monthly and accumulated YTD
- Total Labor Costs -- monthly and accumulated YTD
- Cost of Sales -- monthly and accumulated YTD



MOI, EVA Readouts are in Millions of Dollars. ROIC Readout is in Percent.

1.3590069 ROIC 1.360052
4.9422912 MOI 4.9415221
-24.019685 EVA -24.125146
Projected GENCO Actual

41.019382 ROIC 41.538589
191.69669 MOI 193.16667
181.2547 EVA 182.39745
Projected CORP Actual

0.5599199 ROIC 0.55781917
0.083144939 MOI 0.08272716
-1.414337 EVA -1.4135976
Projected SERVCO Actual

33.655434 ROIC 33.811593
59.735908 MOI 60.15646
41.951467 EVA 42.22248
Projected DISTCO Actual

24.070899 ROIC 23.943596
41.097402 MOI 40.404818
23.08372 EVA 23.394818
Projected TRANSCO Actual

Composite EVA

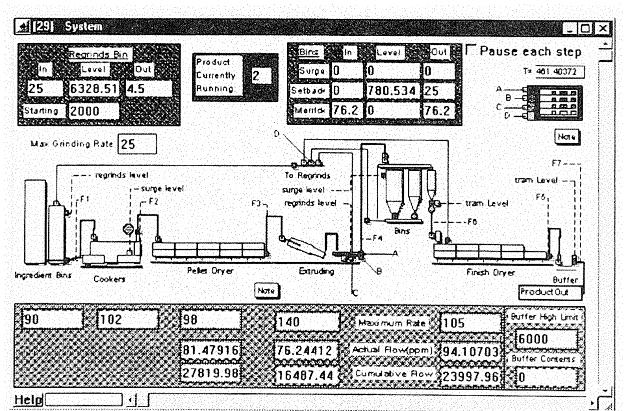
◍ Exceeded Projections
● Missed Projections
○ Within a Window

Figure 9: CAPI Economic Value Added Model

To build this model, standard Extend blocks were grouped together into hierarchical blocks (H-blocks). These were then placed in libraries. Models are built by selecting the hierarchical block from the library, placing it on the model worksheet, connecting other blocks, and filling in model parameters. By using this method a custom library was created without programming that incorporates the expertise of the library developer.

5.3 Bulk Processing

Another application of Extend is a model of a bulk processing system using blocks from Simulation Dynamics Inc. Bulk Flow Library. The blocks used to build the model in Figure 10 can also be used to build detailed models of any type of system that moves bulk quantities of materials.



[29] System

Regrinds Bin: In 25, Level 6328.51, Out 4.5
Starting 2000

Product Currently Running: 2

Bins	In	Level	Out
Surge	0	0	0
Setback	0	780.534	25
Metldk	76.2	0	76.2

□ Pause each step
T= 461.40372

Max Grinding Rate 25

90 102 98 140 Maximum Rate 105
81.47916 76.24412 Actual Flow(ppm) 94.10703
27819.98 16487.44 Cumulative Flow 23997.96
Buffer High Limit 6000
Buffer Contents 0

Figure 10: Bulk Processing Model

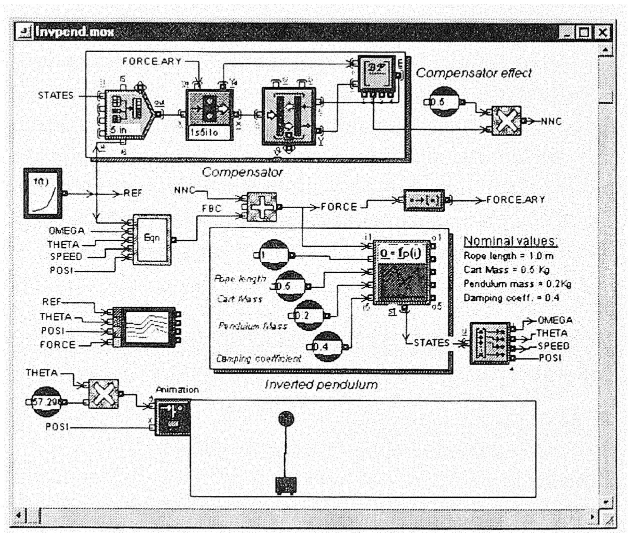
In this example, a library of blocks has been programmed to provide a simulator for a specific industry. Engineers are able to use the bulk processing architecture to build, run, and modify models without the burden of learning the intricacies of a general purpose simulation language. At the same time, Extend's flexibility remains should the need arise to model a system component which is beyond the scope of the existing Bulk Processing library.

The architecture is extremely applicable to production systems in industries such as pulp and paper, food processing, bottling lines, oil refining, high speed plastics molding, and any other application in which production is termed in rates instead of items.

5.4 Neural Networks

Another interesting application of Extend is the Neurolab library of blocks from Mikuni Berkeley. This library allows modelers to build neural networks by combining various network components.

Figure 11 shows a control network for an inverted pendulum. The control objective is to balance the pendulum with desired cart position. The two-layer feed-forward network is the compensator that improves the performance of the state feedback controller by adapting to the nonlinear dynamics and parameter changes of the inverted pendulum.



Invpend.mox

Nominal values:
Rope length = 1.0 m
Cart Mass = 0.5 Kg
Pendulum mass = 0.2Kg
Damping coeff. = 0.4

Figure 11: Neural Network Model Controling an Inverted Pendulum

With the Neurolab product, simulation has been opened up into neural network research and evaluation. Neurolab is applicable in any area where nonlinear mapping and adaptive capability are integral parts of problem solving. This is another example of providing a simulation software product for a specific area of interest through Extend's development environment.

6 SUMMARY

It is easy to focus on any one of Extend's features such as ease of use or graphical interface as its main attraction for simulation modelers. However, this view tends to eclipse the true power of Extend to provide modelers the ability to create unique models for their area of application. Extend's applicability and extensibility are unmatched in the simulation industry. As a result, Extend has become the simulation tool of choice for thousands of engineers and analysts ranging from simulation newcomers to experts.

REFERENCES

Computer Aided Process Improvement. 1996. *The Balanced Scorecard Economic Value Added Model.* Cincinnatti, OH

Imagine That, Inc. 1995. *Extend Software Manual.* San Jose, CA.

Deloitte & Touche. 1996. *Management Consulting Flight Simulator.* Ottawa, ON.

Krahl, David. 1995. Building End User Applications With Extend, In *Proceedings of the 1995 Winter Simulation Conference*, ed. C. Alexopoulos, K. Kang, W. R. Lilegdon, D. Goldsman, 413-419. IEEE Piscataway, NJ.

Mikuni Berkeley R&D Corp. 1995. *Neurolab User's Manual.* Richmond, CA

Simulation Dynamics. 1996. *Making Model.* Maryville, TN.

AUTHOR BIOGRAPHY

DAVID KRAHL is responsible for the technical support and block development for Extend at Imagine That, Inc. Mr. Krahl received a BS in 1986 in Industrial Engineering from the Rochester Institute of Technology and a MS in Project and Systems Management from Golden Gate University in 1996. He has performed consulting, technical support, and development for a wide range of simulation Products.