Proceedings of the 2002 Winter Simulation Conference
E. Yücesan, C.-H. Chen, J. L. Snowdon, and J.M. Charnes, eds.

NON-ITEM BASED DISCRETE-EVENT SIMULATION TOOLS

Richard A. Phelps
David J. Parsons
Andrew J. Siprelle

Simulation Dynamics, Inc.
416 High Street
Maryville, Tennessee 37804-5836, U.S.A.

ABSTRACT

Discrete event simulation has traditionally been defined by items (or entities). This modeling paradigm has served the simulation industry well, but falls far short for many industries in which the parts / pieces mindset simply does not accurately portray their particular processes. For the last ten years Simulation Dynamics has been working with industries where the item paradigm falls short as a descriptive tool. This work has led to the development of a revolutionary set of simulation tools built on the Extend simulation engine.

1 INTRODUCTION

The concept of individual items stepping through a model is inadequate when the number of items gets too large or when there is no "item" to deal with (e.g. liquids, powders, and other bulk commodities). SDI developed two new discrete-event modeling paradigms to deal with these non-conforming situations. These two technologies were developed simultaneously in the mid-nineties, and rely on conceptually separating the item from its future event list. The first of these new technologies is known as 'discrete rate' and uses a flow paradigm for modeling material movement. Flow blocks adjust rates of flow rather than length of item delays. Events are posted when material in tanks reaches set trigger levels rather than when delays are completed. The second technology is an embedded database that allows 'things' to be modeled as data rather than either items or flow. Events are posted when data needs to be manipulated.

Imagine That, Inc. has incorporated these two technologies into its basic simulation offering as the 'Industry' portion of Extend + Industry.

1.1 Discrete Event and Discrete Rate

Let us look at a bottling line example to see the differences in discrete event models using items, flow, and pure data manipulations. Suppose we had a bottling line that could process 120 bottles every minute, and we wanted to process an order for 1000 cases of our favorite beer (24 bottles per case). The bottling line consists of a number of operations; for simplicity's sake we will only consider three: the filler, the sealer, and the case packer. While the line has a stated rate of 120 bottles per minute, that is actually the filler rate. The sealer can process 150 bottles per minute, and the case packer can handle eight cases per minute.

To model this example in the items paradigm (see Figure One below) we will consider each bottle to be an item, and the three operations as delay blocks with queues separating them. Each bottle is delayed in the filler for 0.5 seconds, the sealer for 0.4 seconds, and collected in the case packer until 24 have been accumulated and an item representing a case exits. To simulate the processing of 1000 cases will take approximately 40,000 simulation steps.

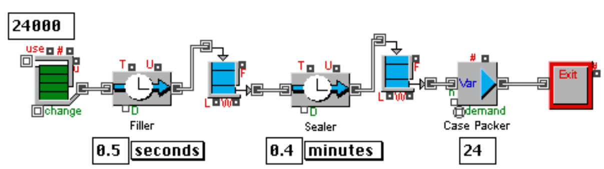

Figure 1: Bottling Line Modeled In Discrete Event Architecture

In the flow paradigm (see Figure Two below), we would represent the processing line as follows: the filler would be a constraint block with a maximum rate set to 120 per minute; the sealer would also be a constraint with its rate set to 150 per minute; and the case packer would be a converter whose output was 1/24 of its input, with a maximum output rate of eight per minute. To model the processing of 1000 cases, assuming no machine failures, would take two simulation steps.

Finally, simulating the same problem using the database paradigm (see Figure Three below) would work as follows. An order is presented for 1000 cases of our favorite

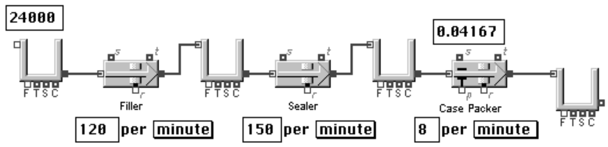

Figure 2: Bottling Line Modeled In Discrete Rate Architecture

Table(4) : Production Rates

	Process	Processing Rate
1	Filler	120
2	Sealer	150
3	Case Packer	8

Table(6) : Orders

	Order Number	Product	Quantity
1	977	Good Beer	1000
2	978	Better Beer	1000
3	979	Best Beer	1000

Figure 3: Bottling Line Modeled In Database

beer. We examine the possible lines that could processes the beer and determine that the best line can bottle at a rate of 120 bottles per minute, meaning the order will take 200 minutes to process. We set an event for 200 minutes in the future, and at that time decrement the beer inventory and increment the cases inventory by 1000 cases. This also takes only two simulation steps.

So which paradigm best fits the problem? The item paradigm causes the user to translate the processing rates (the language of the plant) to delay times (the language of the simulation toolset), and will take 20,000 more steps to process the run. The flow and database paradigms avoid the translation problem and process orders of magnitude faster. Clearly we would wish to use the item paradigm so that we could have time to enjoy one of the beers while running the model.

2 SUPPLY CHAIN MODELING ISSUES

The problems of modeling a supply chain are many and varied. Interestingly, when taken in small pieces, the problem is easy to mold into the classic item mentality of the discrete event architecture. For instance, a supply chain consists of inventories of "stuff" which, when aggregated into a group of cases, can be modeled as items. These items then get transported from place to place in trucks, trains, or planes, which can also be modeled as items.

A supply chain consists of too many things to be modeled individually. If you step back and look at a supply chain from 30,000 feet, you will see a multitude of materials flowing through it. Raw materials are being mined / harvested and moved to processing plants where they are merged into inventories. When needed, they are drawn to other facilities for conversion into products that are again stored in inventories until required. This chain may continue through many links before a consumer finally uses the end product. (Or it may continue even further as waste flows through its own recycling supply chain, re-entering the raw material inventory upstream.) While the material flows from raw materials to finished goods and final consumption, cash flows in the other direction, from consumer to raw material producers. A tremendous amount of information and data is required to describe these flows and the rules that govern them. Supply Chain Builder handles the structure and organization of this data and the problem of supply chains as a whole.

3 IMPORTANCE OF DATA MANAGEMENT

Supply Chain Builder organizes the masses of data associated with a supply chain into a structured database. The information contained in this database describes both the structure of the supply chain (its nodes and how they are connected) as well as the rules for movement within the supply chain (such as when and how much to order from whom). The entire supply chain is thus fully described in the database. All inventories, all resources, all locations, and all of the rules, policies, and procedures that govern movement within the supply chain are spelled out. The current state of the supply chain through time is described by the database as well, with all orders, shipments, and inventory levels detailed. At any given time in a supply chain simulation a full snapshot can be examined. Not only can the depletion of a given item at a given location be detected, but all pending orders for that inventory, and the status of shipments against those orders, can be investigated, leading directly to the root cause; the stock out occurred because a key shipment is waiting in a truck at the raw material supplier until the 'no less than full truckloads' policy has been met.

While many things flow through the supply chain, no items are required to simulate this movement. Everything happens in the database. Supply Chain Builder manipulates, creates, and destroys data as required to represent the movement of things through the supply chain. The discrete event clock is used to time events such as consumption, inventory checking, shipment creation, loading, travel times, resource availability changes, invoice processing, bank deposits and withdrawals.

Supply Chain Builder forces the modeler to focus on the data that will drive the model and the policies and processes that define the flow of information and material within it. This is in direct contrast to the focus on the mechanics of material movement in traditional discrete event tools.

Simulation models can help users gain an understanding of how their processes work and interact. The insight gained is one of the key benefits of a simulation and can be hard to achieve with problems as broad as the supply chain of a business. It is one thing to run a simulation and see that the configuration modeled achieved 94% customer satisfaction. It is yet another to discern what policies within the supply chain limited its performance. To assist users in learning from their supply chain model, Supply Chain

Builder comes with two key user interfaces, the SC Animator and the SC Investigator.

4 THE SUPPLY CHAIN BUILDER ANIMATOR AND INVESTIGATOR

The Supply Chain (SC) Animator provides a visual display of the movement of materials through the supply chain and the status of the supply chain nodes. The simulation can be paused at any time by clicking on a node to see which inventory has hit an alarm level or by clicking on a shipment between nodes to see what it contains. When the animator spotlights a situation that bears further investigation, the Supply Chain Investigator can be used to trace through the network to the underlying cause. See Figure Four below.

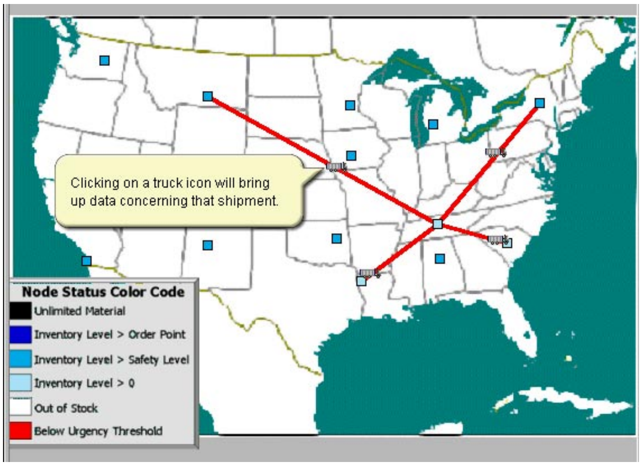

Figure 4: Supply Chain Animator

The Supply Chain Investigator helps navigate through the supply chain, tracking down the orders, shipments, and resources that are ultimately causing an alarm condition such as a stock out in a distribution center. Tracking down root causes of disruptions can be rather involved. The cause could be upstream of the alarm at a supplier's supplier; or the cause could be a downstream disruption or perturbance in the demand pattern for the product. It could be due to competition for a scarce resource in a production facility or a bottleneck operation. Whatever the cause, the Investigator enables it to be tracked and documented.

Supply Chain Builder (see Figure Five) leverages the powerful database embedded in Extend+Industry to provide a fast, efficient modeling tool for supply chains and problems that resemble supply chains.

5 PLANT BUILDER: SIMULATE MULTI-STAGE MANUFACTURING OPERATIONS

Plant Builder is a simulation software package for modeling multi-stage manufacturing operations. It is the only simulation tool that addresses the unique dynamics particular to high-speed manufacturing and process industries. The software's modular framework and wizards allow users to rapidly build custom models and database structures, enabling the modeler to quickly begin the analysis process.

Figure 5: Top Level of a Complex Supply Chain

Plant Builder evolved from SDI's extensive work in the food processing and consumer products industries. Plants in these industries are typically multi-stage (usually Making – Packing) plants, with raw material and in-process inventories typically managed in some form of distribution system that resembles a tank farm, be it a set of silos, bins, or portable totes, as shown in Figure Six below. Issues in these plants often revolve around schedules, new product introduction, or other processing changes. Plant Builder is a set of tools that allows the modeler to rapidly explore the issues in these plants that impact plant throughput.

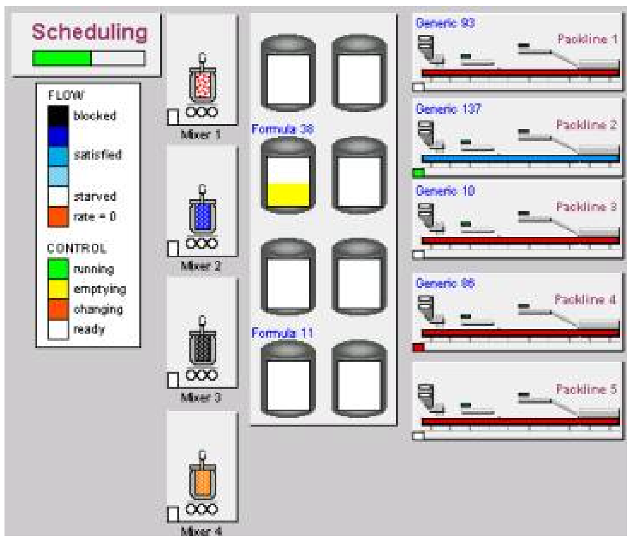

Figure 6: Multi-Stage Plant With Embedded Finite Capacity Scheduler

A structured database is at the heart of Plant Builder, just as it was in SCB. The structure of the plant and its policies is captured in the database and used not only to drive the model, but to build the model as well. Unlike SCB, Plant Builder does require the modeling of the mechanics of material movements within the plant. However, these plants rarely lend themselves to the item paradigm, and so the Discrete Rate element of the E+I platform is exploited.

The Plant Builder Assistant (PBA) initiates the model building process by asking the modeler a series of questions about what materials are moved through the plant, what types of systems process each material, and how the systems are linked together. The information gathered during this process is translated into the model database that can then be printed out or exported into Excel for distribution and data entry.

The database built by the Assistant describes the structure of the plant in great detail, and is sufficient to build an initial model of the plant. At the press of a button, the PBA will build the plant model using hierarchical blocks (h-blocks) from a pre-built library of production systems, distribution systems, warehouses, and schedulers. While building the model, the PBA labels all blocks and sets all of the block dialog parameters. A working model is close at hand.

What remains in the modeling effort is the details. The actual data needs to be entered into the database, and appropriate details added into the production system sub-models. Many of these model details can be preserved from model to model; soon the PBA will be able to use company specific h-blocks to build the model rather than just the 'generic' h-blocks currently provided.

Plant Builder models the flow of materials from raw materials through conversion into finished products. The Warehouse components allow the modeling of the finished goods inventory as well as the consumption of finished goods by downstream customers. Production schedules drive the models, and these schedules can be input into the model via the database, built from a requirements table that reflects ordering patterns, or driven by restocking orders from the Warehouse. In addition, alternative scheduling methods can be modeled to quantify the impact of choosing different production processes as the driving bottleneck of the plant.

6 SDI INDUSTRY: GENERAL PURPOSE SIMULATION WITH UNPARALLELED FLEXIBILITY

Simulation Dynamics Inc. built the 'Industry' piece of this amazing and versatile simulation toolset. Industry adds a powerful embedded database to Extend that is the basis for Supply Chain Builder as well as Plant Builder. It also adds the ability to model continuous flows within the discrete event world using the unique 'Discrete Rate' technology.

The embedded database was designed from the ground up as a tool for simulationists. This resulted in some rather unique database capabilities that only a simulation might require. Individual data cells within a database table can be defined as random cells. Each time a random cell (see Figure Seven below) is queried, it reports a different value based on the cell's defined distribution. This allows data such as machine failures, changeover times, forecast variability, and processing times to be defined as random values with single lookups (rather than multiple lookups for each parameter).

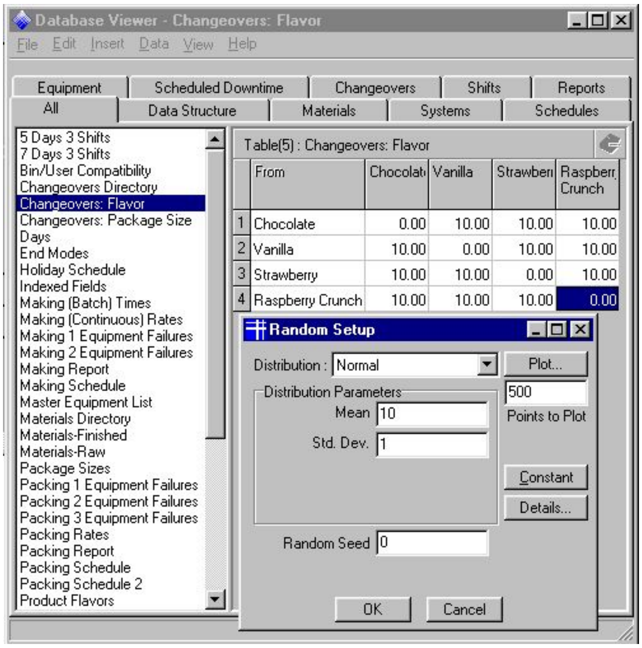

Figure 7: Database Table With Randomly Defined Cells

Data tables can also contain mixed fields, allowing strings, real numbers, and integers to be displayed together in the same table. Fields can also have relationships with other fields that limit the possible values for that field. This unique relationship allows the database to display string names but return integer values for the model to process. This duality allows for powerful constructs such as string attributes, where items can have attributes such as 'color', that display 'RED' but report a value for red, say three (3), for the model to process and use.

This capability is exploited by SCB to allow inventories, resources, and locations to have real names for display purposes, but mathematical representation for analytical computations.

7 CONCLUSIONS

Within the discrete event paradigm, there are other options – discrete rate and database driven models. In the classic discrete event simulations, all the action takes place with items. When an item moves, there is a step in the simulation. With SDI's new tools, things can move in the model without items. Things can flow through the model as rates or be manipulated in the database as data elements.

Discrete Rate is a technology driven by early experience trying to model such commodities as dog food and Cheerios in the discrete event world. Bulk commodities such as these simply do not lend themselves to the classic discrete event item paradigm. Aggregation techniques are inefficient and inaccurate. The best method to deal with such products is as bulk commodities that flow like water through the processes. This is borne out by the way people in the plant talk about these processes –they speak in terms of production rates – so many pounds per hour, etc. The same is true on the packaging side of these plants. Although it is not difficult to visualize a box of cereal as an item, the packing lines language is also a language of rates, and is more naturally modeled as flow.

SDI's embedded database technology is largely the result of building models with huge data requirements. Nowhere is this requirement more evident than in the world of supply chain simulation. Here, even a fairly simple model could generate hundreds of thousands of orders, shipments, invoices, etc. Modeling all these transactions (let alone the corresponding material flows) as items would bring most simulation platforms to their knees, causing extremely long run times. Data manipulations are fast and efficient, and large supply chain models can be run in minutes rather than hours. This means more runs, more analysis, and more insight into what is happening within the dynamic systems. This translates directly into improved understanding, and understanding is what sparks the "ah-ha" moments that lead to true process improvement.

For the past twelve years, Simulation Dynamics has delivered powerful solutions to customers in a wide spectrum of challenging simulation modeling projects, building on the popular Extend simulation platform.

The block architecture and development system of Extend has inspired Simulation Dynamics to evolve the state of the art in modular model building, and to refine the system of blocks for addressing classes of problems in a general and flexible way.

The SDI Product Suite provides tools that allow rapid modeling of systems, from simple lines to plants and entire enterprises. For enterprise modeling, the tools can be used to assemble vertical model architectures that have varied degrees of detail as necessitated by the problem.

Industry's suite of tools has significant problem-solving potential, and they can be exploited by practitioners of simulation at any level of interest or proficiency.

AUTHOR BIOGRAPHIES

RICHARD A. PHELPS is the Product Manager at Simulation Dynamics. Mr. Phelps' industrial experience prior to joining SDI includes time in the cutting tool and aluminum industries. He received his B. S. in Industrial and Systems Engineering from the Georgia Institute of Technology where he is a member of the Council of Outstanding Young Alumni. His email and web addresses are `<Phelps@SimulationDynamics.com>` and `<www. SimulationDynamics.com>`.

DAVID J. PARSONS is a principal of Simulation Dynamics. His experience with simulation began in 1965 with experiments in the use of natural selection algorithms to evolve architectural designs. During the 1980's he designed, built and operated several dairy-processing plants using simulation of key systems as an integral tool for design, value engineering, and troubleshooting. Mr. Parsons received a B. A. from Harvard College and a Master of Architecture degree from the Harvard School of Design. His email and web addresses are `<Parsons@SimulationDynamics.com>` and `<www. SimulationDynamics.com>`.

ANDREW J. SIPRELLE is President and founder of Simulation Dynamics, a firm that provides simulation consulting, training, and custom models. Mr. Siprelle's industry experience includes creation and analysis of models for strategic and capacity planning, market analysis, and the application of industrial statistics. Applications of simulation have been in the areas of business, manufacturing, government and service operations. Before starting Simulation Dynamics, Inc. he worked at the Aluminum Company of America. He received his B. S. in Industrial Engineering and Operations Research from Virginia Polytechnic Institute. His email and web addresses are `<Siprelle@SimulationDynamics.com>` and `<www. SimulationDynamics.com>`.