

DATABASE-INTENSIVE PROCESS SIMULATION AT THE Y-12 NATIONAL SECURITY COMPLEX

Reid Kress
Karen Bills
Jack Dixon
Richard Rinehart

Information Technology
National Nuclear Security Administration (NNSA)
BWXT Y-12
Oak Ridge, Tennessee 37831-2009 U.S.A.

ABSTRACT

The NNSA's Y-12 National Security Complex is a manufacturing facility operated by BWXT Y-12. Y-12's missions include ensuring the US' nuclear weapons deterrent, storing nuclear materials, and fueling US naval reactors. As a consequence of these missions, Y-12 makes dozens of products, having hundreds of parts, each with many different process steps associated with manufacturing components, building sub-assemblies, or assembling final products. Y-12 also disassembles weapon components to support stockpile reduction efforts and retrieve high-value materials and parts. These efforts must be coordinated not only within the Y-12 complex but also within the nation-wide nuclear weapons complex. Y-12 relies heavily on simulation models to evaluate the impact of plant changes over multiple programs and products, spanning multiple-years, and involving multiple interacting organizations. To maintain these simulations with current data and build and execute them rapidly, Y-12's simulation modeling group relies heavily on database-intensive simulations described in this paper.

1 INTRODUCTION

The Y-12 National Security Complex is a premier manufacturing facility dedicated to making our nation and the world a safer place. Operated by BWXT Y-12 for the National Nuclear Security Administration, Y-12 plays a vital role in the Department of Energy's Nuclear Weapons Complex. Y-12 helps ensure a safe and reliable U.S. nuclear weapons deterrent, retrieves and stores nuclear materials, fuels the nation's naval reactors, and performs work for other government and private-sector entities. As a consequence of this mission, Y-12 makes dozens of products, having hundreds of parts, each with many different process steps associated with manufacturing

components, building sub-assemblies, or assembling a final product. In addition, Y-12 disassembles weapons in order to support stockpile reduction efforts and to retrieve high-value materials and components. Y-12's technology-based missions can be distributed into the following broad categories:

- Production and rework of complex nuclear weapon components and secondaries;
- Receipt, storage, and protection of special nuclear materials;
- Quality evaluation and enhanced surveillance of the nation's nuclear stockpile;
- Dismantlement of weapon secondaries and disposition of weapon components;
- Prevention of the spread of weapons of mass destruction; and
- Support work for DOE and other federal agencies.

All of these efforts must be coordinated not only within the Y-12 complex (811 acres, 500 buildings, 7 million square feet of laboratory, machining, dismantlement, and research and development areas), but also within the nation-wide nuclear weapons complex. In addition, DOE and other government managers often request schedule evaluations, equipment justifications, or new facility plans that take a multi-year look at schedule compliance, payback, and implementation. (In fact, sometimes the horizon is multi-decade in the context of very expensive new facilities or upgrades.) The only way for Y-12 managers and planning organizations to evaluate the impact of plant changes over multiple programs and products, spanning multiple-year production runs, involving multiple interacting organizations is to develop detailed simulations of the bulk of Y-12's processes. In order for these simulations to be maintained with current data and to be executed in a timely manner, Y-12's

simulation modeling group has placed a heavy reliance on database-intensive simulations. This paper describes Y-12's use of database-intensive discrete-event simulation focusing on some of the basic approaches as well as our choices in software. The goal of the paper is to illustrate some ideas for improving the utilization of databases in simulation models and how this greater use of a database will streamline a model making it easier to build, debug, maintain, execute, modify, and use as an analysis tool. The paper is organized as follows. Y-12's simulation modeling approach relative to database utilization is described in the next section, some basic results are shown in the third section, and conclusions are made in the fourth section.

2 APPROACH

Simulation modeling at Y-12 is used as a production support tool. As a consequence, commercial off-the-shelf (COTS) tools are preferred to custom software. Y-12's simulation modeling group relies on three primary COTS packages. These are EXTEND™, Supply Chain Builder™, and FlexSim™ (refer to references [1], [2], and [3] respectively). With each of these tools, Y-12's

simulation modeling group relies heavily on database-intensive models. An example of each is presented below.

2.1 "Jobshop" Discrete-Event Model

Y-12's production mission involves hundreds of processes. Y-12's simulation modeling group has spent over five years developing a generic "Jobshop" model in EXTEND™ that can be used to model any process or a collection of processes that can be described by a set of finite-time operations that require identifiable resources. Figure 1 shows a typical front page of one version of the "Jobshop" model.

Consider the following easy-to-understand example as a context within which to present Y-12's "Jobshop" model. Suppose Y-12 wished to build bicycles from a pre-defined set of plans. Each bicycle requires a frame (made of two parts, A and B) and two wheels (made of a tire, rim, and 60 spokes). In addition to assembling bicycles from parts, in this fictitious example, Y-12 also takes old bicycles apart and either directly reuses some of the parts in new assemblies or repairs/refurbishes the parts if possible and then reuses them in new assemblies.

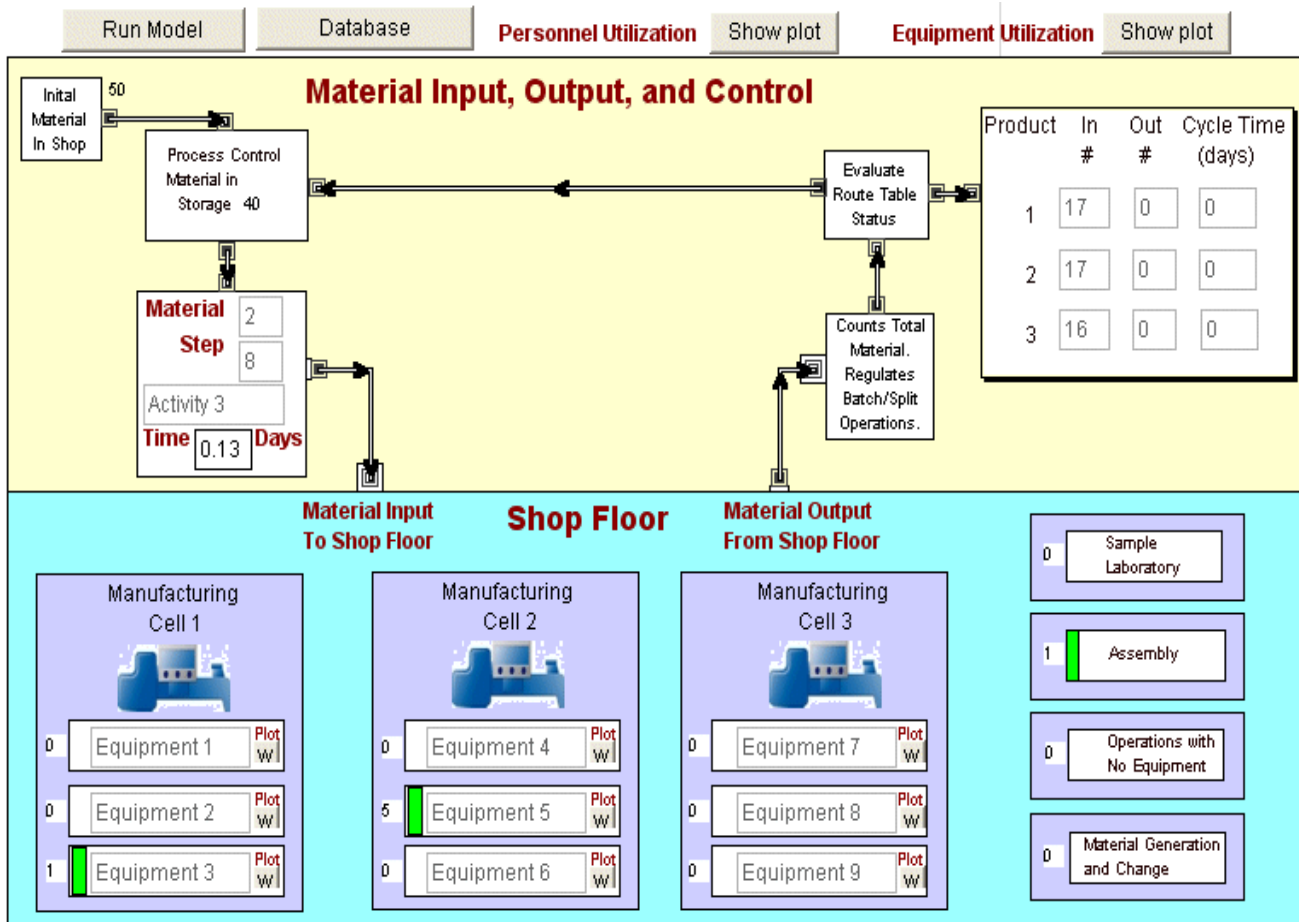


Figure 1: Front Page of a Simplified Version of the "Jobshop" Model

The relational database for this example might include the following tables: Programs, Products, Components, Routes, and Sub-routes. Database tables for this model would be tied together as follows.

1. The program table would contain product information as well as a schedule for each product.
2. The product table would contain components as well as route tables for each component if they are not available from inventory.
3. The route tables would have sequential steps of operations, specified operation duration (constant or distribution function), and resource requirements (e.g. equipment, labor, funds, facility, etc.)
4. A route table for an assembly might be placed on the initial part of the assembly and sub-route tables placed on each part to account for the manufacturing steps.

The following simple example illustrates the construction:

1. Consider two Programs: Bicycle Manufacturing Campaigns 1Q06 and 2Q06;
2. Consider three Products: 10-speed, 3-speed, 1-speed bicycles; and
3. Consider the Components: Frame parts A and B, Wheel (1 Tire, 1 Rim, and 60 Spokes).
4. Consider simple route and sub-route tables consisting of the following steps:
Obtain part/subassembly,
Assemble/Disassemble/Repair, Inspect, Ship.

This example is placed into one possible database format in tables 1, 2, 3, and 4 below.

Table 1: Example “Program” Table Showing Products and Schedule

Program	Product	Quantity	Schedule	Description
1Q06	10-speed	20	Oct 1, 2005	1 st quarter 10-speed bicycle production
1Q06	3-speed	15	Oct 15, 2205	1 st quarter 3-speed bicycle production
2Q06	10-speed	30	Jan 1, 2006	2 nd quarter 10-speed bicycle production
2Q06	3-speed	15	Jan 15, 2006	2 nd quarter 3-speed bicycle production
2Q06	1-speed	10	Feb 1, 2006	2 nd quarter 1-speed bicycle production

Table 2: Example “10-Speed Bike Product” Table Showing Components

Component	Quantity	Assembly Route Table	Making Route Table
10-speed	1	Assemble 10-speed	
Frame A	1		Make Frame Part
Frame B	1		Make Frame Part
Tire	2	Assemble Wheel	Make Tire
Rim	2		
Spokes	120		

Table 3: Example “Assemble 10-Speed Bike” Route Table for the Case When Operations Require Only a Single Type of Equipment and a Single Type of Labor Resource

Operation	Duration	Units	Equipment Resource	Labor Resource
Obtain Parts	0.5	Hour	Dolly	Material Handler
Assemble Parts	1	Hour	Assembly Stand	Assembly Tech
Inspect	10	Minutes	Assembly Stand	Inspector
Ship	20	Minutes	None	Shipping Clerk

Note: If multiple resources of the same type are required, then resource tables are established and resource requirements listed sequentially; that is, one operation would cover several lines in an “Equipment Resource Requirements” table as shown below.

Table 4: Example “Equipment Resource Requirements” Table for the Case When Operations Require More Than a Single Type of Equipment Resource

Operation	Resource	Number
Assemble 10-Speed	Assembly Stand	1
Assemble 10-Speed	12-mm Wrench	1
Assemble 10-Speed	14-mm Wrench	1
Assemble 10-Speed	Medium Phillips Screw Driver	2

Designing the model and organizing the data in this manner provides a number of advantages. First, and extremely important to execution time, the number of items in the model can be minimized. One item can be used to represent an entire assembly. The item simply goes through all of the steps normally associated with an assembly including the steps for manufacturing parts. For example, if one had 5 programs, with 10 products per program, and each product has 100 parts and one developed a model using a single item per part as opposed

to a single item per product, then one would have 5000 items versus 50 items. For a base EXTEND™ model (generator, queue, activity, exit), figure 2 shows that execution time would increase by almost an order of magnitude. This is insignificant when a typical simulation takes 0.01 seconds but is extremely significant when a single simulation takes almost an hour; especially when multiple runs are desired to strengthen output statistics.

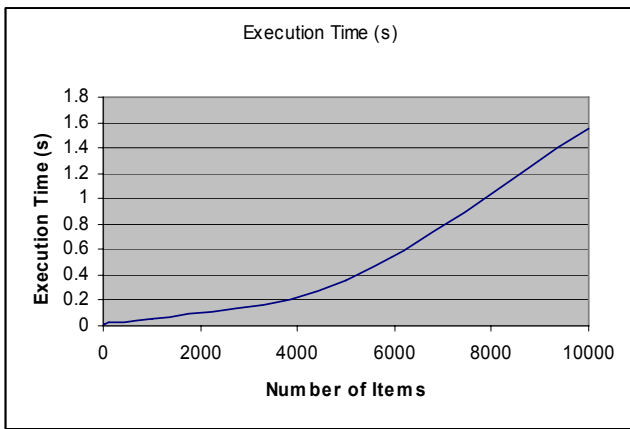


Figure 2: Execution Time Versus Number of Items in a Base Extend™ Model (generator, queue, activity, exit) Running on a PC with a 2.4ghz Xeon Processor Having 1.0 GB of RAM

Results accounting is also done exclusively in the database. Machine utilization, Work-In-Process (WIP), cycle time (c.f. reference [4]), throughput rate, and material balances are all calculated in the database. Adding and modifying products and routes (e.g. creating route tables to reflect the addition of new capabilities in the future) is simplified because one need only copy and rename an existing route table, make the anticipated changes (e.g. add a new machine, reduce a processing time to reflect gains in efficiency, reduce a resource requirement, etc.), and assign the new route table to the affected products. Rapid replication of model blocks to facilitate development is also possible. For example, in figure 1, equipment blocks can be copied and added to increase the number of different pieces of equipment from nine to any number desired and only equipment names and characteristics need to be added to the equipment list in the defining database table. Rapid modification of model inputs to simulate new processes is made easy with a database-intensive simulation. For example, suppose one wanted to model the pre-making of spokes in the bicycle example. One simply adds a line to table 1 to make 10,000 spokes, adds a “Make Spokes” route table at the desired level of detail (in this case just one step), and includes the new operations and equipment used in the “Make Spokes” route table in their respective lists. Tables 5, 6, and 7 illustrate the basic changes required to implement this modification.

Table 5: Addition to Table 1 (Program) to Pre-Make a Year’s Supply of Spokes

Program	Product	Quantity	Schedule	Description
1Q06	Spokes	10,000	Oct 1, 2005	Entire year’s spoke production

Table 6: Example “Spokes Product” Table

Component	Quantity	Assembly Route Table	Making Route Table
Spokes	10,000		Make 10,000 spokes
Raw Material	1000 lbs		Process spoke metal

Table 7: “Make Spokes” Route Table for the Case When Operations Require Only a Single Type of Equipment and a Single Type of Labor Resource

Operation	Duration	Units	Equipment Resource	Labor Resource
Make 10,000 Spokes	1	Week	Drawing Machine	Material Handler

Finally, easy storage of model outputs for what-if evaluation and future comparisons is possible when results are stored in a large database (scenario management).

2.2 Integrated Resource Planning Model

The Integrated Resource Planning Model (IRPM) is created in Supply Chain Builder™ COTS software from Simulation Dynamics Inc [2]. This discrete-event model is completely database driven. There are no items at all in the model [3]. The model has a limited number of blocks with a large amount of functionality programmed into each block. The front page of a Supply Chain Builder model that will simulate the bicycle production example discussed above is shown in figure 3.

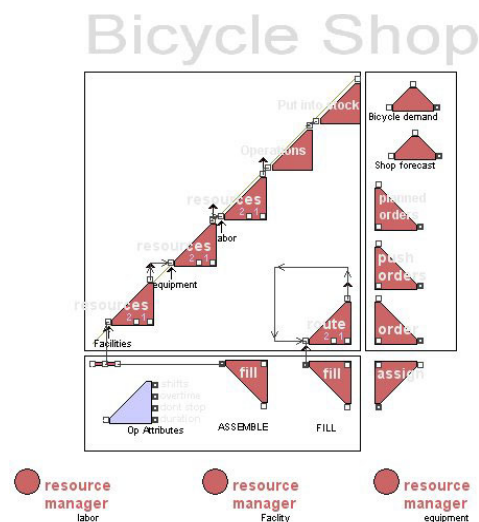


Figure 3: Front Page of The Integrated Resource Planning Model (IRPM) Simulation of the Bicycle Shop Example

Being completely database driven has an important additional benefit. It facilitates the development of a custom Graphical User Interface (GUI) for the simulation in Microsoft Access™. This GUI serves several purposes. First, it establishes a connection between the model database and existing corporate databases such as scheduling data, human resource data, and equipment availability. This keeps model data current.

Critical measures such as utilization, resource requirements, operation costs, etc. are calculated either in the model or in the Access™ GUI, depending on which is easier for the programmer to implement. New programs, products, processes, operations, resource requirements, etc. can be added in Access™ and are automatically transferred to the supply chain model. A run button starts the model and, upon completion, simulation data are transferred back to Access™. Post-processing and data analysis are done in Access™. One version of the IRPM GUI is shown in figure 4.

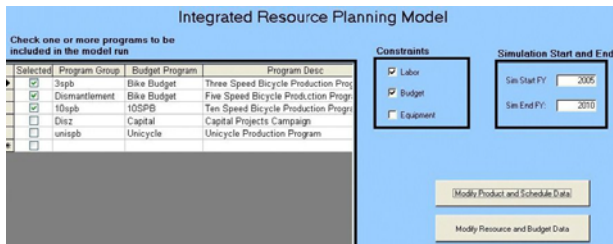


Figure 4: The Custom-Designed Access™ Graphical User Interface (GUI) for the Integrated Resource Planning Model (IRPM) That Keeps Data Current and Handles Data Analysis and Scenario Management

Scenario management capability is also provided in the Access™ GUI as well as reporting and graphing capabilities.

The primary table describing the details of a model in the IRPM is the Bill Of Materials (BOM) table. Unlike a traditional BOM, the IRPM BOM not only has a materials list, but it also has the operations associated with the materials. A BOM for the bicycle model example is shown in figure 5.

The BOM can represent an assembly (see assemble bicycle at the bottom of the table) and it can also represent disassembly, inspection, repair, and other operations. As long as there is an input and an output to an operation it can be represented in the BOM. For example, to simulate a movement of material from location A to B simply have a BOM operation that has as an operation called “Move from A to B” and has an input “Material at A” and an output “Material at B.” In addition, having the model built within data tables enables further investigation of the supply chain via ordering policy and order amounts in the inventory table.

Table(1948) : Bill of Materials						
	Operation	Part	Quantity	Output	Output Quantity	Rule Primary
1	Take bike apart		0.00	Untested frame	1.00	0 Yes
2	Take bike apart		0.00	Untested wheel	2.00	0 Yes
3	Take bike apart	Broken bicycle	1.00		0.00	0
4	Frame is Scrap		0.00	Scrap frame	1.00	0 Yes
5	Frame is Scrap	Untested frame	1.00		0.00	3
6	Frame is broken		0.00	Broken frame	1.00	0 Yes
7	Frame is broken	Untested frame	1.00		0.00	4
8	Frame is OK		0.00	OK frame	1.00	-1
9	Frame is OK	Untested frame	1.00		0.00	0
10	Repair frame		0.00	OK frame	1.00	-1
11	Repair frame	Broken frame	1.00		0.00	0
12	Assemble frame		0.00	OK frame	1.00	0 Yes
13	Assemble frame	Frame part A	1.00		0.00	0
14	Assemble frame	Frame part B	1.00		0.00	0
15	Wheel is Scrap		0.00	Scrap wheel	1.00	0 Yes
16	Wheel is Scrap	Untested wheel	1.00		0.00	1
17	Wheel is broken		0.00	Broken wheel	1.00	0 Yes
18	Wheel is broken	Untested wheel	1.00		0.00	2
19	Wheel is OK		0.00	OK wheel	1.00	-1
20	Wheel is OK	Untested wheel	1.00		0.00	0
21	Repair wheel		0.00	OK wheel	1.00	-1
22	Repair wheel	Broken wheel	1.00		0.00	0
23	Assemble wheel		0.00	OK wheel	1.00	0 Yes
24	Assemble wheel	Spoke	60.00		0.00	0
25	Assemble wheel	Rim	1.00		0.00	0
26	Assemble wheel	Tire	1.00		0.00	0
27	Assemble bicycle		0.00	Bicycle	1.00	0 Yes
28	Assemble bicycle	OK frame	1.00		0.00	0
29	Assemble bicycle	OK wheel	2.00		0.00	0
30			0.00		1.00	0
31		Pgm 1 Budget	1.00		0.00	0
32			0.00		1.00	0
33		Pgm 2 Budget	1.00		0.00	0

Figure 5: The Bill Of Materials (BOM) Table in the Integrated Resource Planning Model (IRPM)

2.3 Three-Dimensional Discrete-Event Simulation

Y-12’s simulation modeling group is beginning to use three-dimensional, discrete-event simulation to model some of its processes, especially those processes that have a significant amount of logistics issues and would benefit from spatial and geometric analyses. Y-12’s modeling group has selected FlexSim™ [4] as the modeling tool for three-dimensional, discrete-event simulation. An example of a simple three-dimensional, discrete-event simulation is shown in figure 6.

Three-dimensional, discrete-event simulations help the modeler visualize the interaction of products, components, machines, and resources and can identify problem areas not evident in a non-visual model. Again, heavy reliance is placed on tabular data to enable rapid model updating and modification. In addition, FlexSim interfaces easily with Microsoft Excel™.

3 RESULTS

3.1 “Jobshop” Discrete-Event Model

Typical simulations for a large Y-12 “Jobshop”-type model (5 programs, 10 products per program, 50 parts per product, 1 year simulation time) is on the order of 1 hour

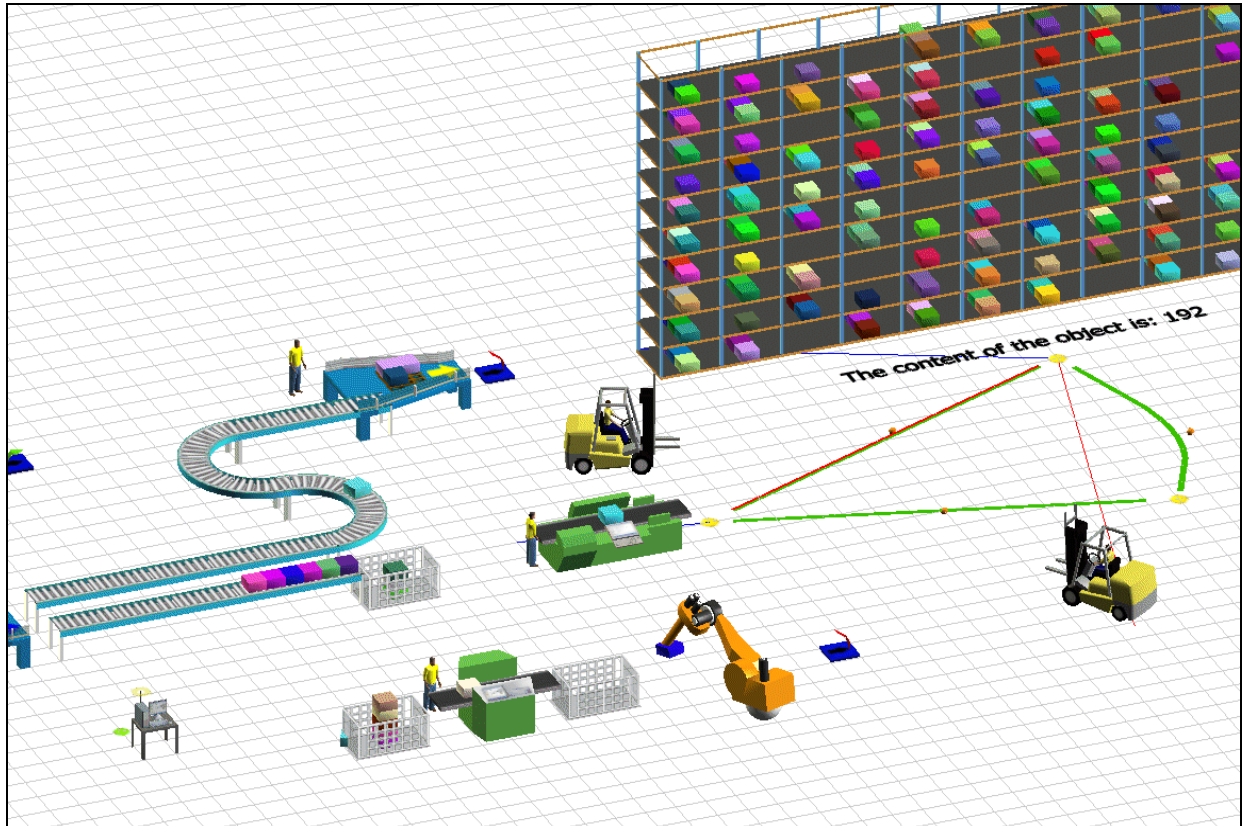


Figure 6: Example of Three-Dimensional Simulation Using Flexsim™

depending upon the machine capabilities and other shared activities. Typical results illustrating bicycle model plant capacity are shown in figure 7.

Note that this result indicates a problem with the bicycle factory since inventory is building and cycle time is increasing.

3.2 Integrated Resource Planning Model

Typical runs for a large Y-12 “IRPM”-type model (3 programs, 1 products per program, 50 parts per product, 10 year run) is on the order of 5 minutes depending upon the machine capabilities and other shared activities. Many of the intermediate results in the IRPM are stored in database tables such as the shipments table shown in figure 8.

The tabular results are automatically transferred back to the GUI where reports and graphs are limited only by the capabilities of Microsoft Access™.

Major modifications to both the “Jobshop” and the IRPM can be made in similar time frames. For example, adding a product with on the order of 50 parts can be completed in a day. What-if analyses can be done “on the fly” with minor modifications (e.g. changing resources requirements, changing resource availability, adding route steps, etc.) taking minutes.

4 CONCLUSION

Heavy reliance upon a database is critical to the success of Y-12’s simulation models for the following reasons.

- A model database facilitates rapid modification of simulations including changes in schedules, program requirements, product components, manufacturing steps, equipment, operations, resource requirements, resource availability, etc.
- A model database enables modelers to connect with existing enterprise databases to improve data fidelity both in terms of actual values and the timeliness of data updates.
- A model database streamlines the simulation model and enables very large, multiple-program, multiple-year models to be executed many times to provide statistical confidence in a time frame that is acceptable to project management.
- A model database provides an excellent means to store and retrieve model configurations for scenario management and in support of future auditing requirements.
- A model database enables rapid modification of model inputs and easy storage of model outputs for what-if evaluations.

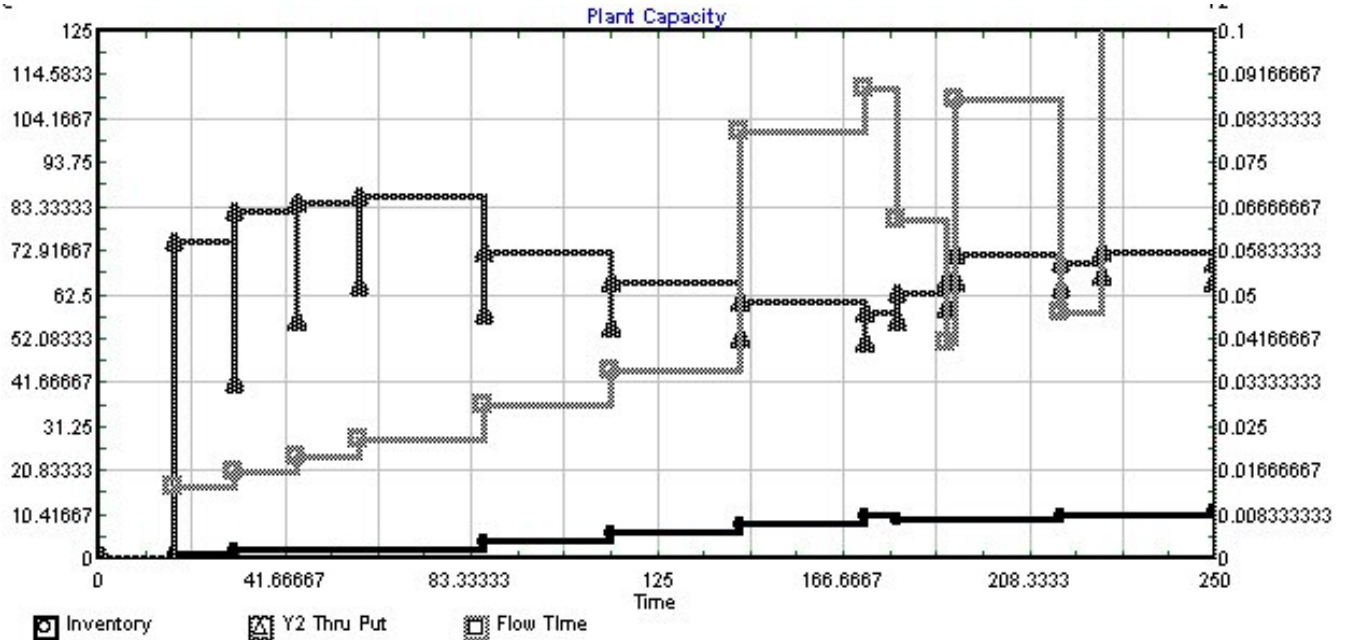


Figure 7: Example Results from a “Jobshop”-Type Model of the Bicycle Factory Showing Inventory (i.e. work in process [5]), Throughput, and Cycle Time (Note: “Y2” on the throughput indicates the respective curve’s axis is the second Y-axis.)

Table(92): Shipments										
Shipment Number	From Location	To Location	Priority	Amount In	Amount Out	Item Type	Item	Operation	Current Location	
1	1 Supplier	Bicycle Shop	Regular	1.00	1.00	Material	Broken bicycle		Bicycle Shop	
2	2 Bicycle Shop	Bicycle Shop	Regular	1.00	1.00	Operation		Frame Repair Ship	Bicycle Shop	
3	3 Bicycle Shop	Bicycle Shop	Regular	1.00	1.00	Operation		Take bike apart	Bicycle Shop	
4	4 Supplier	Bicycle Shop	Regular	1.00	1.00	Material	Broken bicycle		Bicycle Shop	
5	5 Bicycle Shop	Bicycle Shop	Regular	2.00	2.00	Operation		Assemble frame	Bicycle Shop	
6	6 Bicycle Shop	Bicycle Shop	Regular	4.00	4.00	Operation		Assemble wheel	Bicycle Shop	
7	7 Bicycle Shop	Bicycle Shop	Regular	1.00	1.00	Operation		Take bike apart	Bicycle Shop	

Figure 8: The Shipments Table in the Integrated Resource Planning Model (IRPM) Bicycle Shop Example

- A model database improves model robustness and makes debugging easier by consolidating results and reducing the number of items and associated item logic.

Y-12’s simulation modeling group has built simulations of hundreds of processes over the past ten years and has come to rely on database-intensive simulation models for most of its large simulations. This reliance has continued so that even “normal” discrete-event models such as the “Jobshop” approach the Supply Chain Builder™ concept where most of the model information

resides in the database and the discrete-event simulation has no items.

REFERENCES

<<http://www.imaginethatinc.com/index.html>>, last accessed March 27, 2006

<<http://www.simulationdynamics.com/>>, last accessed March 27, 2006

Phelps, R., Parsons, D., and Siprelle, A., “Non-Item Based Discrete-Event Simulation Tools,” *Proceedings of the 2002 Winter Simulation Conference*, pp. 182-186.

<http://www.flexsim.com/>), last accessed March 27, 2006

Hopp, W. and Spearman, M., *Factory Physics*, McGraw-Hill, 2001.

AUTHOR BIOGRAPHIES

REID KRESS is a simulation modeling engineer in the Technical Computing Division at the National Nuclear Security Agency's Y-12 Plant. He has a BS, MS, and PhD in Mechanical Engineering. His PhD at the University of Arizona was completed under a University Fellowship. Besides working at Y-12, he has taught in the Mechanical, Electrical, and Industrial Engineering Departments at The University of Tennessee and has been a robotics research engineer at the Oak Ridge National Laboratory. He holds two US patents and was awarded the "Excellence in Teaching" award by his students in 2001-2002. His research interests include simulation modeling, systems modeling, controls, vibrations, and applied mathematics. He has six very wonderful, and somewhat demanding, kids so he is best reached via e-mail at kressrl@y12.doe.gov.

KAREN BILLS received a B.S. degree in mechanical engineering and a M.S. degree in industrial engineering, in 1980 and 1993, respectively, both from the University of Tennessee, Knoxville. She has worked on simulations of hardware response to earthquakes, graphical simulation of naval and nuclear designs, and robotic simulation for deactivation of the Department of Energy (DOE) Facilities. She has spent 25 years at the Oak Ridge DOE facilities and

is currently working on decision support process simulations for modernization of the Y12 facility.

Vocation includes innovative ways to display and analyze enormous amounts of disparate data to create cohesive information for exchange of ideas in aligning and meeting goals. She may be reached via e-mail at billskc@y12.doe.gov.

JACK DIXON is a computer engineer and project manager in the Information Technology department at BWXT Y-12. He has been actively involved in developing simulation modeling capabilities to support manufacturing operations at the Y-12 National Security Complex. He received his B.S. in Mechanical Engineering at Northern Arizona University and a M.E. in Chemical Engineering at the University of Idaho. His email address is dixonjr@y12.doe.gov.

RICHARD RINEHART is a simulation specialist in the Technical Computing Division at the National Nuclear Security Agency's Y-12 National Security Complex. He joined the Y-12 Complex in 1986, and has specialized in software application design and development. Most recently, he has lead the development and implementation of a decision support application using a Supply Chain Analysis simulation engine. He received his BS in Computer Science from East Tennessee State University and has performed graduate work in Management Science and Statistics at the University of Tennessee. His e-mail address is rinehartrd@y12.doe.gov.