Proceedings of the 1984
Winter Simulation Conference
S. Sheppard, U. Pooch, D. Pegden (eds.)

INTERACTIVE SIMULATION GRAPHICS WITH INSIGHT

C. H. Imhoff, Research Engineer; C. J. Hostick, Research Engineer; L. W. Cannon, Computer Scientist
Pacific Northwest Laboratory
Richland, Washington 99352

The purpose of this paper is to describe the benefits of graphics in recent computer simulation analyses at Battelle's Northwest Laboratory and to present the graphics package developed at Battelle to perform these analyses. These graphics capabilities are part of a collection of simulation tools built over the last three years by Battelle staff to aid the development and understanding of simulation models. The system is called INTERACTIVE SIMULATION WITH GRAPHIC TOOLS (INSIGHT) and is comprised of the following elements:

- Code Development Tools
- Preprocessor
- Graphic editors
- Run-time library
- Postprocessor.

This discussion will focus upon the structure and capabilities of the graphics elements of INSIGHT and upon the usefulness of these graphics capabilities in recent simulation projects.

The paper is divided into three segments. First, a brief description of a recent simulation project performed for the U.S. DOE is provided, including example screens of the interactive graphics developed by INSIGHT tools. The second section discusses the structure and use of INSIGHT; special emphasis upon the graphics editors is presented. These sections are followed by a summary of the work to date, including both the benefits of interactive simulation graphics to work at Battelle Northwest (BNW) and the challenges to making these tools more useful to future simulation analyses.

EXAMPLE APPLICATIONS OF INTERACTIVE SIMULATION GRAPHICS

The interactive graphics capabilities of INSIGHT were recently applied to the Monitored Retrievable Storage (MRS) Program being performed at BNW for the U. S. Department of Energy (DOE). The purpose of the MRS Program is to evaluate designs for a receiving and handling MRS facility to package spent fuel and high-level radioactive wastes for storage. A simulation model of the proposed facility was built using the SLAM II* simulation language, and the model was analyzed using the capabilities of INSIGHTs graphics editors.

The government-owned MRS Facility will be designed to receive spent nuclear fuel and other contaminated waste. Waste will arrive to the Receiving and Handling (R&H) area of the MRS facility by truck and by rail cars at an approximate rate of 1,100 shipping casks per year. The R&H facility layout consists of a cask handling area and a hotcell area for the remote handling of spent fuel. Upon arrival, shipping casks are removed from the transport vehicle (truck or railcar) by an overhead bridge crane, and placed on a self-powered cart in the cask handling area. Following inspection and preparation activities, the cask cart enters the hotcell and the cask is unloaded. The empty shipping cask is then returned to the transport vehicle and removed.

The key design considerations of the R&H Facility layout analysis are listed below:

- Location and number of transport vehicle preparation areas
- Overhead crane count, capacity, and range
- Flow of cask carts in and out of the hotcell.

Cask handling and hotcell operations were modeled using SLAM II for several alternative facility layouts. The meantime between transport vehicle arrivals and the duration of processing activities were assumed to be exponentially distributed. The operation of each alternative facility was simulated on a minute-by-minute basis for time periods from one to three years. The SLAM summary report provided detailed statistics on expected equipment utilization, queue characteristics, and shipping cask turnaround time. These statistics formed the basis for the following design recommendations:

- Three overhead cranes are sufficient to service the cask handling area
- Each cart carrying empty shipping casks can block access to the hotcell for extended periods, resulting in under utilized hotcells. This situation can be eliminated by rearranging cask cart flow.
- The addition of transport vehicle preparation areas significantly reduces shipping cask turnaround time.

Development of graphics using INSIGHT focused on the cask handling portion of the R&H Facility because the major design considerations mentioned above were elements of the cask handling area. The design of the graphic display attempted to show the viewer 1) the potential idleness of the hotcell due to blocking of the hotcell access by carts carrying empty shipping casks and 2) the large queues of transport vehicles when insufficient transport vehicle preparation areas were available. This graphic display is shown in Figure 1.

The primary value of using the above graphical display was the enhanced communication of the models capabilities and the experimental results to decision makers who had little experience with simulation. The display

*Copyright of Pritsker and Associates, West Lafayette, Indiana.

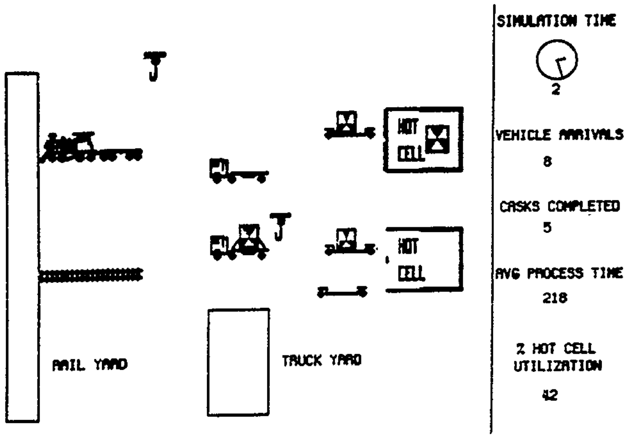

Figure 1. INSIGHT Representation of an MRS Facility

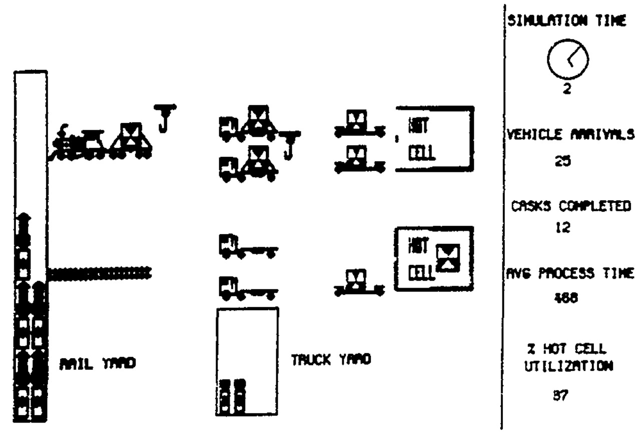

Figure 2. Representation of an MRS Facility with Increased Cask Arrival Rates

was designed to clearly describe the modelled system. To provide a simple and easy-to-understand display, INSIGHT enables the modeller to 1) quickly design icons to display the different resources of the model, 2) arrange the icons in a layout that portrays the actual system layout, and 3) utilize colors, lines, text, and even displays of key statistics on the right margin. The model can then be run to allow monitoring of the model as it progresses. Although the display is not essential for the modelers understanding of the model, researchers at BNW found the graphics to be extremely effective for explaining what the simulation actually does and for showing how different strategies such as altered arrival rates, processing times, and machine reliability can affect key performance variables.

This example display in Figure 1 begins with transport vehicles arriving to the queue for an available transport vehicle preparation area. If vehicle loading areas are empty, they are displayed as a railroad track (for the railcar loading) and an empty space for the trucks. When a railcar is positioned for unloading, a railcar icon is displayed in place of the railroad track. Truck icons are also displayed when an unloading space becomes available. These icons remain until they are reloaded with an empty cask; the screen then either reverts to represent an available loading space or portrays another vehicle to represent that a queued vehicle has been selected. When a crane is available, the shipping cask is transferred from the transport vehicle to a cask cart. Each of the two available hotcells are served by two cask carts. Workstations and equipment are shown in blue when idle and red when busy. Cask arrival count and cask completion count are updated and displayed as the simulation progresses. Statistics on the mean system throughput times and the average hotcell utilization were also selected for display on the right margin of the screen. In this example, one year of facility operation can be viewed in approximately 10 minutes, with the units of time displayed below the clock in Figure 1 representing a 24 hour day. The display shown in Figure 2 represents the facility when the arrival rates are substantially increased, showing increases in queue lenths and hotcell utilization.

The icons and the display layouts were created using the two INSIGHT graphics editors. The editors are simple to use and allow displays such as in Figures 1 and 2 to be built in about an hour. Also, icons can be saved in a library for use on other model analyses.

For example, robots, punch presses, and other standard devices need only be developed once and stored in a library for use in future models. Several systems in addition to the MRS Facility have been modeled and analyzed using INSIGHT graphics. Figure 3 displays an aircraft jet engine rotor stacking facility. A third example of INSIGHT graphics is an I-beam manufacturing facility (Figure 4). These displays were used to explain simulation model structures and capabilities as well as parametric analyses to upper management and others not familiar with simulation analysis. The next section provides a brief explanation of the capabilities of INSIGHT.

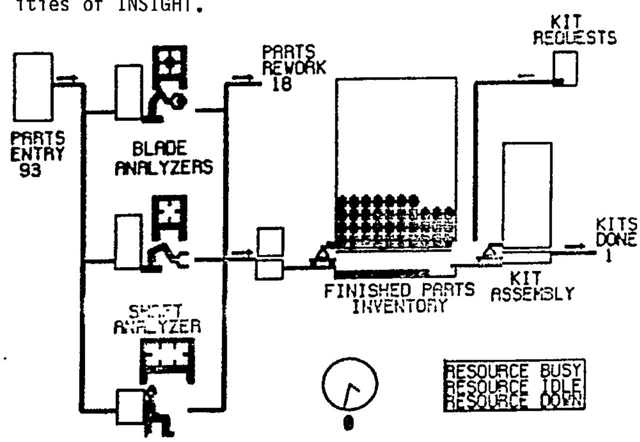

Figure 3. INSIGHT Representation of an Rotor Stacking Facility

INTRODUCTION TO INSIGHT

The primary purpose for developing INSIGHT was to simplify the development and analysis of simulation models. When the software development began in 1981, simulation model development required that the researcher be very familiar with the simulation language. Additionally, the simulation output was formatted for the

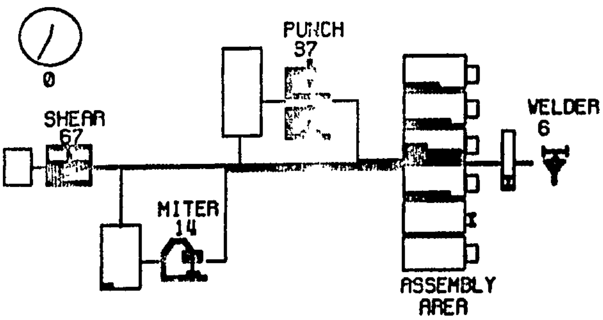

Figure 4. INSIGHT Representation of an I-BEAM Manufacturing Facility

experienced user; the results needed interpretation and reformatting before presentation to management or to others with no experience in simulation. INSIGHT, therefore, was designed to overcome these problems by simplifying the development and interpretation of simulation models. The major elements of the software, which are described below, were code development tools, a preprocessor, several graphics editors, and a postprocessor. The benefits were to be more efficient model development and more effective communication of the simulation results.

Code Development Tools

Each INSIGHT model is made up of a large number of files. Each file contains information about a specific aspect of the model. A command called "MODINIT" will build a default version of a model so that all of the parts are there and in working order. This default model is immediately executable which is a important feature. A model developer only needs to specify things he cares about. The rest of the detail is taken care of by INSIGHT.

To assist the developer, a "MODHELP" command provides on-line assistance in locating a file, specifying the format of a SLAM statement, or deciding what to do next.

The "MODLIS" command generates a listing of the model. This listing contains each of the model files with a table of contents and index for easy reference.

Preprocessor

The preprocessor takes four types of input specification and produces the elements of an executable model as shown in Figure 5. In addition to the fact that the preprocessor allows each input specification to use the same parameters, it allows for symbolic specification of simulation entities where SLAM or FORTRAN would require numeric values. For instance, description names replace file numbers for resources, queues, etc. It also supplies SLAM control commands and allows for conditional generation of commands in FORTRAN and SLAM that are parameter dependent. Finally, it allows items and routines specified in SLAM to be specified in an identical way in FORTRAN. In short, the development environment is greatly simplified and has greater integrity.

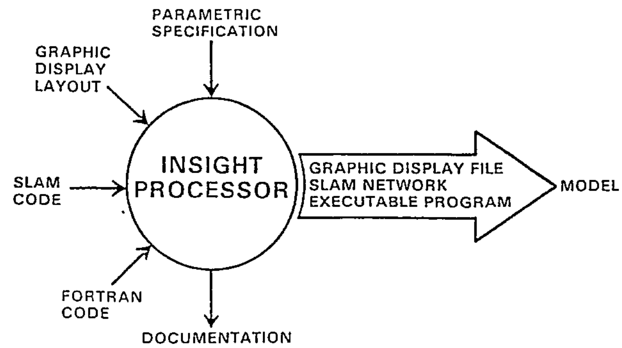

Figure 5. INSIGHT Preprocessor Operation

Graphic Editors

INSIGHT includes two different graphic editors. The first is the symbol editor which is called "GENSYM". This editor allows the user to create or modify rasterized icons (symbolic pictures). These icons are used to represent different resources in the simulation and the states of that resource. Typically, each resource has three distinct icons to represent the busy, idle, and down states. The symbol editor allows the user to draw the icons by turning on or off pixels (picture elements) an enlarged, rectangular display area. This array of pixels is saved and output as a raster image during the simulation when required. Once a symbol has been generated, it can be used in many different simulations for different purposes. We can develop a general symbol library. Figure 6 is an example of two icons used to represent a worker and a robot.

The second editor is the graphics display file editor called "GENGDF". This utility uses form entry and graphical interaction to layout a display for the simulation. The following items are provided for display:

Lines	The user can use lines to provide arrows, boxes, or other background effects. The position, length, and color can be specified.
Text	Text is used to provide labels for resources, legends, and other background information. The text size, position, angle, and color can also be specified.
Numeric Data	Any numeric data known by the simulation can be displayed. This information can be output in two sizes, any color, and any number of decimal places.
Model Time	The current model time is displayed as a clock. This is used to see how long the model has run in simulated time as well as to get a feeling for how fast time is passing as the simulation progresses.
Files	The queues or storage areas are indicated by a boxed array of icons. The icon within the array is representative of the type of items in the queue. The size of the array can be representative of the storage limitations of the queue. A file is assigned a specific color. The icons are always that color. If the amount of

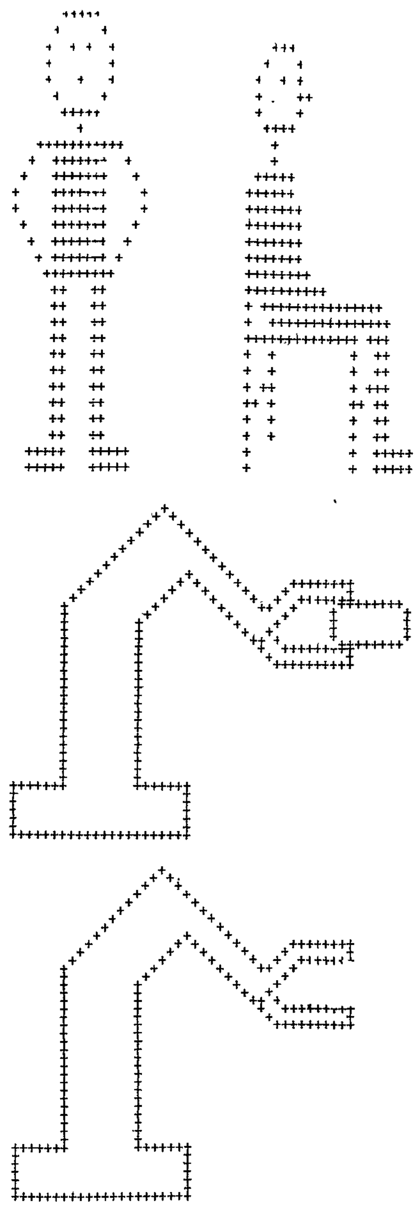

Figure 6. Example INSIGHT Icons

space allotted to that file on the display is exceeded by the number of entities actually in the file, the box around the icons is color coded to reflect the amount of overflow.

Resources — A resource is an entity which is declared to have a specific number of identical copies. Each instance of the resource is either busy, idle (ready for allocation), or down (unavailable for allocation). The resource display is a rectangular array of icons showing the state of each instance of the resource. The color as well as the shape of the icon change with each state.

Gates — A gate has an open and a closed state. It has two icons which represent each state. The icon is green for open and red for closed.

The editor allows for interactive placement of graphic items and for adjusting the visual parameters for each item. In a relatively short time, the user can develop an impressive graphic display for the simulation.

Run-Time Library

There is a set of FORTRAN routines which can be included with any simulation. These routines provide an environment in which the user can communicate with the simulation and in which the simulation can communicate with the user.

The graphics display driver is a set of routines which is always loaded with every mode. It interprets the information provided by the graphic display file and controls the output to the graphic display. The graphics are updated at specific intervals determined by the model time. The interval at which the update occurs can be controlled by the user.

The parameters of the simulation can be controlled with the operator interaction routines. These allow the operator to modify the number of resources available, cycle times, reliability information, and virtually any other parameter used by the simulation. Many of these must be tailored to the particular model, but it is simple to do. The interaction with these routines can be done by typing at a keyboard or by voice.

A voice recognition and response capability is available which allows the user to control the simulation. In addition, the user may interrogate the simulation and obtain details about the current state of the simulation. Any of the statistics available in the system can be queried for. The results are spoken by the voice output unit.

The interfaces and techniques have been developed for interfacing the simulations with scheduling algorithms and product data bases. In one demonstration system of an I-beam production facility displayed earlier we have multiple versions of an I-beam scheduler that are generating the inputs for the simulation. The simulation uses product codes and descriptions to calculate process times and determine routing. The specific details of this capability will probably change with each application, but the general capability is there.

Postprocessor

The preprocessor converts symbolic input specifications into information that slam can understand. The postprocessor takes the output generated by SLAM and adds the symbolic names in selected places. This keeps a consistent interface with the user.

SUMMARY

The interactive graphics capabilities of BNWs INSIGHT system have proven effective in better communication of simulation model capabilities and results. Even though INSIGHT is still under development and improvements are needed, the current system provides a simple yet powerful method of depicting a simulated system, especially to decision makers who are unfamiliar with simulation analysis. The graphics displays explain both the layout of individual system elements and the interaction of the same elements. In addition, the impact of key performance criteria such as process rates, machine reliability, and arrival rates upon overall system performance is easily portrayed by adding variable statistics to the display. A final benefit is the use of interactive graphics in group decision forums. Using graphics in a workshop environment to test alternate scenarios and help extract a concensus on problem solutions from design and/or management teams can greatly improve the power and usefulness of simulation analysis. There are still, however, some shortcomings of the current INSIGHT graphics system.

One shortcoming of the current INSIGHT system is its requirements for a main frame computer environment and an expensive, high-resolution graphics scope and digitizer pad. While this equipment is available at larger companies, access is often difficult. Also, most smaller firms generally will not have this equipment. Therefore interactive graphics systems will have better accessibility if they can run on smaller computers (i.e., 16 bit processors) and use less expensive graphics hardware.

A second problem with our current INSIGHT system is that the user must be reasonably proficient with computer systems. The interface to the graphics editors must be extremely user friendly. Another requirement is a basic familiarity with simulation techniques and languages.

A final problem is the possible misinterpretation of simulation analysis through the use of interactive graphics. Graphics provides an effective tool for communicating system performance and model design yet inexperienced users may infer incorrect conclusions from displayed statistics. For example, start-up phenomena and run length considerations may be ignored and the displayed results may be meaningless. We feel that the simulation analysis must be performed first; the graphics should then be used to display the validated results.

In conclusion, Battelle Northwest's interactive graphics package (INSIGHT) has proven to be a powerful means to communicate to decision makers 1) what the model can do and 2) the impacts of alternate scenarios upon system performance. More experimentation in workshop and/or expert panel environments is planned to better test the value of interactive graphics. Finally, improvement in the user interface and implementation on less expensive systems is needed to broaden the applicability of this and similar interactive simulation graphics systems.