Proceedings of the 1993 Winter Simulation Conference
G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles (eds.)

PROMODEL TUTORIAL

Charles R. Harrell
Jeffrey J. Leavy

PROMODEL Corporation
1875 South State Street
Suite 3400
Orem, Utah 84058, U.S.A.

ABSTRACT

ProModel is a powerful yet easy-to-use simulation tool for modeling all types of manufacturing systems ranging from small job shops and machining cells to large mass production and flexible manufacturing systems. ProModel is a Windows based system with an intuitive graphical interface and object-oriented modeling constructs that eliminate the need for programming. It combines the flexibility of a general purpose simulation language with the convenience of a data-driven simulator. This tutorial provides an overview of ProModel and presents its modeling and analysis capabilities.

1 OVERVIEW OF PROMODEL

ProModel is a simulation and animation tool designed to quickly yet accurately model manufacturing systems of all types. Engineers and managers find the manufacturing oriented modeling elements and rule-based decision logic extremely easy to learn and use. Users are particularly delighted when they discover that ProModel is capable of modeling their most complex systems. Because it provides such an intuitive and straightforward approach to modeling, it is also attractive to professors in engineering and business programs who are interested in teaching modeling and analysis concepts without having to teach computer programming.

While most systems can be modeled by selecting from ProModel's complete set of modeling elements (e.g. resources, downtimes, etc.) and modifying the appropriate parameters, complete programming capability is also provided if needed for modeling special situations. Built-in language features include if-then-else logic, boolean expressions, variables, attributes, arrays and even access to external spreadsheet and text files. For those who prefer coding complex logic using a programming language such as C, Pascal or Basic, external subroutines may be dynamicallly linked to the model and called from anywhere inside of the model at run time. In this way, ProModel allows systems analysts and simulation experts to use the tools they are most comfortable with to provide total flexibility.

Model development is completely graphical and object-oriented. To the extent possible, all input is provided graphically with information being grouped by object type and presented in a "spreadsheet-like" format for quick and intuitive access. For example, when you define a machine you can define its icon, capacity, downtime characteristics, input and output rules, desired output statistics, etc. ProModel complies with GUI standards which means that individuals familiar with other standard Windows programs such as word processing or spreadsheets will have no trouble learning how to use ProModel. This data input approach minimizes the learning curve for beginners and maximizes the efficiency for modifying large and complex models.

A unique feature in ProModel is the ability to bring up a pop-up menu depending on the current context that prompts the user in defining any statement or expression. This enables any expression or statement to be entered using only the mouse. It also eliminates the need to remember variable or other element names that you wish to reference by allowing you to select them from a list box.

Quick and convenient online documentation is available through ProModel's integrated Help system and online tutorial. The Help system uses the Windows Help system which allows maximum flexibility for looking up anything from command syntax to descriptions of model building modules. The ProModel Tutorial is an interactive teaching tool that provides quick lessons on how to build models, how to use the various operational features of the software, and how to run models and access output reports.

To further reduce model development time, ProModel provides model merging capabilities to allow several individuals to be working separately on different sections of a large model. Additionally, frequently defined cells or even commonly used decision logic may be stored as submodel templates which eliminate the need to "re-invent the wheel" with every model. These templates can even have specially designated parameters which may be changed by the user.

Animation development is integrated with the model definition. A major drawback of many simulation software products is that animation development is independent from simulation model development. This makes it time consuming and inconvenient for engineers to use animation as a validation\verification tool. ProModel integrates system definition and animation development into one function. While defining routing locations, conveyors, AGV paths, etc., you essentially develop the animation layout. The layout screen is a virtual screen which can be scaled to an actual factory layout.

Simulation results are informative and may be displayed in tabular or graphical form. Many simulation software products require special commands to generate statistics that are difficult to interpret for non-simulationists. ProModel allows quick and convenient selection of reports and provides automatic tabular and graphical reports on all system performance measures. Output reports from several simulation runs can even be compared on the same graph.

ProModel runs on any standard 386 computer. Most engineers, managers, and professors have easy access to IBM or compatible computers with VGA graphics capabilities. ProModel does not require any special graphics cards, special monitors, or a math coprocessor. This makes it convenient and cost effective for companies and academic institutions that have standard microcomputers. ProModel also runs on LANs (Local Area Networks) .

2 MODELING ELEMENTS

The modeling elements of ProModel provide the building blocks for representing the physical and logical components of the system being modeled. Physical elements of the system such as parts, machines, or resources may be referenced either graphically or by name. Names of modeling elements may be any word containing up to 60 alphanumeric characters. Following is a brief description of each of these elements (see Figure 1).

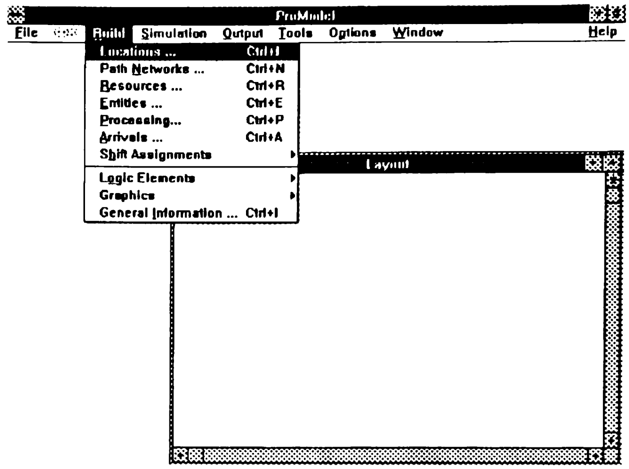

Figure 1: ProModel's Modeling Elements

2.1 Locations

Routing locations are fixed places in the system (e.g. machines, queues, storage areas, work stations, etc.) to where parts or entities are routed for processing, storage or simply to make some decision about further routing. Routing locations may be either single unit locations (e.g. a single machine) or multi-unit locations (e.g. a group of similar machines performing the same operation in parallel).

Routing locations may have a capacity greater than one and may have periodic downtimes as a function of clock time (e.g. shift changes), usage time (e.g. tool wear), usage frequency (e.g. change a dispenser after every n cycles), change of material (e.g. machine setup) or based on some user defined condition. Routing locations may be assigned input and output rules. Input rules are used for selecting what entity to process next while output rules are used for ranking entities (i.e. FIFO, LIFO, user-defined) in a multicapacity location.

Two special types of locations that provide movement as well as performing holding and operation functions are queues and conveyors. A queue mimics the behavior of waiting lines, including the movement of entities through the line. Conveyors are accumulating or non accumulating and have a particular speed and load spacing. Conveyors may be configured together to provide conveyor networks.

2.2 Path Networks

Path networks are optional and define the possible paths that entities and resources may travel when moving through the system. Path networks consist of nodes connected by path segments and are defined graphically with simple mouse clicks. Multiple path networks may be defined and one or more resources and/or entities may share the same network. Movement along a path network may be defined in terms of distance and speed or by time. Path distances are automatically computed based on the layout scale defined by the user.

2.3 Resources

A resource may be a person, tool, vehicle or other object that may be used to:

- transport material between routing locations.
- perform an operation on material at a location
- perform maintenance on a location or other resource that is down.

Resources may be either static or assigned to a path network for dynamic movement. Built-in decision rules can be used for allocation of resources and for prioritization of part pick-up and delivery. Motion characteristics of resources such as empty and full speed, acceleration, deceleration, pickup-up and delivery time can also be specified.

2.4 Entities (or parts)

Parts or entities refer to the items being processed in the system. These include raw materials, piece parts, assemblies, loads, WIP, finished products, etc. Entities of the same type or of different types may be consolidated into a single entity, separated into two or more additional entities or converted to one or more new entity types.

Entities may be assigned attributes that can be tested in making decisions or for gathering specialized statistics. The graphic of an entity can be changed as a result of an operation to show the physical change during the animation.

2.5 Processing (or routing)

This element defines the processing sequence and flow logic of entities between routing locations. The operation or service times at locations, resource requirements, processing logic, input/output relationship, routing conditions, and move times or requirements can be described using the Processing element.

Operation times can be defined by constants, distributions, functions, attributes, subroutines, etc. or an expression containing any combination of these. Operation logic can include IF THEN ELSE statements, loops, nested statement blocks and subroutine calls. Resource related statements such as GET, USE, GET JOINTLY with boolean expressions and built-in operation statements such as ACCUM, JOIN, GROUP greatly simplify otherwise complex logic in describing the processing requirements. Built-in and user-defined routing rules provide flexibility for modeling all types of routing conditions.

2.6 Arrivals (or production schedule)

Deterministic, conditional or stochastic arrivals can be modeled using this element. External files including production schedules or arrival data can be read into ProModel in the Arrivals element. Built-in or user defined distributions or spreadsheet created data can be used to define inter arrival times and quantities.

2.7 Shifts (or work schedules)

A powerful feature is the ability to define custom work and break schedules through ProModel's Shift Assignment module. Work and break schedules are defined graphically by time of day and day of the week. Resources or locations may then be assigned to use a specific shift schedule.

3 LOGIC ELEMENTS

Logic elements are used in statements and expressions to define special decision and operating logic used in a model. Some logic elements are pre-defined such as system functions that return information such as the current available capacity of a location. Fifteen different probability distributions are also provided.

There are several types of logic elements that may be defined by the user. Like model elements, names given to these logic elements may be up to 60 characters in length. Figure 2 shows the Logic Elements menu.

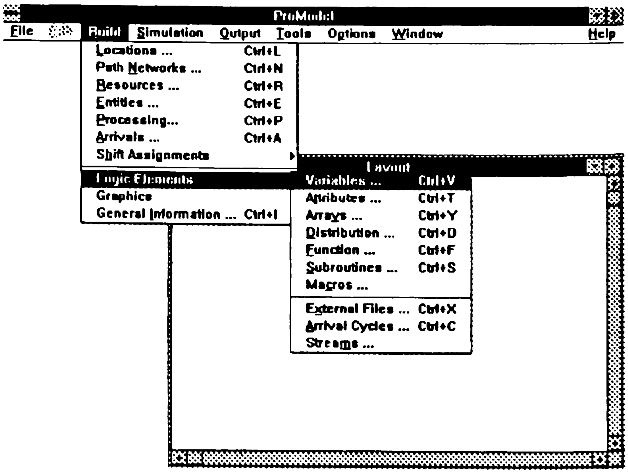

Figure 2: Logic Elements Menu

3.1 Variables

Variables are used for decision making and statistical reporting. The value of a variable may be monitored over time and displayed at the end of the simulation as a history plot or histogram. Variables may hold integer or real values. Local variables may also be used for quick convenience when defining logic.

3.2 Attributes

Attributes for entities and locations can be defined. They can take on real or integer values. Location, resource and entity names may also be assigned to attributes.

3.3 Arrays

An array is a matrix of variables representing multiple values. An array may be one dimensional or multi dimensional.

3.4 Subroutines

A subroutine is a user defined block of statements that can be passed values when called and optionally return values when completed. Subroutines may be referenced in the place of any value or logic. A complex operation performed several places in a model may be appropriately defined by a single subroutine.

3.5 Macros

A macro is a complex expression that can be defined once and used multiple times as part of a logic statement (i.e. processing, scheduling, downtime logic). Macros can be helpful in situations where the same expression is repeated in many places in the model.

3.6 Files

One of the powerful features of ProModel is the ability to read data from external text or spreadsheet files or write data to external files. For example, operation times (even in the form of expressions) from an EXCEL spreadsheet file can be read into ProModel using this element.

3.7 Tables

Tables are used to define table functions, user distributions or arrival patterns. For example, a user defined distribution which returns an operation time of 5 minutes thirty percent of the time and an operation time of 8 minutes seventy percent of the time can be expressed in the Tables element.

4 GRAPHICS

Graphics in ProModel are realistic and easy to create. Visually realistic animation helps simulation to become an effective communication vehicle between engineers and managers. ProModel comes with an extensive library of graphics with provision to create and add other graphics to the library. ProModel's graphics editor comes with a complete set of drawing tools and a full spectrum of color selections. Scaling, rotating, copying and many other editing features are available. You can even import drawing from other graphics packages. With little effort you can develop quick and simple 2D layouts, or, with little extra effort, 3D perspective layouts. CAD layout drawings (e.g. AutoCAD) can also be brought in to use as the model background. The Graphics Editor is shown in figure 3.

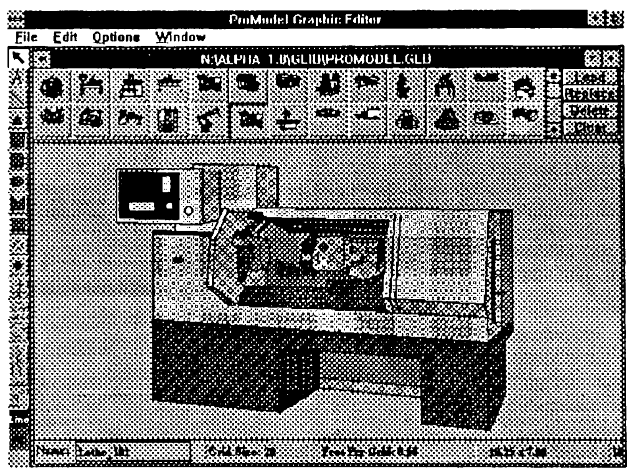

Figure 3: ProModel's Graphic Editor

5 RUNTIME INTERFACE

The runtime interface (RTI) is a convenient and controlled environment for modifying selected model parameters (capacities, operation times, etc.) without having to change the model data directly. It also provides an experimental environment which permits multiple scenarios to be defined and simulated. The RTI can be accessed at the beginning of a simulation run for making modifications for a single run or for saving alternative modifications as scenarios for doing multiple scenario analysis.

6 RUNNING THE SIMULATION & ANIMATION

Models can be run for a specified length of time or until all entities have been processed. Multiple replications may also be specified. Models may be run with or without animation. The animation is very smooth and it maintains great resolution at any zoom factor.

During the simulation, you can query the status of resources or the current value of any logic element. Source level debugging allows you to trace each action with the option to display the actual source statement that triggered the action. Trace statements may be confined to a single area of a model. Status lights for locations change colors to help you see various states such as busy, idle, down, etc.

The animation screen is a virtual screen which means the animation layout is limited only by memory in your computer. By turning off the animation, you can speed up the simulation, run for a while and turn the animation back on. The simulation clock resolution can be expressed in terms of hours, minutes, or seconds with a clock resolution of .00001 a second.

7 OUTPUT REPORTS

By choosing the statistics for resources, locations, entities, variables, etc., users can customize the output reports. The statistics are written to ASCII output files which can be exported to spreadsheets. Additionally, graphical reports of the outputs can be displayed, printed or plotted. These graphs can be individual or comparative pie charts, histograms, time-series plots, etc. See Figure 4 for a sample location state graph.

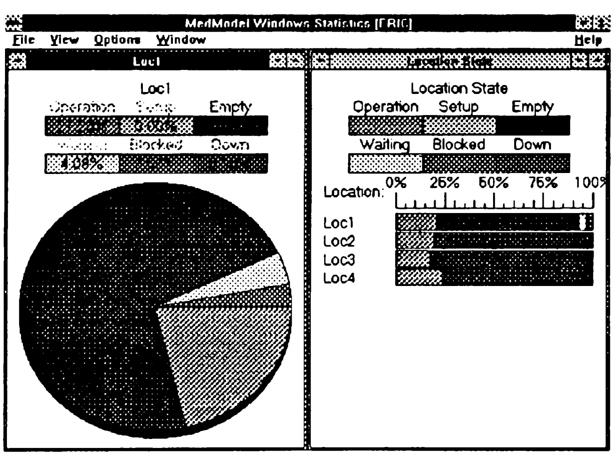

Figure 4: A Location State Graph

8 OPTIONS

One of the most accomodating features of ProModel is the provision for customizing the modeling environment to suit the preferences of the user. Under the options menu, you can zoom in or out, display a grid, define default fonts and sizes for text, set default directories for model files, icon library files, or model output files. You can set and save default settings (i.e. window sizes, appearance).

ProModel also allows you to choose either a beginner or an advanced user setting for model development. While the novice user benefits from the prompts and added explanations provided by the beginner setting, the expert user benefits from the open and direct editing capability of the advanced user setting.

9 CONCLUSIONS

Until recently, manufacturing companies have not fully benefited from simulation in making continuous improvements because of the time, programming expertise, and cost involved in getting useful results. ProModel is designed for manufacturing companies to fully achieve the benefits of simulation technology at an affordable price. ProModel is directed toward making simulation a standard tool in the hands of engineers, managers and systems analysts just as spreadsheet software is in the hands of accountants and financial analysts.

REFERENCES

Harrell, C. R.. 1989. PROMOD (PROduction MODeler) for IBM PC's. In *Supplementary Proceedings of the SCS Multiconference*, eds. S. Spencer and G.Richardson, 65-70. San Diego, California.

Harrell C. R. and K. Tumay. 1990. ProModelPC Tutorial. In *Proceedings of the Winter Simulation Conference*, eds. O. Balci, R. Sadowski and R. Nance, 128-131. New Orleans, Louisiana.

Harrell C. R. 1990. Trends in Manufacturing Simulation. In *Proceeding of Autofact Conference*, eds. A. Adlard, 21-31. Detroit, Michigan.

Harrell C. R. 1991. ProModel Tutorial. In *Proceedings of the Winter Simulation Conference*, eds. B. Nelson, D. Kelton, G. Clark, 101-105. Phoenix, Arizona.

Harrell C. R. and K. Tumay. 1992. ProModel Tutorial. In *Proceedings of the Winter Simulation Conference*, eds. J. Swain, D. Goldsman, R. Crain, J. Wilson, 405-409. Arlington, VA

AUTHOR BIOGRAPHIES

CHARLES R. HARRELL is the founder and Chairman of PROMODEL Corporation. He led the development of ProModel and has been actively involved in future product developments. Charles received his B.S. from Brigham Young University, his M.S. from the University of Utah and Ph.D. in Manufacturing Engineering from the Technical University of Denmark. He has worked in simulation and systems design for Ford Motor Company and Eaton Kenway Corporation prior to founding PROMODEL Corporation. Dr. Harrell teaches Manufacturing Engineering at BYU and is a member of IIE and SME.

JEFFREY J. LEAVY is the Director of Training Services at PROMODEL Corporation. He is in charge of the development of ProModel's training curriculum as well as their trainer certification program. He received his B.S. and Masters degrees from Brigham Young University. Prior to joining PROMODEL Corporation, Jeff worked for a Computer Integrated Manufacturing research center at BYU that utilized simulation technology. He is a member of IIE.