Proceedings of the 1986 Winter Simulation Conference
J. Wilson, J. Henriksen, S. Roberts (eds.)

CINEMA TUTORIAL

Kevin J. Healy
Vice President
Systems Modeling Corp.
248 Calder Way
State College, PA 16801

ABSTRACT

CINEMA is a general purpose, microcomputer based, animation system designed to work with the SIMAN simulation language. A sophisticated yet easy to use graphical interface allows users with little or no programming skills to build highly detailed animations of any SIMAN simulation model.

INTRODUCTION

One of simulations's shortcomings has been that outputs from a simulation model typically take the form of summary statistics or simple graphs. Although these outputs are necessary to measure and draw conclusions about performance of a system, they provide little insight into the dynamic interactions between the components of the system.

Computer animation represents an ideal solution to the problem of understanding the dynamic behavior of a simulation model. In the past few years substantial efforts have been devoted to the development of new simulation/animation systems as well as animation interfaces to existing simulation languages.

In those applications in which animation has been employed, the benefits of "seeing" the system operating have been substantial. Two main benefits which have been cited are the following:

Model Verification - An animation is an ideal method for verifying the correct operation of the model. Subtle errors which might not be apparent from standard simulation output become obvious when the operation of a system is displayed graphically.

Selling the Solution - Animation is an extremely powerful aid in quickly and thoroughly communicating the results of a simulation model to decision makers.

There exists, however, a potential to misuse animation. In addition to modeling errors, the simplifying assumptions that are made in the development of a model of a real system become readily apparent in an animation. Consequently, there is a tendency to add unnecessary detail to the model to produce a more lifelike animation. The model should contain only significant elements with each represented in a level of detail that is sufficient to meet the stated objectives of the analysis.

A more serious misuse occurs when decisions are based solely on the output of an animation. An animation is not a substitute for the controlled experimentation and analysis of outputs from a simulation model. It should be used in conjunction with other analysis aids to gain a detailed understanding of the operation of a model.

Another drawback in the past has been the substantial amount of effort and programming skills required to generate even simple animations. Ideally, the modeler should be able to quickly and easily generate both simple animations for his own use in the development and debugging of a model as well as highly detailed animations for presentation purposes.

These considerations provided the motivation for the development of CINEMA, a general-purpose, microcomputer based animation system designed to work with the SIMAN simulation language. A CINEMA animation is a dynamic display of graphical objects that change size, shape, color, or location on a static background in correspondence with changes in a SIMAN simulation model as it is executing.

The CINEMA program does not generate a simulation model. You must first build a simulation model of the system using the SIMAN language. With minor exceptions, the SIMAN model is constructed without special consideration for whether it will be run with an animation. The CINEMA program is then used to construct an animation layout which is a graphical depiction of the physical components of the system being modeled. You then execute the SIMAN simulation model in conjunction with the CINEMA layout to generate a graphical animation of the system dynamics.

The development of the animation requires no programming skills. The key to the system is the design of the *mouse/menu* user interface which is based on pioneering developments made at the Xerox Palo Alto Research Center (PARC) in the mid-seventies [1].

The mouse is a hand-held pointing device that controls the motion of a cursor on a graphics screen. By moving the mouse across a desk top, you guide the cursor to symbols on the screen that represent the commands of the CINEMA program. By pressing a button on the mouse, you can select the commands you want to perform.

The two screen symbols used to represent commands in the CINEMA program are called *headings* and *menus.* The commands are organized in a hierarchy of levels. At each level, the commands are represented by a set of headings arranged across the top of the screen. When you position the cursor on a heading and press a button on the mouse, a menu "pulls-down", revealing a list of secondary choices. A copy of a actual screen image in Figure 1 illustrates the mouse/menu concept.

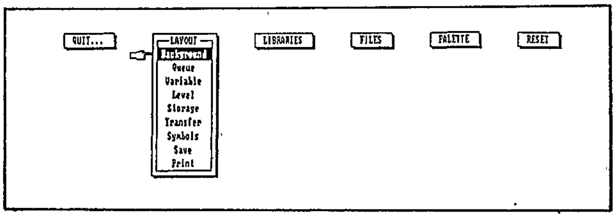

Figure 1: Pull-down Menus

HARDWARE REQUIREMENTS

The CINEMA system is designed to run on an IBM PC-AT with 640K bytes of memory, an 80287 math co-processor, and the DOS operating system. Additional graphics hardware includes a proprietary, high resolution (1024 x 768), noninterlaced color graphics display board (capable of displaying 16 colors simultaneously from a palette of 4096) and a high resolution, 19- inch diagonal color monitor.

THE ANIMATION LAYOUT

An animation layout is a combination of graphical objects that form a representation of the system being modeled. The objects that comprise a layout are one of two types referred to as *static* and *dynamic*. The static objects within a layout form a background which will not change during the animation. In a simulation of a manufacturing system, this might correspond to a sketch of the walls, aisles, posts, etc., of the facility being modeled. The dynamic objects are superimposed on the static background and change size, shape, color, or location, in correspondence with changes in the state of the system during the execution of a simulation. Workpieces, workers, machines, robots, etc. would be represented as dynamic objects within a layout. An example of an animation layout is pictured in Figure 2.

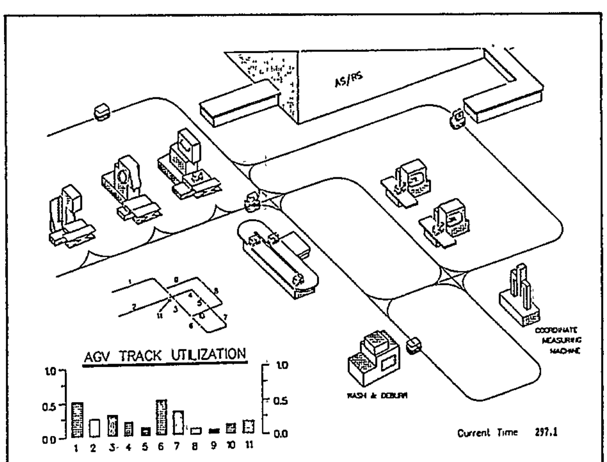

Figure 2: Animation Layout

The Static Component

The static component of the layout is constructed using a set of elementary computer-aided drawing functions such as *line*, *box*, *circle*, and *arc*. You select from headings and menus to choose a drawing function as well as set attributes of the function including color, line width, and line style. The basic functions are drawn in *rubberband* mode, which allows you to view the size and orientation of the object before actually adding it to the layout. For example, to draw a box, you position the mouse cursor on a screen location that represents one corner of the box and press a button on the mouse. You then see the box expand and contract as you move the cursor to locate the opposite corner of the box. After positioning the opposite corner, you press the mouse button a second time to add the box to the layout.

Other features include a *sketch* function which allows you to draw any free-form curve by simply moving the graphics cursor along the desired path while holding down a button on the mouse. A *fill* function allows you to color-in any enclosed region with either solids or patterns. *Text* can be added to the static background in a choice of font, size, color, and orientation that are selected using the pull-down menus. Editing features for manipulating images on the screen include an *erase* function which erases all graphics and text in the path of the cursor as long as a button on the mouse is held down. Portions of the screen can be moved or copied to other areas with the *move* and *copy* functions. Examples of the various drawing functions are illustrated in Figure 3.

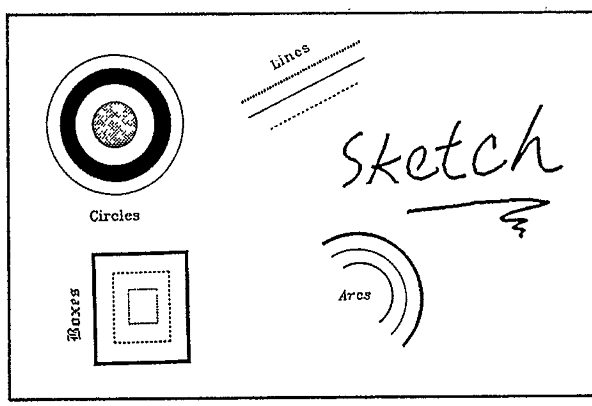

Figure 3: Examples of Static Background Drawing Functions

The Dynamic Component

While a SIMAN simulation model is executing, it is continually updating an internal representation of the state of the system. The state of the system is defined by the current values of status variables such as; the number and location of entities within the system, the values assigned to attributes of the entities, the status of the resources within the system, and the current simulated clock time.

Each dynamic object in an animation layout is associated with a specific element of the system state as represented by the SIMAN model. For example, one object might be associated with the value of a variable such as the simulated clock time while another might be associated with the status of a SIMAN resource. Dynamic objects are automatically updated as the state of the system changes during the simulation.

Following is a discussion of the dynamic objects which can be incorporated into a CINEMA layout and the association between these objects and specific modeling constructs within the SIMAN language.

Entity Symbols

A SIMAN process model is a step-by-step description of the processes (creation, queueing, processing delays, etc.) an entity encounters as it flows through a system. In an animation, an entity is represented as a moving and/or changing symbol. The symbol could be a sketch of a workpiece or a partially assembled car. As the entity moves from workstation to workstation within the simulation model, its corresponding symbol is automatically moved across the static background.

Entity symbols are created by coloring boxes on a screen sized grid. Each box in the grid corresponds to one picture element (pixel) of the actual symbol image. As the symbol is created on the grid, it is displayed in actual size in the upper left corner of the screen (see Figure 4).

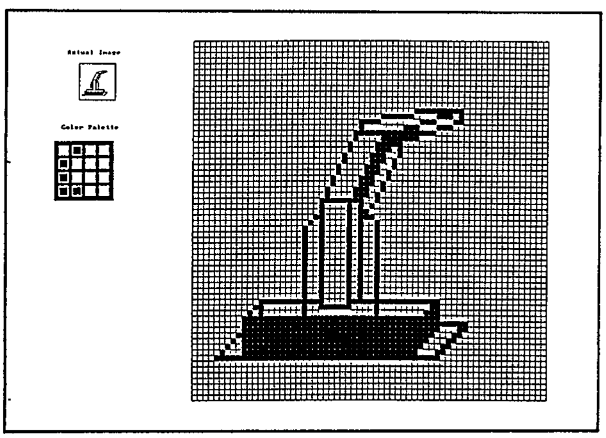

Figure 4: Symbol Drawing Grid

The symbols are stored in an *entity symbol library.* Symbol libraries are maintained separately from the animation layout and may be saved (stored on disk) and recalled for later use.

To establish the association between an entity symbol and a specific entity within the SIMAN model, you must reserve one of the general purpose entity attributes in the SIMAN model. This attribute is used to store a number that, in CINEMA, is associated with a specific symbol from an entity library. When the designated entity attribute is assigned a value in the model, the corresponding entity symbol is displayed in the animation.
To change the symbol you need only assign a new value to the entity attribute. Consider for example, a simulation of an automotive assembly plant. The designated attribute of an entity arriving to an assembly station might be set to a value that corresponds to a symbol of a car body without doors. After leaving the workstation, the associated symbol could be changed to a car body with doors simply by assigning a new value to the designated entity attribute.

Resource Symbols

Resources are used in SIMAN to model limited items in a system, such as machines and workers. Entities compete for the limited number of units of a resource and incur queueing delays when enough units are not available. In a SIMAN model, a resource assumes one of four possible states: busy, idle, inactive, or preempted. The resource is in a busy state when it has been allocated to an entity at a SEIZE block. The resource remains busy until it is released by the entity at a RELEASE block, which changes its status to idle. Units of a resource can be removed from use with an ALTER block. This changes the status of those units of the resource to inactive until they are placed back in service using the ALTER block again. When a resource is allocated to an entity at a PREEMPT block its state is preempted until it is released by the entity at a RELEASE block.

Resource symbols are used to display changes in the status of a resource. A resource is represented by four distinct symbols called the *idle*, *busy*, *inactive*, and *preempted symbols.* Like entity symbols, resource symbols are created by coloring boxes on a screen sized grid and stored in a *resource symbol library.* A resource is added to the layout by selecting an idle symbol from the library and positioning it on the static background using the mouse. At that time you must also associate a SIMAN resource number with that symbol. When the status of a resource changes in the simulation model, the associated resource symbol (busy, idle, inactive, or preempted) is displayed at the proper location on the screen to reflect the status change.

Transporter Symbols

In SIMAN, the term transporter denotes a general glass of movable devices that may be allocated to entities. Examples of devices that might be modeled as transporters include lift trucks, automated guided vehicles and cranes.

In a SIMAN model, a transporter assumes one of three possible states: busy, idle, or inactive. The transporter is in a busy state when it has been allocated to an entity at a REQUEST or ALLOCATE block. The transporter remains busy until it is freed by the entity at a FREE block, which changes its status to idle. A transporter can be removed from use with a HALT block. This changes the status of transporter to inactive until it is placed back in service using the ACTIVATE block.

Transporter symbols are used to display changes in the status of a transporter. A transporter is represented by three distinct symbols called the *idle*, *busy*, and *inactive symbols.* Like entity and resource symbols, transporter symbols are created by coloring boxes on a screen sized grid and stored in a *transporter symbol library.*

To add a transporter to the layout, you associate a SIMAN transporter number with an idle symbol from the transporter symbol library. When the status of a transporter changes in the simulation model, the associated transporter symbol (busy, idle, or inactive) is displayed to reflect the status change.

Transfers

Distinct workstations within a system are modeled in SIMAN as station submodels. The movement of entities and material handling devices between station submodels is accomplished through the use of three transfer blocks: ROUTE, TRANSPORT, and CONVEY. Entities sent from these blocks later arrive at STATION blocks within the SIMAN model. In CINEMA, the tasks involved in defining the paths on the screen that entities follow when they are transfered between stations are grouped into the TRANSFER section.

The first step is to position *station symbols* on the layout. These symbols represent the beginning and end of any route, transporter, or conveyor path. Each station symbol is numbered to match the corresponding station number in the SIMAN model. A route, transporter, or conveyor path is then defined as a series of connected line segments between two station symbols. When an entity is transfered between two stations in the model, its entity symbol is periodically displayed along the corresponding path on the screen. When an entity uses a transporter to move between stations, both the entity symbol and the busy transporter symbol are shown moving along the path.

Queues

Queues are the result of delays that entities incur while waiting for a prescribed change in the state of the system to occur. For example, entities must wait in line when a resource, transporter, or space on a conveyor is needed but is unavailable. In SIMAN, queues are modeled by the QUEUE block. One of the operands of the QUEUE block is the number of an internal file in which the entities reside while queued.

In CINEMA, a *queue symbol* is used to display the entities that reside in a file associated with a SIMAN QUEUE block. The queue symbol is a line segment of any length and orientation that can be added at any location on the layout. The first point you define on the line segment represents the head of the queue and the second point represents the tail. You then associate the queue symbol with a specific file number in the corresponding SIMAN model.

When an entity enters a queue in the simulation model, the entity's symbol is displayed along the corresponding queue line at the proper location relative to the other members of the queue. When an entity exits the queue in the model, its associated symbol is removed and all following symbols are moved forward one position. When the queue in the model contains more entities than can be displayed along the length of the queue line, subsequent arrivals to the queue are not displayed. These entities eventually become visible as they move forward into the display portion of the queue when preceeding entities exit.

Status Variables

The dynamic features described so far are tied to specific modeling contructs in the SIMAN language such as; entities, resources, queues, transporters, and conveyors. CINEMA provides four additional dynamic features that can be used to represent the value of any status variable maintained by SIMAN during the execution of a simulation.

A digital display of the value of a status variable can be added to a layout using a feature called *dynamic variables*. You select the variable and define the display format (number of significant figures and places to the right of the decimal), size, and color using pull-down menus. You then position the variable anywhere on the screen using the mouse. When the variable changes values during a simulation run, the value will be output at that location on the screen using the defined size, color, and format.

A collection of three different analog indicators, called *levels*, can also be also be used to display the value of any status variable. Two of the indicators, one shaped like a box and the other a circle, fill and empty in response to changes in the value of the associated status variable. The third, called a dial, is a circular level with a sweep hand that rotates either clockwise or counter-clockwise. The shape, fill and empty colors, fill direction, size, and location of a level are specified using pull-down menus and the mouse.

With a feature called *global symbols* you can associate specific values of a status variable with different user-drawn symbols. When the variable changes values during a simulation run, the corresponding symbol is displayed on the screen. Like entity, resource, and transporter symbols, global symbols are created by coloring boxes on a screen sized grid and stored in a *global symbol library.* You then copy symbols from the library onto the layout and associate them with specific values of a status variable.

The fourth feature, called *dynamic palette*, can be used to associate the color of an object with specific values of a status variable. As the status variable changes values during the simulation run, the object changes color on the screen.

Each of the sixteen colors in CINEMA are formed by combining different amounts of three primary colors; red, green, and blue. The amounts of each to combine are determined by numeric red, green, and blue intensity values that range from 0 to 15. To incorporate a dynamic color into the layout, you first select one of the 16 color indices and associate that index with a status variable. You then define up to ten colors for that index (in the form of different red, green, blue intensities) and associate each new color with a value of the status variable. As the status variable changes values during the simulation run, any object drawn in that color index will change color.

For example, in the simulation of a steel making facility, you could define several different colors, ranging from gray to red, that are tied to the value of a variable that represents the temperature of a furnace. As the temperature of the furnace increases, you would see the furnace gradually change from gray to red.

RUNTIME FEATURES

The CINEMA system includes a special version of the SIMAN simulation language. This special version incorporates the additional code required to dynamically update the graphics screen when a simulation is executed with an animation layout. A mouse/menu interface has also been added to facilitate the management of a simulation/animation session. Other features in this special version as well as features in the general release version of SIMAN allow you to interact with a simulation model while it is executing.

You control the speed of an animation by specifying values for two parameters that are used to scale simulated time to real screen update time. A skip ahead option allows you to turn the animation off for a specified period of simulated time so that the model executes without the imposed time delays. While the simulation is executing, you can zoom-in by pressing the '+' key and pan to different areas of the screen using the arrow keys (up, down, left, and right).

You can also interrupt the execution of a simulation and recall a new layout file that might represent a different section or view of the system that was modeled. There is no limit to the number of layouts that can be built to correspond to a single simulation model. In addition, a snapshot of the system status and graphics screen can be saved (stored on disk) at any time. Recalling a snapshot immediately restores the state of the system and graphics screen to the point in time when the snapshot was saved.

With the SIMAN interactive debugging facility you can interrupt the simulation randomly or at predefined times or break points in the model to examine or modify the values of system status variables. When used with animation, it also becomes a powerful tool for determining the short term effects of changes in operating policies.

SUMMARY

The development of the CINEMA system and recent enhancements represent a significant contribution toward making animation a practical part of every simulation analysis. With minimal instruction and practice, a user with little or no programming skills can easily construct and execute a highly detailed animation. The mouse/menu interface between CINEMA and the user makes this task simple and efficient.

REFERENCES

1. Lampson, B.W., "Bravo Manual" in Alto Users Handbook, Xerox Palo Alto Research Center, Nov. 1978.

AUTHOR'S BIOGRAPHY

Kevin J. Healy is Vice President in Charge of Software Development at Systems Modeling Corporation. He holds a Bachelor of Science in Industrial Engineering and a Master of Science in Industrial Engineering and Operations Research from the Pennsylvania State University.

Systems Modeling Corp.
248 Calder Way·
State College, PA 16801
(814) 238-5919