

A TUTORIAL ON SEE WHY AND WITNESS

Andrew R. Gilman
Renee M. Watremez
Istel, Incorporated
83 Cambridge Street, Burlington, MA 01803

ABSTRACT

SEE WHY and WITNESS are companion products for visual interactive simulation. SEE WHY is a general purpose discrete event simulation system that offers special integrated features for graphical display and run time interaction. It also provides user programmable extensions to both the display and interaction capabilities. These facilities are highlighted in the discussion of SEE WHY.

WITNESS has been developed from SEE WHY. It is a manufacturing simulation program that fully exploits the visual and interactive capabilities of SEE WHY in bringing manufacturing modeling to the non-specialist. The phases of WITNESS modeling are covered in some detail as well as the characteristics of the user interface. The paper concludes with an overview of the ability to extend the WITNESS program with an interface to SEE WHY.

1. OVERVIEW OF SEE WHY AND WITNESS

SEE WHY was a pioneering development in simulation software when it was commercially introduced in 1980. It was designed from the beginning to help overcome the problem of communication between modelers and decision makers. It was therefore designed to be a visual and interactive simulation system.

Experience has shown that the original design objectives have been satisfied. Visual interactive SEE WHY models invite participation from decision makers in all phases of a simulation study. As a result, the recommendations of the study are used more confidently and more correctly in the decisions taken concerning the system under study.

As a general purpose simulation system, SEE WHY has been used in the improvement of many kinds of systems: manufacturing, retail, distribution, packaging, air and ground traffic, etc. Its flexibility makes it possible to model the control logic of virtually any type of discrete system.

It has been recognized within Istel, as well as the simulation community as a whole, that manufacturing systems provide one of the richest areas of simulation application. It is also recognized that many manufacturing models contain similar types of model constructs and model logic. Hence there is duplication of effort among simulation specialists who work in the area of manufacturing. It is almost certain that there are more manufacturing problems worthy of simulation study than there are specialists to model them. All of these factors led to the development of WITNESS.

WITNESS is a menu driven simulation program created for the study of manufacturing systems. WITNESS has been built with SEE WHY and therefore carries forward the powerful features for real-time animated color graphics and run time interactions found in SEE WHY. The modeling elements of WITNESS are higher level constructs such as Parts, Machines, Buffers, Conveyors, and Labor. While the terminology is manufacturing based, the simulation community will recognize that these same elements can be used to model in a non-manufacturing setting.

Where WITNESS cannot satisfactorily deal with some intricacies of a particular problem, a SEE WHY submodel can be merged into the overall simulation model. This capability yields an unprecedented combination of ease of use and modeling flexibility.

In the tutorial that follows we will look first at the general purpose facilities for visual interactive simulation found in SEE WHY. We will then cover in more detail the modeling capabilities of WITNESS and the characteristics of the user interface. Finally we will discuss the facilities for extending WITNESS with SEE WHY.

2. MODELING IN SEE WHY

SEE WHY models are written using the event scheduling orientation. The modeler codes a series of (usually) Fortran subprograms that call upon the SEE WHY simulation library for virtually all simulation related tasks. This same library provides the means for color display and animation as well as interactive facilities.

The SEE WHY library supports all of the standard requirements for discrete event simulation. SEE WHY routines handle the (real) simulation clock, time updating, event handling, events list (known in SEE WHY as the TIMESSET) maintenance, event scheduling and event initiation. Handled simultaneously are the tasks required for time dependent statistics gathering and simulation run control. Within this structure is the additional provision for interruption of the simulation run for interaction by the user.

SEE WHY supplies pseudo-random number generation and a series of standard theoretical probability distributions with corresponding sampling routines. This capability can be extended with the user defined distributions element described below.

The majority of the SEE WHY library is devoted to providing routines for the definition and manipulation of the SEE WHY modeling elements. The

A Tutorial on SEE WHY and WITNESS

routines available for manipulating entities, attributes, and sets, as well as inquiring on their current states, are similar to those found in other implementations of discrete event simulation. It is not the purpose of this tutorial to detail these functions. Rather, we are concentrating on the special characteristics of the SEE WHY modeling elements and the SEE WHY system as a whole.

SEE WHY maintains all of the simulation data that represents the state of the simulation (plus some information concerning the display) within its own flexible internal data storage. This flexible storage scheme allows any combination of modeling elements within the available space. As an example, any entity can have any number of attributes.

Specifically, SEE WHY provides the following elements for modeling, statistics collection, and user defined probability distributions.

ENTITIES
SETS
VESSELS
TIMESERIES
HISTOGRAMS
DISTRIBUTIONS

A key characteristic of these modeling elements is that they are given implicit display related parameters by the modeler when they are initially defined within the model (with the exception of distributions which are not displayed). In other words, the modeler gives the element display characteristics such as screen location, color, height, and spacing when it is defined. From that point on the modeler concerns himself only with the modeling logic. The changes in system state, for example an entity moving from one set to another, is automatically reflected on the screen by the system. This includes the updating of timeseries and histogram displays as each new simulation value is observed.

Another important characteristic of these elements is that the system provides standard interactions that allow the user to inspect and change almost any parameter of the defined modeling elements by stopping the simulation at any time during the run. Examples of this capability are:

- *inspecting or changing any attribute of any entity
- *finding and moving entities among sets
- *changing screen location of any element
- *changing input rates, capacity, display size of any vessel
- *erasing, displaying, rescaling any timeseries or histogram
- *respecifying user defined distributions

When the user has interrupted the model he may also, through standard system interactions, change the run mode and perform a variety of utility tasks concerning the simulation run. For example, the user can:

- *run the model in batch mode or step mode

- *save the complete state of the simulation
- *restore a previously saved state
- *restart the simulation
- *reset the simulation time
- *reschedule or deschedule events
- *restart the random number streams
- *redirect printed output to file or printer

Last but not least among the system interactions is the facility for coding interactions specific to the model. These are referred to as "OWN" interactions. When the model has been interrupted by the user the keyword "OWN" causes a subprogram, written by the modeler, to be invoked. In this subprogram the modeler can employ prompting routines from the SEE WHY library to query the user for the type of changes to be made. The modeler can then alter model parameters, model state, or even model logic using any of the routines of the SEE WHY library.

To augment the implicit display characteristics of the modeling elements, SEE WHY also provides routines for generation of static color displays. Lines, text, rectangles and areas of color can provide enhancements to element displays in order to make the screen display recognizable as the corresponding physical problem area.

For both static displays and element displays SEE WHY utilizes a powerful technique to control which part of the model is currently displayed on the screen. Any display information written to the screen, either by the system or directly by the modeler, is written using one of the one hundred "pens" available in SEE WHY. It is the pen that has a particular combination of foreground and background colors. The pens are also either active or inactive. When inactive pens are used to update the display no change is seen. Pens can be grouped together into "pictures". By activating one picture and deactivating another, different portions of the model display can be viewed selectively. The pens and pictures yield a wide variety of methods for displaying a model that requires more than one screenful of information.

The flexibility for modeling, display and interaction found in SEE WHY has made possible the construction of extremely user-friendly simulation models. SEE WHY models can truly be run, experimented with, and understood by decision makers. These same features have allowed the development of WITNESS, which fully exploits the features of SEE WHY to bring manufacturing modeling to the non-specialist.

3. MODELING IN WITNESS

WITNESS models are described in a manner very different from the general purpose orientation of SEE WHY. WITNESS is a visual interactive simulation program. WITNESS views the world as separate components or elements, that interact in specified ways. The basic elements of WITNESS version 1 are:

PARTS
MACHINES
CONVEYORS
BUFFERS
LABOR
VARIABLES
ATTRIBUTES
DISTRIBUTIONS

This group of elements will be extended in WITNESS version 2 to provide additional material handling constructs, including vehicles.

There are three main phases involved in building a WITNESS model: Define, Display and Detail. Each of the phases will be described below.

All of the commands for constructing and running WITNESS models are provided by menus and prompts within the single environment of the WITNESS program. There is no code to be compiled and linked.

There are virtually no restrictions on the order in which the phases are employed to create and run WITNESS models. At any time the user is free to move between Define, Display, Detail and the execution phases of WITNESS modeling. Elements can be added, parameters or display characteristics changed, at any time, even while the model is running. The user need not leave WITNESS to change models or restore saved model states. No run termination time need be specified, the model may be halted, saved, restored or restarted at will.

Unlike the old technology of creating a model and then running it, the recommended approach of WITNESS is to build up the model cumulatively. The Define, Display and Detail modes may be used for each component or group of components. This is in contrast to the traditional practice of using the Define mode for what is believed to be the whole of the model followed by the Display mode and then finally the Detail mode. The WITNESS user does not have to wait for the model to compile and link before running - it runs immediately. WITNESS is truly interactive modeling.

The primary WITNESS menu provides the following options:

- DEFINE - The elements of a WITNESS model may be defined.
- DISPLAY - WITNESS elements may be positioned on the screen.
- DETAIL - The timings of the model and the flow of Parts may be established.
- INITIALIZE - Certain starting conditions may be provided.
- GO - Run the WITNESS model.
- KEEP - Save the parameters of the WITNESS model.
- LIST - List on the screen the elements of the WITNESS model.
- QUIT - End the WITNESS session.

After the model has been run (i.e. GO has been selected) the user will see these same options, along with others, appearing in a blue box in the top right hand corner of the screen. This is called the interact box and will be familiar to SEE WHY users as an area of the screen reserved for interaction. All interaction after the primary WITNESS menu is controlled through the interact box.

3.1 WITNESS ELEMENTS

A WITNESS simulation model is comprised of physical elements and control elements. The Define phase is used to create and name each new physical and control element. Machines, Parts, Conveyors, Buffers and Labor, WITNESS physical elements, are used to represent the manufacturing plant. The control elements, which are comprised of Attributes, Variables and Distributions are used to control and monitor the plant.

3.1.1 WITNESS Physical Elements

PARTS are the individual items which are processed by the rest of the system being modeled. They move into, out of, and between other elements. They may combine together in an operation (as in an assembly Machine) or split apart (as output from a production Machine). Parts are referred to by name. Effectively, an unlimited number of each Part name may be introduced to the system. Routings through the other modeling elements can be based on the Part name. Attributes may be associated with Parts to provide further differentiation.

MACHINES represent the physical facilities that work on a Part changing it in some way. Machine types available are single, batch, assembly and production. Machines go through a series of possible states in an attempt to process Parts. The duration of these reflect the times specified by the user in the Detail phase. Their duration also reflects times imposed because of dependencies on the state of other elements in the model. For a Machine the user may specify times and requirements for Part processing, setup, and breakdowns. As the model runs the Machine status will change appropriately among the states of idle, busy, down, blocked, setting up, and waiting for Labor.

CONVEYORS are transportation devices where all Parts being carried move in unison. Typically Conveyors are used to represent real world conveyors. WITNESS provides two types, Fixed and Queueing. Fixed Conveyors stop when a Part reaches the end but cannot move (or be moved) off. Queueing Conveyors allow Parts to accumulate in a line from the front of the Conveyor until the Conveyor is full. Conveyors generally connect two or more elements (Machines, Buffers, or other Conveyors) when transportation over a physical distance imposes a significant delay in the availability of Parts between the two connected elements.

BUFFERS represent storage areas for Parts between processes, in particular, Machines and Conveyors. Buffers are completely passive. Parts that enter or leave a Buffer, and how long they stay in the Buffer, depends completely on the actions of other simulation Elements that precede or follow the Buffer. Buffers may have a specific internal organization that specifies the position at which Parts may enter or from which they may leave.

A Tutorial on SEE WHY and WITNESS

WITNESS also provides for dedicated Input and Output Buffers that are tied to a particular Machine. These are specified when defining Machines and are separate from the Buffer element.

LABOR represents resources which provide a service to Machines in order that a particular task (or tasks) may be carried out. Machines may require Labor for set up, production cycle, or repair. Each type of Labor in the model can be given a name and quantity of operatives. In describing the detail of a Machine, specific Labor types can be chosen individually for set up, production, and repair. In a more general sense Labor can be thought of as a movable but limited resource. It might therefore be used to represent an expensive jig or fixture that is shared by several Machines.

3.1.2 WITNESS Control Elements

ATTRIBUTES are associated with Parts. They can be used to distinguish Parts from one another. An Attribute, referred to by name, is numeric, either integer or real. Part numbers, model numbers and process times are good examples of Attributes.

Parts may be associated with common groups of Attributes when the group is defined by a reference number assigned to Attributes.

Attributes are not part of the process flow. However, they can be used as part of the flow logic.

VARIABLES are storage units for numeric values. They may be used anywhere in a WITNESS model for the purpose of process times, interarrival times, mean time between Machine breakdowns, production quantities, etc. A Variable plays no part in the process although it may be used as part of the logic, recording or display.

DISTRIBUTIONS -WITNESS offers you the use of 10 standard distributions such as the uniform, exponential, and poisson distributions. In addition to these statistical distributions the user may define "customized" Distributions. These Distributions are created from actual data entered into the system as values and weights. WITNESS determines the probabilities, creates a probability density diagram and takes random samples from this Distribution as specified by the modeler.

3.2 DISPLAY PHASE

Six of the eight WITNESS elements can be displayed in a WITNESS model. Machines, Conveyors, Buffers, Labor, Variables and Parts are positioned using a mouse or the keyboard cursor control keys.

WITNESS users can position the elements on any one of four windows. The windows are analagous to the picture facility of SEE WHY. Only one window can be viewed at any one time. This is a particularly useful feature in modeling a very large system. The modeler can dissect the system and display the Parts separately on the four windows.

WITNESS allows the simulator to have complete control over the colors, size, position and shape of the model display. WITNESS has a library of 20 shapes (or icons) with which you can represent the physical shape of a Machine. This library can be extended through SEE WHY. There is also the option for these icons to change color depending on the status of the Machine.

3.3 DETAIL PHASE

The Detail phase is the facet of WITNESS modeling that allows the modeler to identify times in the model, flow of Parts, Labor requirements, breakdowns and setups. This information is specified in a Detail Form.

Each element, with the exception of Attributes, has a specific Detail Form. The modeler simply fills in the blanks to establish the model's parameters. These details such as Conveyor length, Machine type, process times, and lot sizes, may be changed at any time while creating and running the model.

Machine cycle time, Conveyor speed, Part interarrival time, setup time, mean time between Machine failure, repair time, etc. constitute the timings of the model. Essentially, any numeric response may be detailed as a constant, Variable, Attribute, sample from a Distribution, algebraic expression, dimensioned Variable or dimensioned Attribute. When Variables and Attributes are used for times, the manipulation of these values may be performed in the Actions feature. Actions are described below.

The flow of Parts through the system is controlled through input and output rules. Machines, Conveyors and Buffers all have input and output rules. The modeler specifies the input and output for the elements through the Detail Forms.

The basic function of the input and output rules is to push and pull Parts through the Machines, Buffers and Conveyors in the system. An output rule of push will cause a Part to be pushed to a Machine, Buffer or Conveyor. Should the element to which the Part is being pushed not be prepared to accept the Part then the element pushing will assume a blocked state. Pull will cause a Machine or a Conveyor to only receive input when it needs input and not when the previous operation is ready to provide it. In a WITNESS model either push or pull systems can be modeled or a combination of push and pull can be modeled. There are seven input rules and eight output rules. This extensive variety gives the WITNESS user the ability to model complex routing. A description of each of the rules follows.

The seven basic input rules are called PULL, MOST, LEAST, %, SEQUENCE, SELECT and WAIT.

PULL: This rule allows a Machine or Conveyor to pull a Part from specified Conveyors or Buffers. Several alternatives can be specified.

MOST: This rule is used when you wish to pull a Part from one of several elements. The Buffer or Conveyor from which the Part will be pulled has most Parts in it, or has the most free capacity.

LEAST: This is similar to the MOST rule except the element chosen has the least Parts or the least free capacity.

%: This rule allows you to pull Parts in a probabilistic manner, according to percentages specified by the modeler. ie: 40% from element A, 30% from element B, 30% from element C.

SEQUENCE: This rule is used when you wish to sequence the input from a series of elements. Eg. if the first time you need a Part it comes from element A, the second time it comes from element B and the third time it comes from element C, then you would use a sequence rule. When the sequence cannot be maintained, there are three alternatives: WAIT, NEXT AND RESET. These allow you to control the sequence in different ways.

SELECT: This rule allows you to pull a Part from an element, that is selected from a list of elements, based on the evaluation of an expression.

The selection mechanism works in a manner similar to a Fortran computed GOTO. The expression may involve Variables, Attributes, sampling functions, and status functions.

WAIT: Do nothing.

There are eight output rules. They are: Push, Most, Least, %, Sequence, Part, Select and Wait. The output rules differ in two ways from the input rules. The main difference being that rather than an element pulling a Part for input, the element will push the Part somewhere for output. The second difference is the addition of the rule called "PART". This output rule allows the modeler to specify different output paths for different Parts.

Much of WITNESS' power is derived from selectively using the input and output rules in conjunction with the Actions feature. Special actions can be programmed to take place at entry to or departure from some component of the system. The Actions are a series of steps. These steps are computations which can draw on Attributes, Variables, dimensioned Attributes, dimensioned Variables and various system functions. Creation or modification of the Actions is carried out in the Detail mode using a simple editor.

3.4 FURTHER FEATURES OF THE DETAIL PHASE

3.4.1 SYSTEM FUNCTIONS

WITNESS offers several system functions that enable complex modeling logic. NPARTS will return the number of Parts in the Machine Buffer or

Conveyor specified. NFREE returns the free capacity of a specified Machine, Buffer or Conveyor. ISTATE will return an integer value associated with the particular status (idle, busy, blocked, setup, down) of a Machine, Buffer or Conveyor.

These functions can be used wherever numeric expressions are required in the Detail Forms. Or, the values returned by these functions can be assigned to Variables and Attributes in Actions.

3.4.2 SYSTEM VARIABLES AND ATTRIBUTES

WITNESS also provides system Variables and system Attributes. The system Attributes will change Part descriptions and Part colors. TIME is a real Variable holding the current simulation clock time. M and N are also system Variables. These hold sequence and index numbers.

3.4.3 REPORTS

A simple "Yes" entered next to "REPORTING" on the Detail Form will engage statistics collection for Machines, Parts, Buffers, Conveyors, and Labor. The modeler can specify a particular warm-up period. During the time specified the modeled system will be allowed to achieve steady state and then the statistics will be set back to zero. However, at anytime, the statistics may be manually set to zero.

The reports provide important information including: average size of a Buffer, average time spent by a Part in a Buffer, percent of time a Conveyor was blocked, average work in process over time, average time for a Part to move through the system, percent of time a Machine was blocked, percent of time a Machine spent waiting for an operator, average number of jobs waiting for an operator, number of jobs started, and idle time spent by Labor.

3.5 LIBRARY

A WITNESS model may be stored on the hard disk as a file which may be printed out or which may be re-used in another model. This file is known as a library file. It can be altered outside of WITNESS using a standard text editor. By editing library files segments of models can be created, stored and pieced together to create new models. These files can provide documentation for models, in hard copy form, for later reference.

3.6 LINKS TO SEE WHY

Where WITNESS is unable to deal with particular modeling intricacy, a sub-model can be created in SEE WHY and linked into the WITNESS model. Internally WITNESS is a generic SEE WHY model. WITNESS Parts and Labor units are SEE WHY entities. WITNESS Machines, Buffers, Conveyors and operator job and idle queues are SEE WHY sets.

The WITNESS/SEE WHY interface allows Parts to be moved to and from a user written SEE WHY submodel. In the SEE WHY submodel SEE WHY elements may be defined, events may be written, and system state information may be passed to and from WITNESS. Data files can be read into WITNESS Variables with the complete facilities of Fortran I/O. Additions to the display can be made and special interactions may be coded just as in SEE WHY. The link to SEE WHY is

A Tutorial on SEE WHY and WITNESS

also useful for performing calculations and decision rules that are beyond the scope of WITNESS. The values can then be passed to a WITNESS model from a SEE WHY subroutine.

3.7 CONCLUSION

WITNESS, developed for modeling manufacturing systems, enables people with no simulation experience to develop models easily and quickly.

Little training is required for building simulation models with WITNESS. The system prompts users for the model parameters through menus and questions. There is no compiling and linking required. The model is simply written in three phases and executed. Changes to the model may be made in any of the three phases at any time.

Complex routing and logic is achieved with numerous input and output rules as well as special actions. However, SEE WHY may be linked into WITNESS when WITNESS cannot achieve a particular complexity. This most important feature reflects Istel's commitment to retain modeling flexibility while bringing ease of use to manufacturing modeling.

REFERENCES

Istel Inc., SEE WHY VISUAL INTERACTIVE SIMULATION REFERENCE MANUAL, January 1984.

Istel Inc., INTRODUCTION TO WITNESS - USER MANUAL, July 1986.

AUTHORS' BIOGRAPHIES

ANDREW R. GILMAN is currently Technical Consultant with Istel Incorporated. He is responsible for training and technical support coordination for SEE WHY and WITNESS, visual interactive simulation software packages. Before joining Istel he worked as an Industrial Engineer for the Xerox Corporation where his duties included simulation projects in the North American Manufacturing Division. Mr. Gilman received a B.S. degree in Operations Research from Cornell University in 1979.

Andrew R. Gilman
Istel Incorporated
83 Cambridge Street
Burlington, MA 01803 USA
(617) 272-7333

RENEE WATREMEZ is a Technical Consultant with Istel, Incorporated. She received her Bachelor of Science degree in Industrial Engineering and Operations Research from the University of Massachusetts. Prior to joining Istel, Inc. Renee was employed by Wang Laboratories. Her experience includes simulation studies in such areas as plant startups and receiving operations. Her present responsibilities include training, consulting and technical support for SEE WHY and WITNESS, visual, interactive simulation software packages.

Renee M. Watremez
Istel Incorporated
83 Cambridge Street
Burlington, MA 01803 USA
(617) 272-7333