HSS: A TOOL FOR EVALUATING THE PERFORMANCE OF OFFICE SYSTEMS

Richard L. Gimarc and K. Mani Chandy

Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712

TR-85-12 June 1985

HSS: A Tool for Evaluating the Performance of Office Systems

Richard L. Gimarc
K. Mani Chandy

The University of Texas at Austin
Department of Computer Sciences
Austin, Texas 78712

1. Abstract

The automated office is often referred to as the office of the future. This may have been true a few years ago, but with the proliferation of information processing equipment such as personal workstations, local area networks, file servers, and specialized output devices, the automated office is quickly becoming the office of today. Office planners are currently faced with the complex task of evaluating à myriad of possible office system configurations to determine which will satisfy their performance requirements. To perform this task, office planners need a tool which will enable them to easily and effectively evaluate alternative configuration strategies.

The Hierarchical Simulation System (HSS) has been designed to be a tool office planners use to plan for the automated office. HSS is an interactive discrete-event simulation system which provides the facilities for building, storing, managing, and executing office system simulation models. HSS has a rich set of generic model building blocks such as local area networks, file servers, bridges, and workstations. These building blocks are an integral part of the system and are parameterized to simplify the modeling of a wide variety of information processing equipment. The library of building blocks is growing; our goal is to include building blocks for all equipment commonly used in office systems.

The user interface to HSS is through the HSS Workbench. The Workbench was designed for the novice computer user and consists of a set of full-screen displays and menus which provide access to all the facilities of HSS. The Workbench is complemented with extensive input validation, on-line help, and a tutorial which greatly increases the utility of the system.

HSS also provides the ability to execute a single simulation model concurrently on multiple processors. Distributed simulation is currently the topic of much research and has the potential of greatly reducing the often lenghty elapsed time required to execute a simulation model.

2. Introduction

The office of today is quickly evolving from the typical IN/OUT boxes, typing pools, and rows of dusty file cabinets into an automated office of personal workstations, word processors, electronic mail facilities, and communication networks. Careful planning is required to select equipment with the required functions and capabilities and to determine how the selected equipment should be configured.

Office planners need a tool which they can use to evaluate the myriad of possible office system configurations. This tool must be easy to use and flexible enough to facilitate the timely evaluation of alternative equipment and configuration strategies. This tool must also speak the language of the office planner for both describing a proposed office system and presenting the evaluation results.

The Hierarchical Simulation System (HSS) has been designed to address the modeling needs of the office planner. HSS is an interactive discrete-event simulation system which provides the office planner with a complete problem solving environment. With HSS, the office planner can build and evaluate office system models. The models are built in terms the office planner is familiar with (workstations, local area networks, file servers, etc.) and model execution results are presented in terms that are meaningful to the office planner.

3. Design Goals

The four major design goals proposed for HSS were

Goal 1: HSS must be an effective tool for modeling office systems. The system should be able to model

a. office system components such as local area networks, workstations, file servers, etc.

b. the flow of information between the various office system components

c. the logical structure of the office system

Goal 2: Design a system which could easily be used by the novice office planner. We did not want to require any knowledge of computer programming and only limited computer skills. This meant that HSS should require minimal user interaction, provide intelligent error detection and recovery, and assist the novice planner as much as possible during a modeling session.

Goal 3: The system should be flexible and powerful enough to be of practical value to the experienced user, yet retain those features which would benefit the novice.

Goal 4: Develop a system which could evolve over time to meet the needs of its user community. Since it is impractical to expect a single system to address all the needs of all its users, we wanted to design a system which could easily be extended to satisfy future requirements. The system should be structured in such a way that users could make the modifications themselves without having to rely on the system developers.

The Hierarchical Simulation System was designed and developed with these goals in mind.

4. HSS Architecture

The architecture of HSS is illustrated in Figure 1. All user interaction with HSS is through the HSS Workbench. The Workbench presents the user with a set of full-screen displays and menus which provide access to all the facilities of HSS.

The two major facilities provided by the Workbench are Model Build and Model Utilities. The Model Build facility is used to build an HSS simulation model. When model building is complete the model may be saved in the HSS model library. Model Utilities provides a set of utilities for manipulating the models stored in the HSS model library. Utilities are provided for updating, browsing, copying, renaming, deleting, and executing a model.

The architecture of HSS enables the system to be completely self-contained. All the services and functionality required to build, store, execute, and manipulate HSS models are provided by the Workbench. This feature promotes the usefulness of the system in that familarity with an editor, compiler, linker, or other software package is not required to make effective use of the system.

5. User Interface

The user interface to HSS is through a set of full-screen displays and menus managed by the HSS Workbench. The complexity usually present in developing interactive applications was alleviated by using the Dialog Manager component of the Interactive System Productivity Facility (ISPF), an IBM program product [Joslin 81] [IBM 81].

The Dialog Manager provides the control facilities and services which simplify the development of full-screen interactive applications. Displays may be defined which make full use of terminal display characteristics such as high and low intensity field display, protected fields, and cursor control. Services are also provided which support program function (PF) key recognition, display of error messages, input field validation, and on-line help and tutorials.

The HSS Workbench makes full use of the services provided by the Dialog Manager to present a set of panels which are well-defined, consistent, and structured in a manner which makes the system easy to user yet remain flexible. All input is checked for validity, and if an error is detected a descriptive error message is displayed. The input error must then be corrected before proceeding any further. Also, each panel is backed up by a help panel which may be displayed by pressing a PF key. The help panels provide a concise on-line reference manual which further enhances the usability of the system.

6. Modeling a Hypothetical Office System

This section will present a hypothetical office system and show how HSS can be used to evaluate its performance.

The Widget Corporation manufactures and sells the general-purpose widget. The Manufacturing Division has sole responsibility for producing widgets. Data entry terminals are located throughout the Manufacturing Division for entering and updating production related data such as inventory counts in the Manufacturing Division data base.

The Sales Division is responsible for selling widgets. Each of the four salesman in this division has his own workstation through which he may place orders and query the production data kept in the Manufacturing Division data base. All sales orders are placed in the Sales Division data base.

Each division has its own local area network (LAN) connecting the terminals and workstations in the division. There is also a bridge between the two LAN's to allow salesmen to query the Manufacturing Division data base.

6.1 Modeling Methodology

HSS embodies a modeling methodology which provides a framework which structures the perception, representation, and reasoning about office systems. This methodology enables the office planner to design and construct an HSS simulation model which directly reflects the hierarchical structure of the office system being modeled. This hierarchical structure is then simulated by HSS.

There are two steps in the HSS modeling methodology:

Step 1: decompose the office system into a hierarchical collection of information processing subsystems

Step 2: describe the information paths in the office system

Information Processing Subsystems

An office system can be viewed as a hierarchical collection of information processing subsystems. At the highest level, the office system can be decomposed into first-level subsystems. This decomposition may, for example, be driven by business areas, physical location, or department boundaries. After the first-level subsystems have been identified they may themselves be decomposed into lower-level subsystems. The result of this decomposition is a hierarchical collection of information processing subsystems. The decomposition of an office system into a hierarchy of information processing subsystems may be obtained in a manner analogous to the programming methodology of stepwise refinement described in [Wirth 71].

The HSS modeling methodology enables the office planner to obtain a hierarchical description of an office system by a sequence of refinement steps. Initially the office system being modeled is viewed as a single entity. The first step in the refinement process is to identify the major subsystems of the office system. These major subsystems are then decomposed repeatedly until the components of the office system (workstations, LAN's, file servers, etc.) appear as leaves on the refinement tree. The resulting tree structure is referred to as the HSS model tree.

The HSS model tree represents the hierarchical structure of the office system being modeled. Each subtree represents a logical subsystem. Unlike the "program tree" built during program design using stepwise refinement, the HSS model tree is seen directly in the HSS simulation model. All vertices in the model tree have a corresponding representation in the simulation model.

The Widget Corporation office system described earlier can be decomposed along business area boundaries into manufacturing and sales. The manufacturing subsystem can then be decomposed into the following components: data entry terminals, LAN, and the Manufacturing Division data base. The sales subsystem can be decomposed in a similar manner: workstations, LAN, and the Sales Division data base. But there is still one component which we have not included, the bridge connecting the two LAN's. Since the bridge is not a proper component of either the sales or manufacturing subsystem, we will put it at the same level as the manufacturing and sales subsystems in the model tree. Figure 2 shows the information processing subsystems for the Widget Corporation.

Information Paths

An office system is composed of both information processing equipment and the information that is processed by that equipment. Information in an office generally flows along fixed paths. That is, information originates at an information source, travels through various stages of information processing and finally reaches its destination. The source and destination can usually be identified as specific components in the office system. An information path is the sequence of office system components through which information flows. There may be several information paths within an office system, each characterized by the type of information that flows along the path. And further, information paths need not be disjoint.

To complete the design of an HSS model the information paths must be identified. The office system components which appear on these paths will be the leaf nodes of the HSS model tree.

The information paths in the Widget Corporation model can be identified by examining the office system information sources. In the manufacturing subsystem information originates at the data entry terminals. From there it goes to the LAN and then to the data base. Information originates in the sales subsystem at the sales workstations. For each workstation there are two paths, one that goes directly to the Sales Division data base and another that goes to the Manufacturing Division data base. Also, workstations differ from data entry terminals in that a workstation requires that a replay be returned for every request entered. Thus, the information paths originating at a workstation will also terminate at the workstation. Figure 3 shows the information paths for the Widget Corporation model.

6.2 HSS Simulation Model

An HSS simulation model may be constructed in a mechanical fashion directly from the design obtained by applying the HSS modeling methodology. The simulation model is constructed using the same two-step procedure applied during model design. Namely,

Step 1: build the model tree

Step 2: build the information paths

There are three basic components which are used to construct an HSS simulation model: messages, message processing nodes, and message routing paths.

Messages

Messages are used to model the information that flows within an office system. For example, mail distributed by an electronic mail system and data transmitted on a local area network would each be modeled by a message. A message represents a distinct piece of information, there are no dependencies or relationships between messages. To facilitate modeling different types of information, each message has a set of three attributes: size, priority, and local variables. These attributes may be used to characterize the amount of information modeled by a message, the relative importance of a message, and control the flow of the message through the modeled office system.

Message Processing Nodes

HSS provides message processing nodes to model the various components of an office system that process information. Table 1 lists the seven categories of node types supported by HSS. Each node type is generic and parameterized to facilitate modeling of a large class of office system equipment.

Message generating nodes provide the only mechanism to introduce messages into an HSS simulation model. A message source node will generate messages at a user-specified rate. A workstation node also generates messages but a message will be generated only after the previously generated message has returned to the workstation. After a message is generated, the workstation enters a wait state and will remain in the wait state until the message it just generated returns. When the message returns the workstation enters a think-time period of user-specified duration. When the think-time period is over another message is generated and the cycle repeats. The node definition panels for a workstation node are shown in Figures 4 and 5. All user input is shown in lower case.

Message absorbing nodes remove messages from a model. A message sink node simply absorbs messages with no side effects. A workstation node also absorbs messages, but when it does the node enters a think-time period as described above.

The service center nodes model servers which operate under a specified scheduling discipline. Each message arriving at a service center node carries its service time requirement in a message attribute. After a message has received its required service it will leave the service center node. The node definition panel for a FCFS server node is shown in Figure 6.

Assignment nodes provide the means to set and manipulate message attributes. An integer sampler node will set a message attribute with a sample drawn from a user-specified distribution. A computation node will evaluate a simple arithmetic expression involving message attributes and constants, the result is then stored in a message attribute.

Local area network nodes model an entire LAN. The stations, communication medium, and access protocol are all modeled by the LAN node. When a message arrives at a LAN node, it will be transmitted by a source station and received by a destination station on the LAN. The message will not leave the LAN node until it has been received by the destination station. The token ring definition panel is shown in Figure 7.

A protocol node is designed to be used with a LAN node to provide the services of a high-level protocol on the LAN. The only protocol node supported at present is for the High-Level Data Link Control (HDLC) protocol.

User nodes will be discussed later.

HSS node types hide the details of the generic objects they model. Office planners do not have to concern themselves with the time consuming task of writing code to simulate a file server or local area network. Instead, all they have to do is select the appropriate HSS node type and supply the required parameters. HSS nodes provide the high-level abstraction required by the office planner, and nodes are parameterized in terms the office planner is familiar with, not the often cryptic parameters usually found in simulation models.

Message_Routing_Paths

Message routing paths are used to model the information paths in an office system model. For each piece of information (message) originating at an information source (message generating node), the routing path lists those components of the office system that the information passes through (message processing nodes) before it reaches its destination (message absorbing node).

A routing path consists of a list of message processing nodes. The first node in the list is a message generating node and the last is a message absorbing node. Intermediate nodes may be any of the other types of message processing nodes listed in Table 1.

Model_Structure

Designing an HSS simulation model in a sequence of decomposition and refinement steps, as described earlier, results in a tree structure referred to as the HSS model tree. The model tree represents a hierarchy of subsystems identified in the office system being modeled. These subsystems may be completely isolated from each other, but more likely, there is some interaction between subsystems. That is, information will flow from one subsystem to another.

HSS refers to the information processing subsystems as communication systems. Each communication system consists of a communication system manager node and its immediate children nodes. To form a hierarchy, children nodes may themselves be communication system manager nodes. Thus, all non-leaf nodes in the model tree are communication system manager nodes and leaf nodes are message processing nodes.

The communication system manager node was not listed as a message processing node in Table 1 since it is merely a modeling construct used to structure the simulation model into hierarchical subsystems.

6.3 Using the System

All facilities provided by HSS are accessed through the HSS PRIMARY OPTION MENU shown in Figure 8. The two major facilities, MODEL BUILD and MODEL UTILITIES may be invoked by entering a 1 or 2, respectively, in the SELECT OPTION input field. The P option code will invoke the Program Development Facility, an IBM program product. TUTORIAL will bring up a set of panels describing the major features of HSS. The TUTORIAL is not intended to give an in-depth description of HSS, but rather it describes what facilities are available and how they may be used.

Model Build

MODEL BUILD provides the means to build an HSS simulation model. As noted earlier, an HSS simulation model can be built directly from the design obtained by applying the HSS modeling methodology.

After selecting the MODEL BUILD option on the HSS PRIMARY OPTION MENU the HSS MODEL BUILD panel shown in Figure 9 will be displayed. On this panel the name of the model being built, WIDGET, and an optional description are entered.

The model tree is built top-down, exactly the way the model was designed. The first node defined in the model tree is the top-level communication system manager. After this node is built the remaining nodes in the model tree may be built in a depth-first and/or breadth-first manner. The only restriction on how the model tree is built is that it must be built top-down. HSS will not allow a model tree to be built any other way. The communication system manager node definition panel is shown in

Figure 10. This figure shows the definition of the top-level communication system manager, WDGTCORP.

Once the root of the model tree has been defined the HSS MODEL UTILITY panel is displayed (see Figure 11). This is the primary panel used during model build (and model update). Since the model tree may be arbitrarily large, a windowing technique is used to display a subset of the model tree, namely a communication system. The CURRENT MODEL VIEW displays the current communication system in the viewing window. The name of the communication system manager node (COMMUNICATION SYSTEM), the name of the next node up the tree (NEXT NODE UP TREE), and the names of all the nodes within the communication system (NODE NAME) are shown on the panel. If the communication system contains too many nodes to be displayed on one panel, PF keys may be used to scroll forward and backward through the NODE NAME list.

A node may be defined by typing +<code> in the COMMAND input field where <code> identifies the type of node being defined (see Table 1). For example, to model the bridge connecting the two LAN's in the Widget Corporation model with a FCFS server, we enter +FCFS to bring up the FCFS server definition panel shown earlier in Figure 6. Whenever a node is defined it is added to the communication system shown in the viewing window.

Two other commands are provided for manipulating nodes in the viewing window. The U and D SELECT CODES may be entered next to a node name. U will retrieve the node definition for updating and D will delete the subtree rooted at the selected node.

The viewing window may be moved up and down the model tree. Typing the SELECT CODE M next to a node will move the viewing window down to the communication system rooted at the selected node if that node is a communication system manager. To move the window up the tree, U may be entered in the COMMAND input field.

After the model tree has been built routing paths may be defined by entering R in the COMMAND input field. The next panel to be displayed will be the MESSAGE ROUTING FACILITY panel shown in Figure 12. This panel lists all message generating nodes defined in the model along with a status of the routing paths originating at the node. The routing path for a single node may be built or updated by entering an S next to the node name. Alternatively, entering a D next to a node name will delete the routing paths currently defined for that node.

The routing path for workstation SELLER1 is shown in Figure 13. Ninety percent of the messages generated by SELLER1 will access the Sales Division data base. A message generated by SELLER1 is transmitted on the Sales Division LAN, SALESLAN, to the data base, SALESDB. After accessing the data base, the message is again transmitted in the LAN back to SELLER1.

Model building may be terminated at any time from the HSS MODEL UTILITY panel. Typing an S in the COMMAND input field will save the model in the HSS model library and return control to the HSS PRIMARY OPTION MENU. The C command will cancel the model build session (the model is not saved).

Model Utilities

MODEL UTILITIES provides utilities for manipulating models stored in the HSS model library. Figure 14 shows the HSS MODEL LIBRARY UTILITY panel. This panel displays the HSS model library directory which contains an entry for each model stored in the model library. Each directory entry contains the name of the model, the date and time the model was last modified, a model status indicator, and a short descriptive comment.

Model status is used by the HSS Workbench to signal deficiencies in the model definition. Only those models with a valid model definition, model status is 'O.K.', are eligible for execution. Deficiencies monitored by the Workbench include a model without any message generating nodes, model status is '(NM)' for No Messages, and models with incomplete routing paths, model status is '(RI)' for Routing Incomplete.

The utilities provided by MODEL UTILITIES are those which are usually provided by any well-designed library management system. The COPY utility will make an exact copy of a model. The RENAME and DELETE utilities allow a model to be renamed or deleted, respectively. BROWSE will display a model definition in read-only mode. And the UPDATE utility retrieves a model from the model library and allows it definition to be modified.

The EXECUTE utility will execute a model. As noted earlier, only those models with a status of 'O.K.' may be selected for execution. Model execution will be discussed in the next section.

All the utilities are invoked by moving the cursor to the left of a model name, typing the desired utility code, and pressing the ENTER key. If there is not enough room on the display to list all models stored in the model library, PF keys are supported for scrolling forward and backward through the directory.

Model Execution

A model may be selected for execution from the HSS MODEL LIBRARY UTILITY panel (Figure 14). As noted earlier, only those models with a model status code of 'O.K.' are eligible for execution. HSS will reject an execution request for a model without the 'O.K. status code. After the model has been selected the RUN-TIME PARAMETERS panel shown in Figure 15 is displayed.

The RUN-TIME PARAMETERS panel is used to specify execution-time parameters for the model. Parameters such as the duration of the simulation, the initial seed for the simulator's random number generator, and the time interval between simulation progress report messages may be specified.

The last two fields on this panel control the collection of intermediate statistics for the computation of confidence intervals. Confidence intervals are used to obtain an indication of the accuracy of the simulation results. HSS uses the method of batch means to compute 90% confidence intervals for several of the reported simulation results [Sauer 81]. With batch means, the model is executed for a batch where BATCH SIZE specifies the number of simulation events in a batch. Results from each batch are accumulated and confidence intervals are computed.

The model is executed in batch increments. After the specified minimum number of batches have been executed (MINIMUM NUMBER OF BATCHES), the HSS SIMULATION END OF BATCH panel is displayed (see Figure 16). This panel shows how far the simulation has progressed and what batch has just completed. Several options are available on this panel. The simulation may be terminated by entering the command QUIT, or the next batch may be started with the command GO.

Intermediate simulation results may also be viewed. The lower portion of the panel lists all nodes in the simulation model. By placing a SELECT CODE next to a node name, the requested results for the subtree rooted at the selected node may be printed or examined on-line. The S and C SELECT CODEs will display batch statistics and confidence intervals, respectively. SP and CP will write the same results to a file for printing at a later time.

7. System Extensions

The one aspect of HSS which some users may view as deficient is its collection of message processing nodes. Clearly, not all equipment which may be part of an office system may be modeled with the nodes currently supported by HSS. To handle these special cases, a special node type, a user node, is provided.

A user node provides the mechanism to add special-purpose nodes to HSS. For example, it might be necessary to have a preemptive service center node or another type of local area network. User nodes are identified by a unique user node number. Thus, several user nodes may be added to the system, each with its own unique user node number.

The implementation of a user node is not difficult, but it does require some knowledge of programming. The HSS Reference Manual contains a detailed description of the steps required to add a user node to the system.

8. Distributed Simulation

One of the major drawbacks of simulation is that the run-time may be prohibitively long. One way to reduce the run-time is to use multiple processors to execute a single simulation model. This is called distributed simulation. By judiciously partitioning a single model and assigning each partition to a separate processor, the concurrency inherent in the model may be exploited by having separate parts of the model executing simultaneously on separate processors. In this way the model will execute in less time than required on a single processor.

Synchronization of the multiple processors is the most challenging problem with distributed simulation. Since each processor is executing asynchronously, there must be some mechanism to synchronize two processors when their respective model segments must interact with one another. The majority of the research that has been performed on distributed simulation has been aimed at developing mechanisms which provide the required synchronization yet strive for maximum concurrency. The most notable work in this area has been done by Chandy and Misra [Chandy 79] [Chandy 81]; Peacock, Wong, and Manning [Peacock 79] [Peacock 80]; and Jefferson and Sowizral [Jefferson 82].

The HSS model tree may be partitioned at communication system boundaries. That is, each subtree rooted at a communication system manager node may be assigned to a separate processor. The model may then be executed. Each processor will execute its portion of the model, and synchronization will occur when messages are sent from a message processing node on one processor to a message processing node on another processor.

The results of our research describing the synchronization technique and empirical performance results will be the subject of a forthcoming paper.

9. Implementation

HSS currently runs on an IBM mainframe under the VM/SP operating system. The IBM program product ISPF is required for display management. The HSS Workbench consists of approximately 5,000 lines of PL/1, 120 panel definitions, and 1,000 lines of EXEC2 code. The HSS Simulator is approximately 30,000 lines of FORTRAN.

10. Summary

In this paper we have described a new tool for evaluating office system performance, the Hierarchical Simulation System. HSS is of immediate practical value to the industry. HSS was targeted at a specific industry segment, office systems. By restricting the audience of HSS we have been able to refine and polish the system into a viable tool for studying office system performance.

Throughout this paper we have tried to emphasize the fact that HSS is easy to use. Ease of use was obtained by developing a flexible user-interface and by introducing a modeling methodology which simplifies the design and construction of office system simulation models.

HSS is an evolving tool. New features and functions are being added and suggestions from early users of the system are the subject of future enhancements.

HSS is in the public domain. It is being developed as a research project at the University of Texas at Austin with support from the IBM Corporation. HSS is not a product and it is not sold. Requests for additional information regarding HSS should be sent to K. M. Chandy, Computer Sciences Department, University of Texas at Austin, Austin, Texas 78712.

11. Acknowledgements

Special thanks goes to Fred May of IBM Austin for his encouragement and advice during the development of HSS. Brooks Hickerson of IBM Austin also deserves thanks for his comments and suggestions on how to improve the early versions of HSS. Also, Paul Attie and Robert Berry of the University of Texas are thanked for their work and help with the HSS Simulator.

12. References

[Chandy 79] K. M. Chandy and J. Misra "Distributed Simulation: A Case Study in Design and Verification of Distributed Programs", IEEE Transactions on Software Engineering, Vol SE-5, No 5, September 1979

[Chandy 81] K. M. Chandy and J. Misra "Asynchronous Distributed Simulation via a Sequence of Parallel Computations ", CACM, Vol 24, No 11, April 1981

[IBM 81] Interactive System Productivity Facility, Dialog Management Services, SC34-2088 (1981)

[Jefferson 82] D. R. Jefferson and H. A. Sowizral "Fast Concurrent Simulation Using the Time Warp Mechanism. Part I: Local Control", Technical Report N-1906-AF, The Rand Corporation, Santa Monica, CA, July 1982

[Joslin 81] P. H. Joslin "System Productivity Facility", IBM Systems Journal, Vol 20, No 4, 1981

[Peacock 79] J. K. Peacock, J. W. Wong, and E. Manning "Distributed Simulation Using a Network of Processors", Computer Networks, Vol 3, No 1, February 1979

[Peacock 80] J. K. Peacock, E. G. Manning, and J. W. Wong "Synchronization of Distributed Simulation Using Broadcast Algorithms ", Computer Networks, Vol 4, No 1, Februrary 1980

[Sauer 81] C. H. Sauer and K. M. Chandy Computer System Performance Modeling, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981

[Wirth 71] N. Wirth "Program Development by Stepwise Refinement", CACM, Vol 14, No 4, 1971

Message Generating Nodes
- Message Source (SRCE)
- Workstation (WORK)

Message Absorbing Nodes
- Message Sink (SINK)
- Workstation (WORK)

Service Center Nodes
- FCFS Server (FCFS)
- Priority Server (PRIQ)
- Delay Node (DLAY)

Assignment Nodes
- Computation Node (COMP)
- Integer Sampler (ISMP)

Local Area Network Nodes
- Token Ring (TOKN)
- CSMA/CD (CSMA)
- Register Insertion Ring (RIRG)

Protocol Nodes
- HDLC/ABM (HDLC)

User Nodes

Table 1. Node Categories

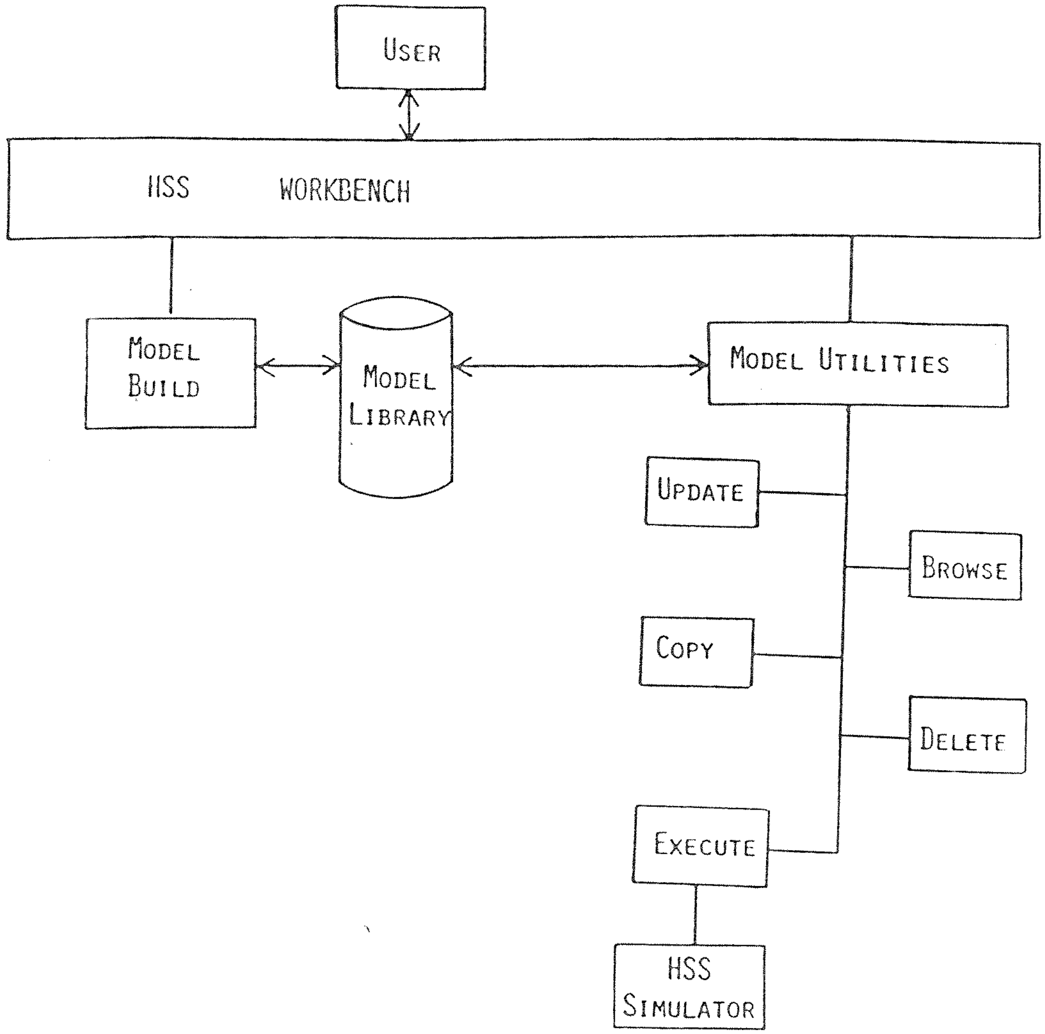

Figure 1. HSS Architecture

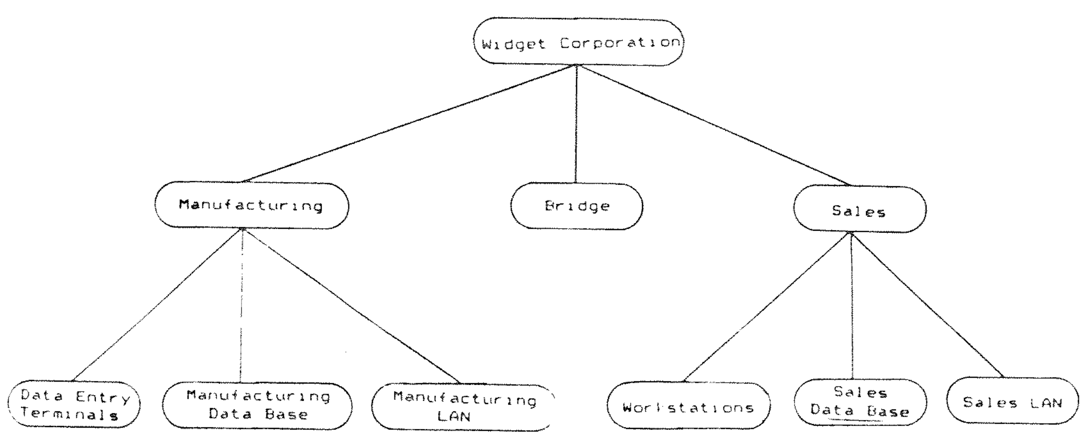

Figure 2. Widget Corp. Information Processing Subsystems

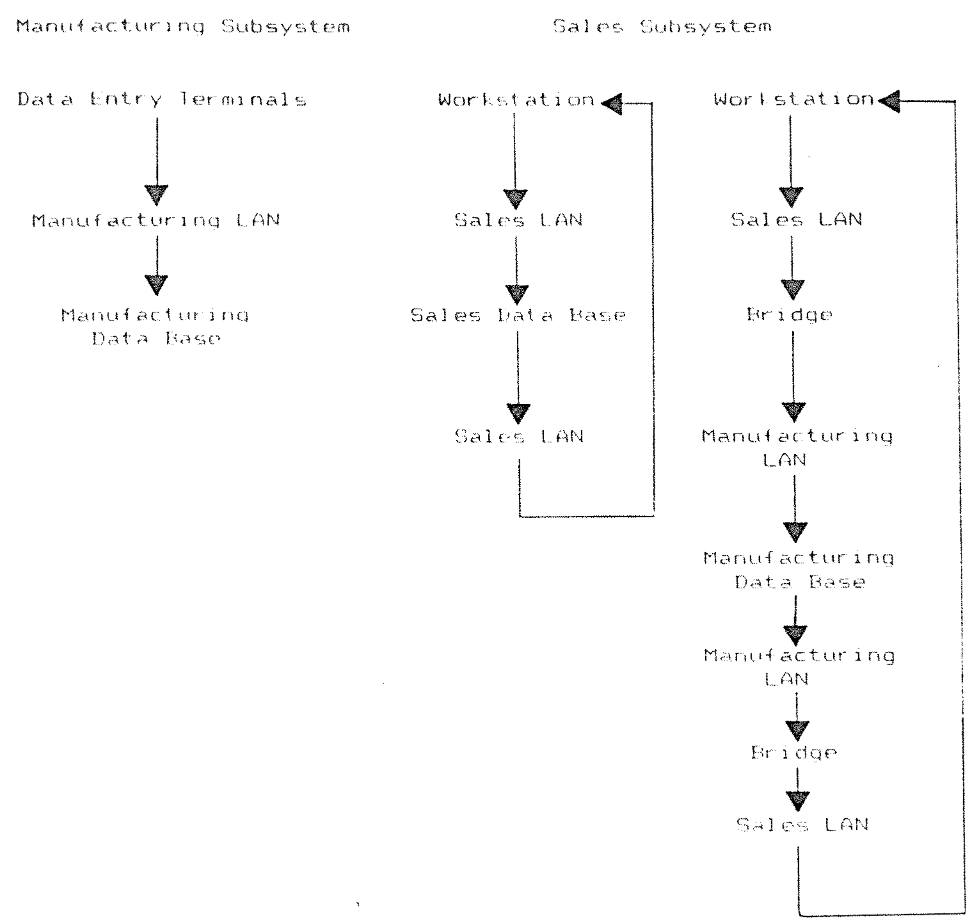

Figure 3. Widget Corp. Information Paths

```
------------------------------- BUILD WORKSTATION -------------------------------
COMMAND ==>

   MODEL NAME: WIDGET                        COMMUNICATION SYSTEM: SALES

   ENTER THE PARAMETERS BELOW AND PRESS THE ENTER KEY TO BUILD THE NODE.

      NODE NAME ==> seller1  IS SAME-AS NODE ==>
      COMMENT   ==> workstation for salesman number 1

      DISTRIBUTION TYPES:                 --- DISTRIBUTION TYPE CODES ---
         MESSAGE SIZE ==> iu                    ER, EX, HE, IU, NM
         THINK-TIME   ==> ex
```

Figure 4. Workstation Definition Panel (Part 1)

```
-------------- BUILD WORKSTATION DISTRIBUTION PARAMETERS --------------
COMMAND ==>

   MODEL NAME: WIDGET                         COMMUNICATION SYSTEM: SALES

   ENTER THE PARAMETERS BELOW AND PRESS THE ENTER KEY TO BUILD THE NODE.

      NODE NAME : SELLER1
      COMMENT   : WORKSTATION FOR SALESMAN NUMBER 1

      MESSAGE SIZE DISTRIBUTION: INTEGER UNIFORM
         INTERVAL LOWER BOUND ==> 1000
         INTERVAL UPPER BOUND ==> 2000

      THINK-TIME DISTRIBUTION: EXPONENTIAL
         TIME UNIT ==> min                    --- TIME UNIT CODES ---
         MEAN      ==> 5                        MIN, SEC, MSEC, USEC
```

Figure 5. Workstation Definition Panel (Part 2)

```
------------------------- BUILD FCFS SERVER --------------------------
COMMAND ==>

   MODEL NAME: WIDGET                       COMMUNICATION SYSTEM: WDGTCORP

   ENTER THE PARAMETERS BELOW AND PRESS THE ENTER KEY TO BUILD THE NODE.

      NODE NAME ==> bridge   IS SAME-AS NODE ==>
      COMMENT   ==> bridge between sales and manufacturing divisions

      SERVICE TIME LOCATION ==> s       -- SERVICE TIME LOCATION CODES --
                                                   L<X>, S
      SERVICE TIME UNIT     ==> msec    ---- SERVICE TIME UNIT CODES ----
                                              MIN, SEC, MSEC, USEC
```

Figure 6. FCFS Server Definition Panel

```
------------------------- BUILD TOKEN RING -------------------------
COMMAND ==>

   MODEL NAME: WIDGET                    COMMUNICATION SYSTEM: SALES

   ENTER THE PARAMETERS BELOW AND PRESS THE ENTER KEY TO BUILD THE NODE.

      NODE NAME ==> saleslan IS SAME-AS NODE ==>
      COMMENT   ==> 5-station sales division lan

      PROTOCOL NODE ==>

      NETWORK PARAMETERS:
         NUMBER OF STATIONS ==> 5
         TOKEN LENGTH       ==> 18
         RING CAPACITY      ==> 1   mbps      - STATION SERVICE CODES -
         STATION SERVICE    ==> 1                      <N>, E

      RESPONSE TIME HISTOGRAM PARAMETERS:     ---- TIME UNIT CODES ----
         BUCKET COUNT ==> 10                    MIN, SEC, MSEC, USEC
         BASE         ==> 0        UNIT ==> sec
         GRANULARITY  ==> 100      UNIT ==> msec
```

Figure 7. Token Ring Definition Panel

```
----------------------- HSS PRIMARY OPTION MENU -----------------------
SELECT OPTION ==>

   1  BUILD     - BUILD A NEW MODEL                   USERID   - USER1
   2  LIBRARY   - MODEL LIBRARY UTILITY               TIME     - 13:05
                  BROWSE, UPDATE, EXECUTE, COPY,      TERMINAL - 3277
                  DELETE, AND RENAME A MODEL          PF KEYS  - 12


   P  PDF       - ENTER PROGRAM DEVELOPMENT FACILITY (PDF/SPF)
   T  TUTORIAL  - DISPLAY INFORMATION ABOUT HSS
   X  EXIT      - TERMINATE HSS SESSION USING LIST/LOG DEFAULTS
```

Figure 8. HSS Primary Option Menu

```
------------------------------- HSS MODEL BUILD --------------------------------
COMMAND ==>


   ENTER THE PARAMETERS BELOW AND PRESS THE ENTER KEY TO BEGIN MODEL BUILD


      MODEL NAME        ==> widget
      MODEL DESCRIPTION ==> widget corporation model
```

Figure 9. HSS Model Build Panel

```
-------------------- BUILD COMMUNICATION SYSTEM MANAGER ------------------------
COMMAND ==>

   MODEL NAME: WIDGET                    COMMUNICATION SYSTEM: (NONE)

   ENTER THE PARAMETERS BELOW AND PRESS THE ENTER KEY TO BUILD THE NODE.

      NODE NAME ==> wdgtcorp IS SAME-AS NODE ==>
      COMMENT   ==> top-level communication system manager

      RESPONSE TIME HISTOGRAM PARAMETERS:          ---- TIME UNIT CODES ----
         BUCKET COUNT ==> 10                          MIN, SEC, MSEC, USEC
         BASE         ==> 0          UNIT ==> sec
         GRANULARITY  ==> 1          UNIT ==> sec
```

Figure 10. Communication System Manager Definition Panel

```
------------------------------ HSS MODEL UTILITY ------------------------------
COMMAND ==>                                                    SCROLL ==> HALF

  MODEL NAME: WIDGET   ( WIDGET CORPORATION MODEL )

  COMMANDS: +<CODE>, P, R, U, C, S
  SELECT CODES: D, M, U

----------------------------- CURRENT MODEL VIEW ------------------------------
                        COMMUNICATION SYSTEM: WDGTCORP
                         NEXT NODE UP TREE: (NONE)

    NODE NAME  NODE CODE  COMMENTS
    ---------  ---------  ------------------------------------------------
    WDGTCORP     COMM     TOP-LEVEL COMMUNICATION SYSTEM MANAGER
    MANUF        COMM     MANUFACTURING DIVISION SUBSYSTEM
    SALES        COMM     SALES DIVISION SUBSYSTEM
```

Figure 11. HSS Model Utility Panel

```
-------------------------- MESSAGE ROUTING FACILITY ---------------------------
COMMAND ==>                                                    SCROLL ==> HALF

   TO EXAMINE/DELETE A MESSAGE ROUTE, ENTER THE APPROPRIATE SELECT CODE
      NEXT TO THE MESSAGE GENERATING NODE NAME AND PRESS THE ENTER KEY.

   SELECT CODES: S - EXAMINE/UPDATE ROUTING
                 D - DELETE ROUTING


      NODE          NODE         ROUTING
      NAME       DESCRIPTION     STATUS
    --------   --------------    -------
    DATAENTY   MESSAGE SOURCE    O.K.
  s SELLER1    WORKSTATION       (NR)
    SELLER2    WORKSTATION       (NR)
    SELLER3    WORKSTATION       (NR)
    SELLER4    WORKSTATION       (NR)
```

Figure 12. Message Routing Facility Panel

```
---------------- MESSAGE ROUTING: INITIAL ROUTE SEGMENT ----------------
COMMAND ==>

   MODEL NAME: WIDGET                MESSAGE GENERATING NODE: SELLER1

   ENTER THE PROBABILITY OF THE ROUTING PATH AND THE NODES TO BE VISITED
      ALONG THE PATH THEN PRESS THE ENTER KEY.

   PATH PROBABILITY ==> .9

       SELLER1
       |
       +--> saleslan
            |
            +--> salesdb
                 |
                 +--> saleslan
                      |
                      +--> seller1
                           |
                           +-->

   CONTINUE ROUTING PATH ? ==> NO   (YES OR NO)
```

Figure 13. Message Routing Panel

```
------------------------- HSS MODEL LIBRARY UTILITY -------------------------
COMMAND ==>                                                  SCROLL ==> HALF

  ENTER A SELECT CODE NEXT TO THE DESIRED MODEL AND PRESS ENTER

  SELECT CODES: B - BROWSE A MODEL        R - RENAME A MODEL
                C - COPY A MODEL          U - UPDATE A MODEL
                D - DELETE A MODEL        X - EXECUTE A MODEL

     MODEL      LAST MODIFIED
     NAME       DATE       TIME    STATUS  MODEL DESCRIPTION
   ---------  ---------  -----  ------  --------------------------------
   HDLCLAN    02/22/85   08:20   (RI)    HDLC REGISTER INSERTION RING
   MM1        12/05/84   12:55   O.K.    SIMPLE M/M/1 QUEUE
   PRIORITY   11/12/84   09:30   O.K.    TEST PRIORITY ADJUSTMENTS
   SAMPLE1    03/15/85   20:17   O.K.    20-STATION TOKEN RING
 x WIDGET     10/22/84   19:10   O.K.    WIDGET CORPORATION MODEL
```

Figure 14. Model Library Utility Panel

```
------------------------------ RUN-TIME PARAMETERS ------------------------------
COMMAND ==>

                         MODEL NAME: WIDGET

   UPDATE THE PARAMETERS BELOW AND PRESS ENTER TO EXECUTE THE MODEL


      MODEL DESCRIPTION ==> WIDGET CORPORATION MODEL

      SIMULATION PROGRESS MESSAGE INTERVAL ==> 30        UNIT ==> SEC
      INITIAL RANDOM NUMBER GENERATOR SEED ==> 65537
      MAXIMUM SIMULATION DURATION          ==> 25        UNIT ==> MIN
      CONFIDENCE INTERVAL PARAMETERS:
         BATCH SIZE                 ==> 10000        -- TIME UNIT CODES --
         MINIMUM NUMBER OF BATCHES  ==> 3             MIN, SEC, MSEC, USEC
```

Figure 15. Run-Time Parameters Panel

```
------------------------ HSS SIMULATION: END OF BATCH 4 ------------------------
COMMAND ==>                                                     SCROLL ==> HALF

  COMMANDS: GO, QUIT
  SELECT CODES: C, S, CP, SP

  MODEL NAME: WIDGET  (WIDGET CORPORATION MODEL)
  START TIME=00:05.033210050, END TIME=00:10.110004502
  BATCH EVENT COUNT: 10003



    NODE NAME     NODE CODE
    ---------     ---------
    WDGTCORP        COMM
    BRIDGE          FCFS
    SELLER1         WORK
    SELLER2         WORK
    SELLER3         WORK
```

Figure 16. HSS Simulation End of Batch Panel