Howard University
Washington, D.C. 20059

IN-32-CR
136203
P- 23

NAG5-995

NCC Simulation Model: Phase II:

Simulating the Operations of the Network Control Center

(PROGRESS REPORT: DECEMBER 15, 1992)

By: Norman M. Benjamin
Dr. Arthur S. Paul
Dr. Tepper L. Gill

NCC Simulation Model: Phase II:

Simulating the Operations of the Network Control Center

By Norman M. Benjamin

Table of Contents

1. INTRODUCTION

The simulation of the NCC is in the second phase of development. This phase seeks to further develop the work performed in phase one. Phase one concentrated on the computer systems and interconnecting network. The focus of phase two will be the implementation of the network message dialogues and the resources (i.e schedulable elements of the SN) controlled by the NCC. These resources are requested, initiated, monitored and analyzed via network messages. In the NCC network messages are presented in the form of packets that are routed across the network. These packets are generated, encoded, decoded and processed by the network host processors that generate and service the message traffic on the network that connects these hosts. As a result, the message traffic is used to characterize the work done by the NCC and the connected network.

Phase one of the model development represented the NCC as a network of bi-directional single server queues and message generating sources. The generators represented the external segment processors. The server based queues represented the host processors. The NCC model consists of the internal and external processors which generate message traffic on the network that links these hosts. The external processors represented are the POCC, NASCOM, WSGT, FDF, SDPF and JSC. These connect to the internal processors which are the CCS, SPS and ITS. To fully realize the objective of phase two it is necessary to identify and model the processes in each internal processor. These processes live in the operating system of the internal host computers and handle tasks such as high speed message exchanging, ISN and NFE interface, event monitoring, network monitoring, and message logging. Inter process communication is achieved through the operating system facilities. The overall performance of the host is determined by its ability to service messages generated by both internal and external processors.

2. NCC OVERVIEW

The NCC is located at the Goddard Spaceflight Center and provides scheduling and control services to the NASA SN. The SN provide tracking and data acquisition services to a large community of low-earth-orbiting spacecrafts. Spacecraft data is down linked through TDRS to the WSGT. The data is then quality checked at the NASA NGT which is co-located with WSGT. The data is then forwarded over NASCOM's telecommunications links to the NCC at Goddard and relayed to the project users. User spacecraft telemetry data do not pass through the NCC.

The NCCDS is specifically responsible for scheduling, control, fault isolation and accountability to ensure that users receives their data as scheduled. The NCCDS is provided with support services from the FDF and the SDPF. The FDF provides tracking data analysis together with orbit determination for spacecraft vector generation. The SDPF receives and processes scientific data for users.

The NCCDS is devided into three major subsystems. The service planning segment (SPS), which schedules TDRSS services; the communication and control segment (CCS), which controls and monitors the quality of active services; and the intelligent terminal segment (ITS) with the intersegment network (ISN) which provides the interface between the console operators and the SPS and CCS. The specific functions of providing the management and resources for scheduling, controlling, and monitoring the performance of NASA's SN are summarized as follows:

•Scheduling of Network Resources

        Forecast Scheduling and conflict resolution
        Real-time scheduling and conflict resolution

•Network Performance Monitoring

        Ground equipment status messages from WSGT
        Data quality messages from NGT

•Acquisition Data Management (Secure and Un-secure)

        Non-secure- Routing of data from FDF to WSGT
        Secure: Generation of pointing data using the Acquisition Data facility software and subsequently transmitting vectors to WSGT.

•NCC Database Management

        Maintain a library of all current databases.
        Restores data as required.
        Troubleshooting of database anomalies
        Maintain all database documentation

•Network Fault Isolation

        Real-time fault isolation which is accomplished by teams of Performance Analysts and TDRSS Network Controllers.
        Post-event analysis performed by teams of TDRSS Network Analysis who evaluate electronically logged messages in the NCC data systems.

•Network Accountability and Reporting

        Operation of a Service Accounting System (SAS) that counts the data messages received from user services and uses the data to compute time and resource usage for billing purposes. This data is also used by the SN Network Anomaly Committee to evaluate and resolve all documented network anomalies.

3. MEASURING THE PERFORMANCE OF THE NCC

3.1. FRAMEWORK

Having decided to use a queuing-type model to represent the NCC, we are constrained by queuing theory to limited set of performance measures. These measures are basically:

Mean waiting time

Mean service time

Mean delay (waiting time plus service time)

Mean queue length

These are fairly stable statistics that should not vary greatly from one run to the next, if, of course, the run-times are sufficiently long to attain stability in the process.

Other measures such as:

Maximum waiting time
Maximum service time
Maximum delay (waiting time plus service time)
Maximum queue length

can be observed, but they are likely to vary substantially from one run to the next, even if the run-times are long and the system attains stability.

Still other measures such as:

Probability that the waiting time exceeds some threshold value $P(W > T_w)$
Probability that the service time exceeds some threshold value $P(W > T_s)$
Probability that the delay exceeds some threshold value $P(W > T_d)$
Probability that the queue length exceeds some threshold value $P(Q > q)$

can be computed, but the problem would be to establish threshold values that have some physical or operational significance.

We have suggested one measure of performance that is not typical that is "the minimum capacity of the server required for zero wait or no queuing". This measure will help to establish the required capacity for the processors being modeled.

At the current stage of development, all of these measures can be observed for the NCC as a system, and the CCS, the SPS, and the ITS as subsystems.

We are proposing that the next logical step in developing this model should be to separate the manual processing from the automated processing. This should have a tremendous impact on the fidelity of the model. It will allow us to take a more analytical look at different processes with the goal of optimizing the allocation of tasks and processes to manual and automated servers. It will allow us to conduct "What If" analyses. It will provide insights into design options for the SNC, and it can be the basis for a subsequent elaboration of the model to include cost as a measure of performance.

3.2. SELECTED INDICATORS OF THE NCC'S OPERATIONAL EFFECTIVENESS

The NCC has established specific quantifiable requirements that it must achieve in providing services to SN users. These requirements have been analyzed as a basis for selecting the following indicators of the operational effectiveness of the NCC:

3.2.1. Utilization of NCC's Communications Capacity

- Average communications capacity utilized by incoming messages (single user)
- Percentage of times that incoming messages (single user) exceeds 56 kilobits per second
- Maximum communications capacity utilized by incoming messages (single user)
- Average communications capacity utilized by incoming messages multiple user
- Percentage of times that incoming messages (multiple user) exceeds 112 kilobits per second
- Maximum communications capacity utilized by incoming messages multiple user
- Average queue time at the CCS from incoming messages
- Minimum communication speed to ensure zero queue time at the CCS from incoming messages
- Maximum queue time at the CCS from incoming messages
- Average communications capacity utilized by outgoing messages (single user)
- Percentage of times that outgoing messages (single user) exceeds 56 kilobits per second
- Maximum communications capacity utilized by outgoing messages (single user)
- Average communications capacity utilized by outgoing messages multiple user
- Percentage of times that outgoing messages (multiple user) exceeds 112 kilobits per second
- Maximum communications capacity utilized by outgoing messages multiple user
- Average queue time at the CCS from outgoing messages
- Minimum communication speed to ensure zero queue time at the CCS from outgoing messages
- Maximum queue time at the CCS from outgoing messages

3.2.2. Operational Effectiveness Measures

The following are suggested as indicators of the operational effectiveness of the NCC:

- Average delay [processing time plus queuing time] at the NCC
- Average delay at the CCS
- Average delay at the ITS
- Average delay at the SPS
- Average processing time for the NCC
- Average processing time for the CCS
- Average processing time for the ITS
- Average processing time for the SPS
- Average queue time within the NCC
- Average queue time at the CCS
- Average queue time at the ITS
- Average queue time at the SPS
- Maximum queue time within the NCC
- Maximum queue at the CCS
- Maximum queue at the ITS
- Maximum queue at the SPS
- Average Utilization--NCC
- Average utilization--CCS
- Average utilization--ITS
- Average utilization--SPS
- Minimum capacity of the NCC to ensure zero queue
- Minimum capacity of the CSS to ensure zero queue
- Minimum capacity of the ITS to ensure zero queue
- Minimum capacity of the SPS to ensure zero queue

3.2.3. Acknowledgements and Response Time

- Percentage of times the NCC fails to send response to originator of specific schedule request within one (1) minute of receipt of request.

3.3. OPERATIONAL SCENARIOS

o   What is the effect on the NCC upon losing a particular SN service (SA or MA antenna(s))?
    [Actual loss of the service will no be modeled, however, an estimation of the arrival distribution
       of the SDRs and SARs due to the loss will need to be identified]
o   What is the effect on the NCC of adding new users?
[An increase in the number (and arrival rate) of all message types will need to be determined using
       the NPAS mission and extrapolation
o   What will be the effect on NCC's performance of changing from WSGT/NGT to STGT
[The FIMS and NSS messages will be removed and the number of bursts of SHOs sent will be
       reduced but the size (number of SHOs) of the bursts be larger]
o   What will be the effect on NCC's performance of adding another TDRS?
[Same as adding new users ///]
o   What will be the effect on NCC's performance from installing faster computers at the CCS, ITS,
    and SPS?
[Change the service rates for the respective processors, and observe changes in the NCC's
       performance]
o   What will be the effect on NCC's performance from incorporating improved scheduling
    algorithms (more flexible gereric request parameters, better conflict resolution, more
    automatiom)?
o   What will be the effect on the NCC's performance from incorporating improved priority
    schemes?
o   In the current configuration, at what point will the NCC become saturated?

4. SCOPE OF MODEL

In this version of the simulation model of the NCC the objects that represent the NCC internal processors
are being enhanced. The processors are the CCS, SPS, and ITS. These objects will now include
representations of the major processes that run on these computers. Each processor, (CCS, ITS or SPS),
will contain a single server that will act as the centural processing unit (CPU). The CPU will be
responsible for servicing the processes of these hosts on a priority basis. The CCS processes modeled are
the interfaces to both the ISN and NFE, the high-speed-message-exchange, the network monitor, and the
event monitor. On the SPS, the active and forecast schedulers, and the acquisiton processes are
represented. The display request process is the only process of the ITS that is modeled.

NCC network dialogues are also implemented. network dialogues are any specific set of messages that are
necessary to complete a task. Dialogues take place between external segment processors and one or more
of the NCCDS. An example of a dialogue is the schedule dissemination. During schedule dissemination
to POCCs the *active scheduler* or the *forecast scheduler* process will initiate the dialogue with a 9401, or
a 9402, or a 9403. The POCC will respond with an 0360 acknowledgement. The following message
dialogues between the NCC and the external segments have been implemented.

Message Dialogues:

        Ground Control Message Requests
        Schedule Dessemination
        Performance Data Dessemination
        Return Channel Time Delay Measurement
        Schedule Services
        Acquisition failure Notification

The arrival statistics of all messages will be determined by the output of the study performed by Stanford - ————. The dialogues will be initated by setting up statistical distributions to trigger the first message in the dialogue sequence. The means and variances of these distributions will be determined by Eric Richmond.

5. SYSTEM ABSTRACTION AND MODEL DESCRIPTION

5.1. System Abstraction

The system abstraction and model description that was done in the Phase I model was modified to eliminate the representation of the NFE. The NFE will be synthesized into the interface between the external segments and the NCC. This change will not affect the implementation accuracy of the model since the actions of the NFE is not considered in the problem analysis. These model abstraction changes will reduce the global view of the network to two distinct set of objects. The objects are classified as the *external segment objects* and the *internal segment objects*. A significant reduction in the runtime performance of the model is also expected.
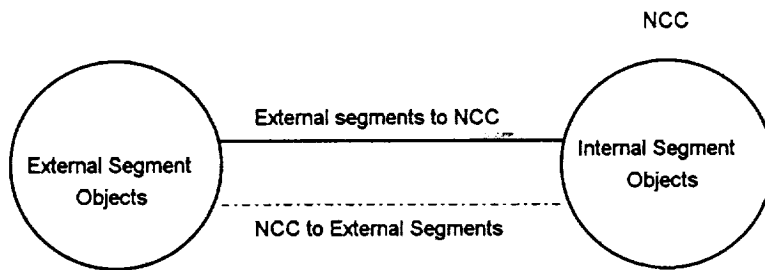

Figure 1:   Model Abstraction: Network View.

The inter-relationships between the segments of the system as seen from a performance point of view is directly related to the message traffic generated by each object. This approach to abstraction will allow the objects to be represented as processes and generators and the message traffic will characterize the work done in the model.

The *external segment* is mainly concerned with the generation of messages that are sent by the external processors to the NCC. A message generator objects will be use to generate messages to drive the simulation model based on data collected from NCC log tapes. The model will be required to handle the generation of messages for an eight hour shift. To ensure that the rate at which messages are generated closely matches the actual message traffic pattern in the real system, stochastic distributions will be attached to each message during each shift. The arrival rates of each message type/class will be computed from the sample data collected. The arrival rates will be used as input into the model as parameters of the stochastic distribution function used to determine the times at which messages will be generated.

The *internal segment objects* represent the NCC. The model of the NCC is made-up of three fixed communication nodes that represent the CCS, ITS, and SPS, see Figure 2. Each of these fixed communication nodes has incorporated into it, a unique node object. This was necessary because of the functional difference between the CCS, ITS and SPS. The internal segment objects are externally linked to the external segment objects via the CCS by two message streams. The SPS and ITS are linked to the CCS by the ISN which is a fiber based ethernet LAN. In this phase of the modeling effort, the ISN is represented by a virtual ethernet backbone model. This virtual backbone is a media access protocol developed in the form of an OPNET process model. The virtual backbone does not require a physical bus implementation. It however, has all the attributes of a bus except the collisions associated with this type of CSMA/CD protocol. All processors has the ability to initate and process message dialogues.

5.2. Model Description

The model is represented by two sets of objects connected by two uni-directional streams. The objects are representations of the *external segment* and the *internal segment*. The streams act as an interface between the external segments and the NCC. The *external segment* object will house the all the sources and sinks that are external to the NCC. The *external segments* represented are the FDF, NASCOM, NGT, POCCs, SDPF, and WSGT. The NCC houses the CCS, ITS, and SPS. The CCS, ITS, and SPS contain servers that are used to process incoming messages and generators for message creation. The *external segment* objects contain one or more message generators and a sink. Each generator will be linked to an object called the segment queue that is used as an interface between the CCS and the *external segments*. The link between the generator and the segment queue is a uni-directional stream. Uni-directional streams also link the segment queue to the sink process of the external segment object. All process timing is controlled by the generation of interrupts.

5.2.1. Server Process

The server processes performs all the required message processing within any processor. This object receives messages from any number of sources. The message are held for a simulated duration that is equal to the *service time* specified for that message. The message is then forwarded to its destination module. The forwarding algorithm uses the message type_class as an index to determine where to send the message.

The possible message sources are streams and sub-queues that may be part of the server object. The server polls the sources, bases on a predetermined priority, to access messages waiting to be processed. When a message is found it is given its simulated service time then removed from the source to a destination specified by the message path field in the message.

5.2.2. Generator Process

The generator processes are process objects that are responsible for the generation of messages within the system. The generoator uses a message traffic definition record of the following format:

type_class:	The message type/class
path	Message path through the processors within the NCC
mean	Message traffic mean
variance	Message traffic variance
destination	Message traffic destination
distribution_type	Message generation distribution type

The generator uses the *mean* and *variance* to generate messages of that specific type/class based on the outcomes of the distribution. The destination is the final destination of the message. The path is the route through the sub-processes of the CCS, ITS, and SPS that the message will take.

5.2.3. Link Facility

Connectivity between the objects is accomplished through point-to-point links and a virtual bus. The bus exist between the CCS, SPS, and ITS. The bus represent the ISN. All other objects are connected by uni-directional point-to-point links.

5.2.4. Interrupt Facility

The heartbeat of the model is based on four events. The first occurs when the time to generate a new message arrives. The inter-arrival times of these events are characterized by the parameters of the distribution function assigned to that message type. All message generators contain distribution functions that generate interrupts which initiate the creation of a message. The second type of events are the class of stream interrupts that are generated when a message arrives at an object. This type of event is automatically generated whenever a message sent from one object to another arrives at its destination. The third event is a service completion interrupt. This event is a server generated interrupt and it signals the completion of simulated service to a message. All post message servicing activities are initiated by this event. The final event is the dialogue response event. Messages that are a part of a dialogue sequence will either be destroyed or trigger another message. The receiving processor will determine if the message received needs a response or is triggering message. If either case is true the creation of the response message will be scheduled with the generator.

5.3. Model Work Characterization

A measure of the work done by the system is obtained by summing the simulated processing time for each message by each processor. NCC host processor perform pre-determined operations on messages by allocating each message processing time on its server. These operations are associated with logging, monitoring, routing, and scheduling. The simulation of these operations is accomplished by holding the message in a queue representing the process for some specified time. The work done by any processor can be computed as a function of the total time spent servicing messages. Each host processor is also equipped with an

5.3.1. Internal Processes

The processes identified in the CCS are the NFE and ISN interfaces, the high speed message exchange, the logger, the network monitor and the event monitor. All messages entering the CCS travel to the high speed message exchange and is simultaneously logged by the logger. The forwarding algorithm uses the path field of the message to determine the path the message takes through the processor. The ISN provides the interface between the CCS and the ITS and the SPS.

The SPS consists of four distinct processes. These are the acquisition process, the SAR router, the active schedule process and the forecast schedule process. Messages entering the SPS are routed to the acquisition process or the SAR router. From the SAR router the messages are passed to other SPS processes.

Messages to the ITS are destroyed. Those leaving the ITS are sent to the ISN were they are routed to their respective destinations.

5.3.2. Internal Message Paths

Messages generated by the external processors first enter the CCS internal processor. Messages entering the internal processors follow specific paths to their final destination. Messages generated by the internal processors passes through the CCS as they travel to the intended external processor. Combination of the possible routes of the messages through each internal processsor led to the identification of distinct paths.

5.3.2.1. CCS MESSAGE PATHS

Messages entering the CCS are created by the the external segment processors, the ITS and SPS or the network and event monitors within the CCS. All messages entering the CCS go to the logger and the high speed message exchange. Messages to the logger are sent to a sink. Those to the high speed message exchange are routed to either the ISN, the network monitor or the event monitor. Messages to the ISN proceed to the SPS or the ITS for processing. Messages sent to the network monitor initiates dynamic display messages before they are destroyed. These dynamic displays are routed to the ISN to be displayed on the ITS. Messages sent to the event monitor are the GCMR's. These initiates GCM messages before being destroyed.

Messages generated by the ITS are sent to the CCS via the ISN. The ISN broadcasts these messages to the network monitor where they initiate the generation of other messages before being destroyed. The newly generated messages are sent to the high speed message exchange where they are routed to the respective external generators.

Messages generated by the SPS are also sent to the CCS via the ISN. The ISN broadcasts these messages to the high speed message exchange where they are routed to the respective external processors.

Seven distinct paths were identified for the CCS processor. These paths and messages traversing them are described below:
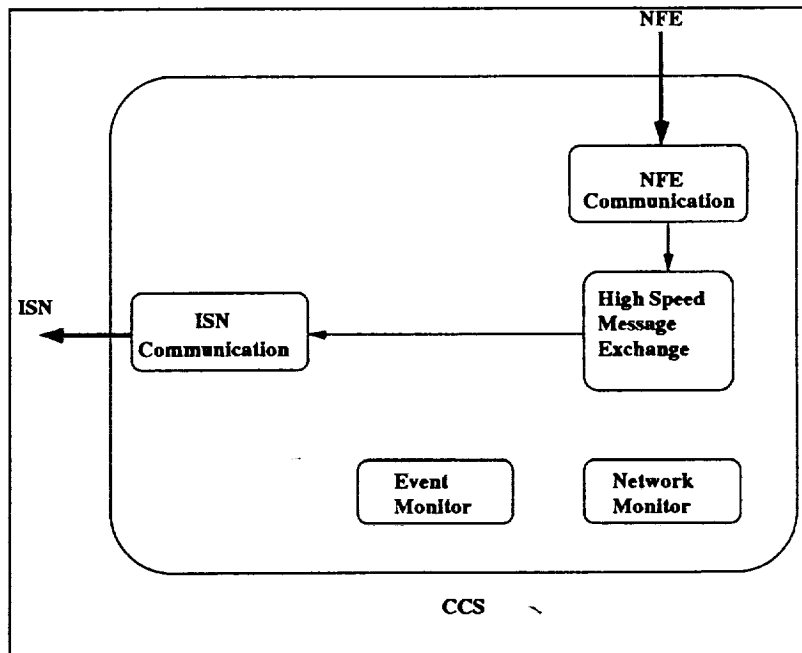
PATH 1   NFE -> high speed message exchange -> ISN

PATH 2   NFE -> high speed message exchange -> network monitor -> ISN
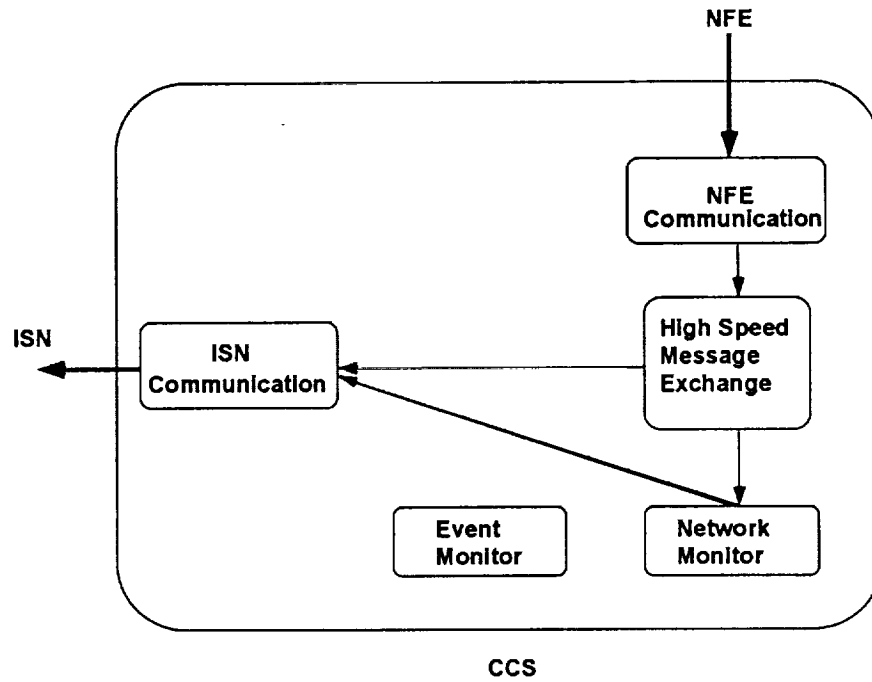
PATH 3   SPS -> ISN -> high speed message exchange -> NFE

PATH 4   ITS -> ISN -> network monitor -> high speed message exchange -> NFE

PATH 5   NFE -> high speed message exchange -> ISN

PATH 6   NFE -> high speed message exchange -> event monitor

PATH 7   Event monitor -> high speed message exchange -> NFE

5.3.2.2. SPS MESSAGE PATHS

SPS PATH

Messages through the SPS can be generated by a SPS process or can be received via the ISN from the CCS. Messages received from the ISN travelled through the CCS according to path 1. As a result the path of all messages entering the SPS is defined as path 1. These messages are routed to the acquisition process or the SAR router.

The acquisition process routes the messages it receives back to the ISN to be broadcast to the CCS. The SAR router routes the messages it receives to either the active schedule process or the forecast schedule process. The messages to these two processes from the SAR router initiates the generation of other messages. These generated messages are sent to the SAR router where they travel to the ISN to be routed to their various destinations.

Messages leaving the SPS processor enter the CCS via the ISN. These messages travel through the CCS in the path defined as path 3. As a result the path of all messages leaving the SPS is also defined as path 3. This path is independent of the possible routes travelled by messages through the SPS.

There are three possible routes for messages entering the SPS. Descriptions of these routes and messages are given below:

Route 1) ISN -> Acquisition Process

Messages for this route are: 0309, 0310, 0315, 0351, 0353, 0361, 0362, and 0365. The names of these messages were given for the CCS paths.

Route 2) ISN -> SAR Router -> Active Schedule Process

Messages for this route are: 8651, 9910 and 9911. The names of these messages were given for the CCS paths.

Route 3) ISN -> SAR Router -> Forecast Schedule Process

Message for this route is 9910.

Three routes also exists for messages leaving the SPS. Description of these routes are given below:

5.3.2.3. ITS MESSAGE PATHS

All messages to the ITS are destroyed at the sink. Display requests generated at the ITS are sent to the ISN where they are routed to their various destinations. The architecture for messages travelling through the ITS is shown in figure .

6. SIMULATION PROGRAM

The simulation program is an OPNET implemention of the abstracted model, that uses discrete event simulation in a network of queues to represent the NCC and its external segments. The model is made up of fixed communication nodes connected by a virtual bus and point-to-point links. The virtual bus is a network bus implementation that does not allow collisions. This design decision was necessary to reduce the model runtime. Each node acts as a message generating and message processing servers. The external nodes are made up of generator and sink processes. The internal nodes are single servers that service multiple subqueues within each node. These internal processors are represented by a combination of OPNET queues and processor models. The networks that connect the processors are represented by OPNET packet streams that are initiated and terminated by point-to-point transmitters and point-to-point receivers respectively. A complete set of model reports are provided in Appendix A.

6.1. Network Domain Description

The network domain is represented by the external subnet and the internal subnet. The external subnet is made up of one fixed communication node, that contain packet generators, connected to a subnet that represents the NCC. The NCC subnet contains fixed communication nodes that represent the CCS, ITS, and SPS. These internal processors are connected by ISN which is represented by a virtual bus implementation that is collision free. Packets are moved from the external subnet to the internal subnet and back via two packet streams. The network level model is shown in Figure 6.1.
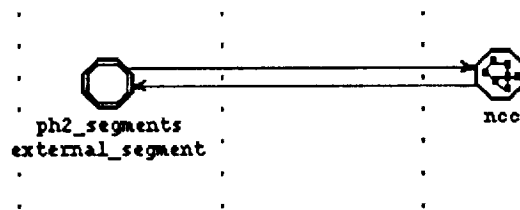

Figure 6.1: NCC Model - Network Level

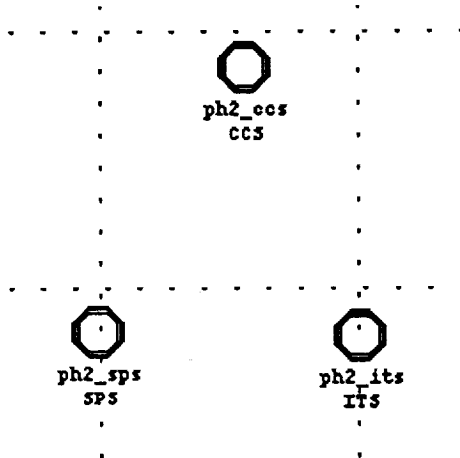A view of the NCC subnet is shown in Figure 6.2.

Figure 6.2:   NCC Internal Subnet

6.2. Node Domain Description

The node model domain, like the network domain contain node models that are associated with both the internal and external segments.   The node models are the representation of server, sink, generation and communication elements. In the external segment node model attributes are used to make each segment (node) unique.   This is to facilitate creating one generic process model for all seven segments.   The node model attributes will be read by using the OPNET kernel functions.   The additional attributes are:

NAME	TYPE
number of messages	integer
station address	integer
message info file	data file

The "message info file" will contain information necessary for generating messages for a given distribution. The details of the file format will be explained in the process models discription.

6.2.1. External Subnet Node Models

Each external segment node is a combination of a message generator and a message sink. The message generator sends packets via a stream to the external segment queue. The function of this queue is to collect messages from the external segment message generators and pass them on to the CCS within the NCC. The segment queue is used only as a funnel for messages and is not an abstraction of any subsystem being modeled. The External Segment node models are shown in Figure 6.3.
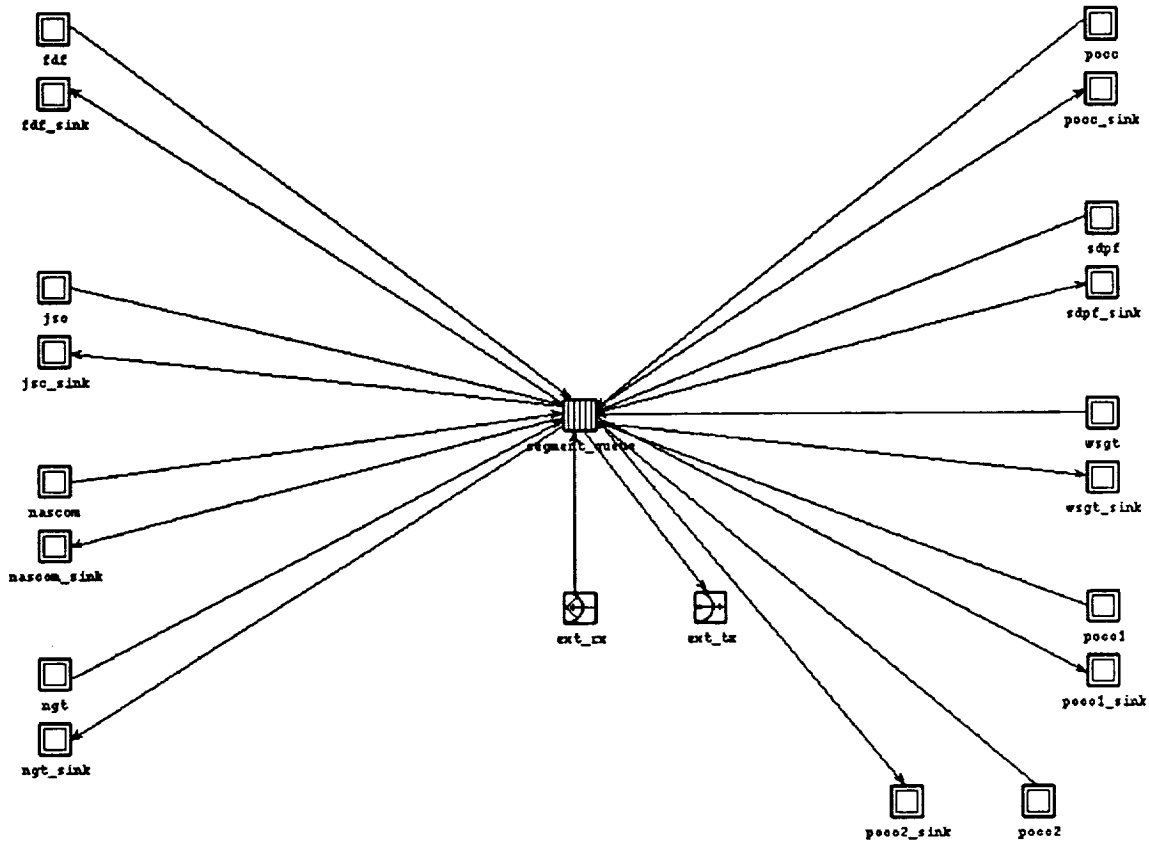

Figure 6.3: External Segment Node Models

6.2.2. Internal Subnet Node Models

Three primary node models make up the internal processors. These are the ph2_ccs, ph2_its, and ph2_sps. The ph2_ccs represent the CCS and is the interface to the external segments. Communication with the external segment objects is achieved through the use a transmitter, ext_int_tx, and a receiver, ext_int_rx. The at the heart of this object is a server process, server_router, in which all sub-processes are represented as sub-queues. Two generators are present in this node, and are used to generate messages for the event monitor and the network monitor. The isn is the implementation of the virtual bus used in communication with the ITS and SPS. A sink is provided to destroy messages that exit the system. Figure 6.4 show the CCS node model. All communications between the elements of this model is by packet streams.
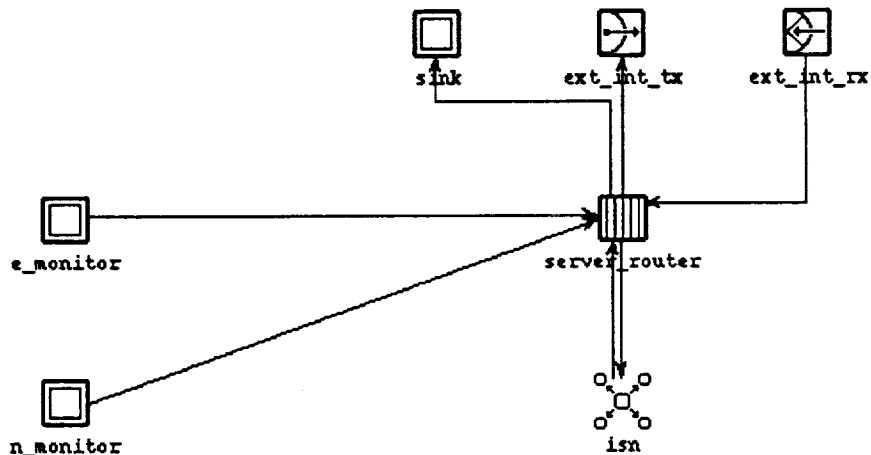

Figure 6.4: CCS Node Model

The SPS node model uses an architecture similar to that used by the CCS. In the SPS node model a server element that represents the CPU of the SPS is fed packets from an implementation of the Acquisition Process, the Active Scheduler and the forecast Scheduler. Messages arriving from the ISN are also accepted. The messages entering the SPS are processed in accordance with the rules that govern the paths taken by specific messages through the SPS. These rules are defined in the section labelled SPS MESSAGE PATHS. Figure 6.5 show the elements that make up the SPS Node Model.
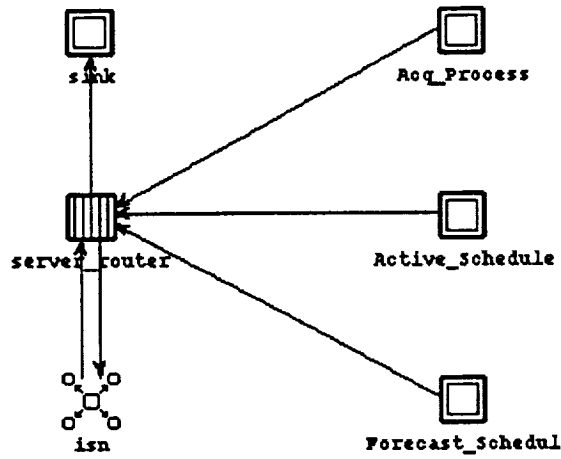
The ITS messages like the SPS and CCS messages are governed by the definitions given in the section named ITS MESSAGE PATHS. The ITS is a server based model that transmits display requests the Network Monitor located in the CCS. accepts input messages from

6.2.2.1. PROCESS DOMAIN DESCRIPTION

The models on this domain are used to describe the behavior of the processor and queue models within the node model domain. The process models used, are of two basic types, communicating processes, and server processes. The communicating processes are logical entities that interact to accomplish some common goal. The server processes are the entities that processes the messages in an effort to impose the notion of work being done within the system being modeled. The logic of the process is specified in FSM's. A FSM may be generally defined as an automaton which has states, inputs, and outputs. The FSM models its process by responding to changes in its inputs, modifying its state, and producing new outputs. In OPNET the primitives for building process models are states and transitions. OPNET employs the concept of event scheduling and interrupts, in which each incident is called a *simulation event* and an *interrupt* represents the actual execution of the scheduled event. See Appendix A for the process models used in this model.

7. MODEL VERIFICATION AND VALIDATION

7.1. Model Performance Characteristics

7.2. Paper Model

8. OUTPUT ANALYSIS

9. APPENDIX A: MODEL REPORTS

10. REFERENCES