# NAVAL POSTGRADUATE SCHOOL
## MONTEREY, CALIFORNIA

# THESIS

IMPROVEMENTS TO A SURGE AND
SUSTAINMENT MODEL FOR WARGAMING

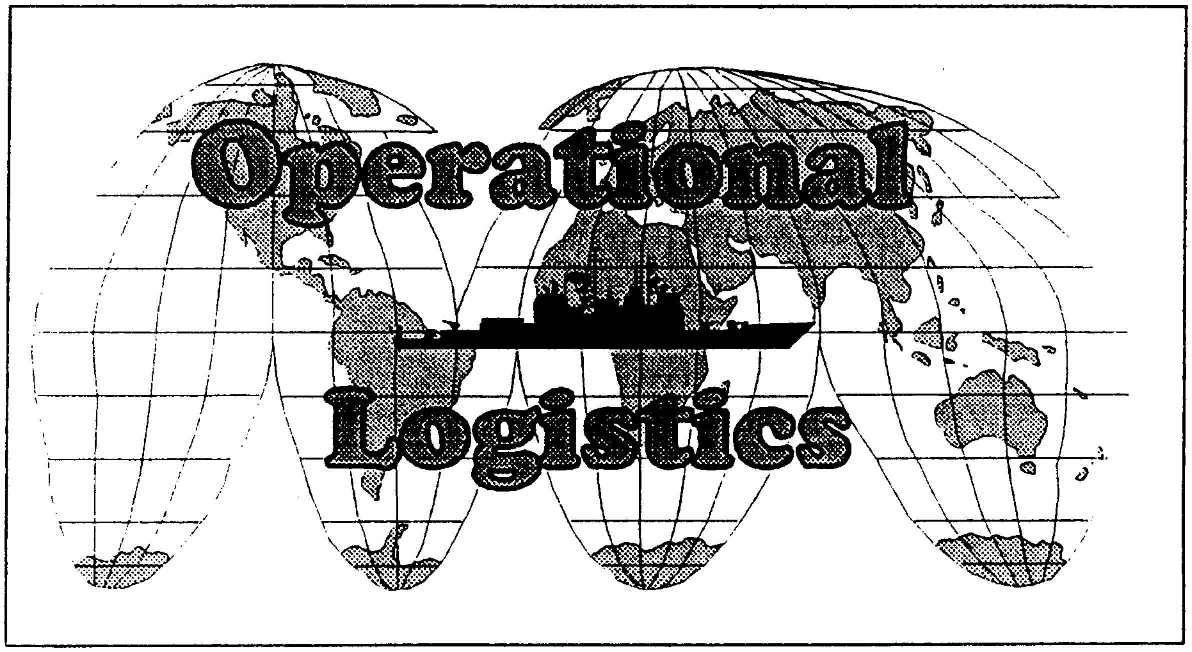
by

Thomas G. Halvorson

September 1994

Thesis Advisor: David A. Schrady

Approved for public release; distribution is unlimited.


Operational
Logistics

|   |   |
|---|---|
| **REPORT DOCUMENTATION PAGE** | Form Approved<br>OMB No. 0704-0188 |

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

|   |   |   |
|---|---|---|
| **1. AGENCY USE ONLY** (Leave Blank) | **2. REPORT DATE**<br>September 1994 | **3. REPORT TYPE AND DATES COVERED**<br>Master's Thesis |

|   |   |
|---|---|
| **4. TITLE AND SUBTITLE**<br>IMPROVEMENTS TO A SURGE AND<br>SUSTAINMENT MODEL FOR WARGAMING | **5. FUNDING NUMBERS** |
| **6. AUTHOR(S)**<br>Halvorson, Thomas G. |   |
| **7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**<br>Naval Postgraduate School<br>Monterey, CA 93943-5000 | **8. PERFORMING ORGANIZATION REPORT NUMBER** |
| **9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)** | **10. SPONSORING / MONITORING AGENCY REPORT NUMBER** |

**11. SUPPLEMENTARY NOTES**

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

|   |   |
|---|---|
| **12a. DISTRIBUTION / AVAILABILITY STATEMENT**<br>Approved for public release, distribution is unlimited | **12b. DISTRIBUTION CODE**<br>A |

**13. ABSTRACT** (Maximum 200 words)

This thesis documents the design, validation and demonstration of the improved Surge and Sustainment Simulation for wargaming. The model is an object orientated, discrete event simulation written in the Modsim II programming language. The objective is to improve upon an already existing model through logic refinement, and the addition of objects designed to enhance user access to logistics system performance information. The finished product is an interactive simulation model designed to be used as a tool at Service wargaming facilities. Most notable is the model's ability to interact with the user in time steps versus only once at the beginning like other models currently being used in the area of force surge and sustainment.

|   |   |
|---|---|
| **14. SUBJECT TERMS** Object-orientated, Logistics, Wargaming, Surge and Sustainment | **15. NUMBER OF PAGES**<br>106 |
|   | **16. PRICE CODE** |

|   |   |   |   |
|---|---|---|---|
| **17. SECURITY CLASSIFICATION OF REPORT**<br>Unclassified | **18. SECURITY CLASSIFICATION OF THIS PAGE**<br>Unclassified | **19. SECURITY CLASSIFICATION OF ABSTRACT**<br>Unclassified | **20. LIMITATION OF ABSTRACT**<br>UL |

Approved for public release; distribution is unlimited

# IMPROVEMENTS TO A SURGE AND SUSTAINMENT MODEL FOR WARGAMING

by

Thomas G. Halvorson
Lieutenant, United States Navy
B.A., Northwestern University, 1988

Submitted in partial fulfillment of
requirements for the degree of

**MASTER OF SCIENCE IN OPERATIONS RESEARCH**

from the

**NAVAL POSTGRADUATE SCHOOL**
**September 1994**

Author: ______________________________
Thomas G. Halvorson

Approved by: ______________________________
David A. Schrady, Thesis Advisor

______________________________
Thomas E. Halwachs, Second Reader

______________________________
Peter Purdue, Chairman,
Department of Operations Analysis

# ABSTRACT

This thesis documents the design, validation and demonstration of an improved Surge and Sustainment Simulation for wargaming. The model is an object orientated, discrete event simulation written in the Modsim II programming language. The objective is to improve upon an already existing model through logic refinement, and the addition of objects designed to enhance user access to logistics system performance information. The finished product is an interactive simulation model designed to be used as a tool at Service wargaming facilities. Most notable is the model's ability to interact with the user in time steps versus only once at the beginning like other models currently being used in the area of force surge and sustainment.

# THESIS DISCLAIMER

The reader is cautioned that the computer code developed for this thesis has been completed within a limited period of time and has not been fully tested. Although a strong effort was made to ensure the correctness of the programming, the code still needs to be verified and validated. Any results or conclusions brought forth from the use of this program are done so at the user's risk.

## TABLE OF CONTENTS

# EXECUTIVE SUMMARY

The Surge and Sustainment Simulation was developed for the Naval War College in December of 1993. The first version, Version 1.0, did not completely fill the original programming requirements. Additional modifications were needed in two primary areas, program logic and information accessability. The subject of this thesis is an update to the Surge and Sustainment Simulation code to alleviate these problems.

The Surge and Sustainment Simulation is an object orientated discrete event computer program written in Modsim II. It was designed to provide logistics "truth" for both seminar and computer wargaming by modeling the logistics infrastructure for wargame scenarios.

The model operates a logistics network made up of nodes and arcs through which commodities flow. Network nodes represent bases and units. Each node maintains an inventory of commodities. The network arcs which connect these nodes allow movement of cargo and are modeled as transporters. The S3 user develops a specific logistics network by creating nodes, arcs, and commodities that reflect those used in the wargame scenario of interest.

During simulation a scenario timeline is followed using discrete time steps. Units begin the simulation as commodities which need to be surged into the scenario theater. Upon deployment and activation these units begin to consume commodities and pull additional supplies into theater The ability of the logistics network to provide the

transportation required for this surge and sustainment is reflected in the long run status of unit inventories. These inventories in turn can be used as constraints for a concurrently run computer or seminar wargame.

The first version of the Surge and Sustainment Model brought this logistics network to life. It did so despite some logic problems and with little user access to information about the network performance. Version 2.0 of the Surge and Sustainment Simulation has been created to deal with these shortcomings. Several important improvements in the usability and validity of the model are incorporated in its design. An orders tracking system now provides intransit visibility for all supplies inbound to the theater of operations. A diagnostics module gives the user instant access to information about network performance for node ports, transporters, and overall closure of forces. Several modifications in the simulation logic have been engineered. Surface ship transporters travel the oceans using waypoints such as the Straits of Gibraltar rather than simply taking the shortest path between two nodes. Unit nodes are no longer stationary allowing the modeling of multiple theater scenarios in which units move between theaters. Numerous smaller improvements have also been made to increase model fidelity and in debugging the Version 1.0 computer code.

A demonstration of the the Surge and Sustainment Simulation's capabilities is given using a scenario with two nearly simultaneous conflicts. The demonstration is also given as a user's guide, providing the menu steps required to operate the model.

# I. INTRODUCTION

## A. AN OVERVIEW OF NAVAL WARGAMING

The United States Navy maintains a strong interest in wargaming as a tool for analyzing strategy, investment, and policy. Wargaming is an artificial representation of actual warfare. Clausewitz called war an art[Ref. 1]. Wargaming then is the paint by numbers used to groom military artists for the real canvas. Much in the same way that practice teaches the painter to choose the best brushes and paints in the art world, wargaming is meant to assist the military artists in choosing the right forces, and the best way to employ those forces in war.

Although wargames are designed to produce realistic decision making environments and to approximate the events of war no wargame can do this perfectly. Many assumptions are made in their design to ensure both timely operation and and to prevent an unreasonable level of complexity. Therefore wargames give only approximate answers and any assumptions must be scrutinized and well understood before wargame results are taken to be valid. It cannot be overstated that wargaming is merely educated guessing, the real solutions to problems put before wargamers can only be found in actual combat.

Various methods of wargaming exist in the Department of Defense. These include seminar style wargames, computer simulations, and actual non-combat field manuevers. Neither one is a substitute for the other, but they all focus on the same goal, helping make the correct decisions about structuring the military forces and how to engage those forces to secure the objectives of the national military strategy. Each method could be considered a separate decision aid in the wargaming family.

The U.S. Navy uses all three methods of wargaming, but for the purpose of this thesis only those used at the Naval War College are of interest. Seminar and computer wargames are the primary wargaming tools for these games. Since 1887, United States Naval officers have been discussing strategy and tactics at the Naval War College in Newport, Rhode Island. Today, nearly fifty wargames a year are hosted, averaging from one to two weeks in length.

The majority of these wargames are seminar wargames. In this method the timeline and events for the game scenario are preset. Players are faced with determining if that timeline is realistic. The game timeline is typically broken into phases, such as force deployment, the campaign battles, and redeployment. Each of these phases is examined seperately by the game players and usually played as a single time step.

The primary goal of seminar wargames is to single out weaknesses in investment, strategy, or policy decisions. If players believe the game timeline can't be met and/or the scheduled events are not feasible, the perceived fragile points are singled out. These areas are then reexamined and ranked in significance with each other. Some of these can be show-stoppers that game players believe will result in operation failure, while others may only be the relative weak points in a winning strategy.

The reasoning used by players in making these problem determinations can come from several sources. Military expertise, quantitative analysis, computer models and simple technical facts are good examples. Often-times several groups will have differing opinions on what the most important problems are and how the events of a particular game will play out. For this purpose there is a game umpire or umpires. Usually, the umpire is one of the senior officers present. The umpire's job is to make the final decisions over the interpretation of events and the weighting of scenario weaknesses.

Most seminar wargames last about a week, but the game preparations can take several months. First, the scenario must be designed and verified. Next, the game participants must analyze the timeline and events, and ensure that they collect all the information they need to represent their area of expertise during game play.

The second style of wargaming used in Newport is computer wargaming. The Enhanced Naval Wargame System (ENWGS) is the computer wargame used by the Naval War College. ENWGS is a multi sided game, with several groups interacting in one scenario. In ENWGS the events and/or timelines of force movement are not predetermined. The parties involved in the game are given beginning force postures, missions and policies to carry out during play. The subsequent actions of these forces are subject to human decision making. Each team is placed in its own individual room or cell. From this room they are able to send commands to their forces through a computer interface.

As in real war, the overall big picture is not available to any one side in the game. They know only what they see in their cell during game play. The big picture is available only to the umpires in the control area or game floor. There they have complete control of the computer and thus the wargame. The outcome of actions between the forces commanded by individual cells can either be left completely to the computers computations or altered to reflect the umpires wishes. Typically a game scenario is played until precontrived conditions are met by one or more cells and a decision as to the game outcome can be made.

Computer wargaming is especially good at testing tactics of force employment and fostering the skills needed to command those forces. It is not as good in determining whether forces will be victorious. Often the modeling of attrition is an inexact process, with many factors only roughly modeled. Ingredients such as the enemies tactics and will to fight are hazy guesses made from just as hazy intelligence reports.

## B. THE NEED FOR THE SURGE AND SUSTAINMENT SIMULATION

Today the world of wargaming finds itself changing as quickly as ever. The number of possible conflict sites considered is growing and the forces available to complete these missions are shrinking. Both of these factors put pressure on force planners to make the most informed decisions possible about how to shape and fight tomorrow's military with little room for error. One way for them to do this is to improve the quality of information that is used in making those decisions. In wargaming, the way to improve accuracy is to increase the fidelity in modeling.

An area of naval wargaming that invites improvement is logistics. Both seminar wargaming and computer wargaming at the Naval War College have logistics deficiencies. Neither addresses campaign level logistics satisfactorily for wargame play. A stronger representation of logistics at Newport will give more truthful insights to the many important questions confronting analysts today. Both seminar and computer wargaming have distinct requirements for improvement.

In seminar wargaming there is a need for a flexible, on-site logistics model. Currently the only logistics data available to game players is either precontrived or a product of rigid non-interactive logistics models. Due to the nature of seminar wargaming, not all problems are anticipated and an on-site model is needed to look into "on the spot" logistics questions. The model must be interactive and flexible, to ensure all sides of a question can be looked into. In order to accomplish this it must be easy for the user to reconfigure the model scenario and the user should be able to time step through the game.

Simple "what if " questions concerning things like delayed port openings, or shipping losses go beyond the capabilities of the rigid logistics models in use today. A logistics model designed with the above requirements in mind will not only improve logistics analysis but analysis in general at seminar wargames.

ENWGS also has a weakness in the area of logistics. Individual unit fuel and ammunition levels are the only logistics monitored in ENWGS. Once a unit runs out of either quantity there is no effect other than the umpire magically resupplying it. Therefore currently ENWGS has an infinite supply of ammunition and fuel, the only logistics constraint on forces being the whims of the game umpire. The real world is not this way. As wargaming is an attempt to model the real world, the modeling of logistics in ENWGS then falls severely short of this goal. The introduction of campaign level logistics constraints is needed in ENWGS to better reflect the true capabilities of forces.

An improvement in logistics modeling is needed for seminar and computer wargaming at the Naval War College. A single computer simulation fits the requirements for both. Lieutenant John Long of the Naval Postgraduate School began the Surge and Sustainment Simulation (S3) in 1993[Ref. 2]. This logistics model is a step in the right direction for naval wargaming in Newport. The logistics problems currently faced using ENWGS or in seminar wargame can both be handled using the methodology found in S3.

The output of S3 is designed to produce the answers to the questions of surge and sustainment; for instance, closure times and unit sustainment levels. For ENWGS this simulation offers theater level logistics. The logistics picture for each unit can be determined at any point in the simulation run allowing logistics to play a part in the smallest of engagements. Post game closure data is also produced to show the sustainment provided by the game's logistics system. For seminar wargaming S3 offers flexibility and human interaction throughout play for exploring all of the game "what ifs", it also provides closure data for insight into strategic mobility questions.

The Surge and Sustainment Simulation was designed from the ground up to fill this role. This thesis is part of the ongoing effort to ensure S3 meets that goal. In Chapter II, the original version of S3,Version 1.0, will be reintroduced and its capabilities summarized. Next, in Chapter III, the additions to S3 incorporated in Version 2.0 will be described. Chapter IV presents S3 (Version 2.0) with a demonstration walk through of a typical wargame scenario. In the final chapter conclusions about the progress made with S3(Version 2.0) and recommendations for Version 3.0 will be offered.

# II. AN INTRODUCTION TO S3 (VERSION 1.0)

## A. BACKGROUND

The Surge and Sustainment Simulation (Version 1.0) was completed in December of 1993. This discrete-event simulation is an interactive, object-orientated program built using MODSIM II, a C based simulation language, and run on a Unix based workstation. An overview of the model follows.

The Surge and Sustainment Simulation based upon a simple logistics network. This network consists of nodes and arcs through which commodities pass. Four basic events define the commodity flow-cycle in this network; production, storage, movement, and consumption. Nodes are the storage points for commodities and arcs are the means by which those commodities transit from node to node. In addition, special nodes act as gateways in and out of the network. These nodes model production and consumption or birth and death of commodities. Figure 2.1 below gives a graphic representation of the network idea.

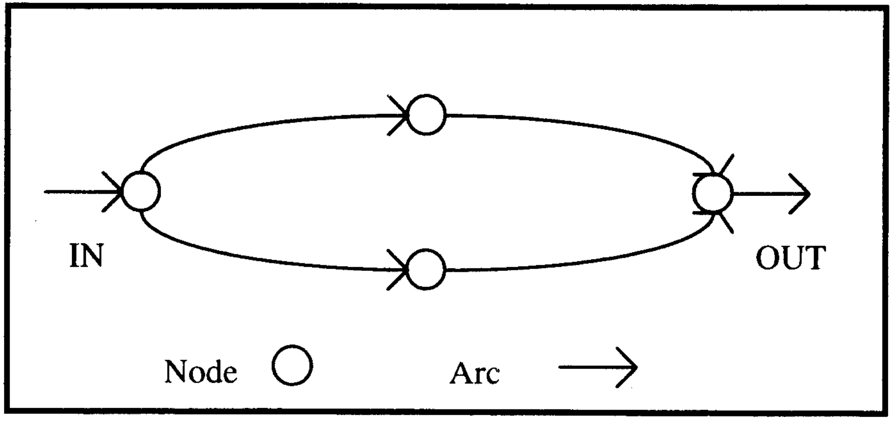


**Figure 2.1 A Generic Network**

This commodities (or logistics) network is the heart of S3. The commodities, nodes, and arcs that form the core elements of this system are the first principles which need to be understood before any other aspects of the model are discussed. These building blocks are represented in MODSIM as objects.

> Objects are dynamically allocated data structures coupled with routines, called methods. The fields in the objects data structure define its state at any instant in time while its methods describe the actions which the object can perform. [Ref. 3:p 101]

For instance, a node in the logistics network above could be represented as an object. It has data fields that hold information concerning the quantity of commodities currently being stored, it also has methods to effect the events that guide the entry or departure of those commodities to or from storage.

In summary, the Surge and Sustainment Simulation is based upon a simple network of nodes and arcs. Commodities travel through this network along arcs moving between many origin and destination nodes. Certain special nodes allow these commodities to enter or exit the network. The commodites, nodes, and arcs of this network are represented as objects in the MODSIM simulation language. Any of the objects in S3 can be boiled down to the part it plays in the life of the logistics network at the core of S3. The output of this simulation is the measurement of the surge and sustainment of unit supplies provided by a user defined and operated network.

## B. S3 BUILDING BLOCKS

The logistics network in S3 is built using six building blocks. All of these are represented as objects in Modsim. They are commodities, transporters, subunits, ports, bases, and units. Each of these building blocks fall into one of three categories from the logistics network: node, arc, or commodity. Commodities make up the lifeblood of S3, flowing through the network when the model is in operation. The network arcs are represented by transporters and the nodes of the network are bases and units. Both bases and units have lower level building blocks called ports. Furthermore, units are built of several smaller elements called subunits.

Reviewing the four basic events of S3, we can see that all of the above building blocks play an active role in the logistics network. Transporters are the arcs of S3, moving commodities through the network. Bases and units store commodites, and use ports to transfer the commodites between transporters and storage. Units, made from subunits, are the consumer nodes of the network, and the supply base, is the sole production node. As a sidenote, follow on work to this thesis will explore altering this network to allow additional production nodes modeling host nation support. Figure 2-2 below shows the relationship between these building blocks and the logistics network.

| S3 Objects | Node | Arc | Commodity |
|---|---|---|---|
| Commodities | | | X |
| Transporters | | X | |
| SubUnits | X | | |
| Ports | X | | |
| Bases | X | | |
| Units | X | | X |

**Figure 2-2 Objects As Network Elements**

It is important to note the role of the unit in this network. The unit is really three things, it is the consumer, a node, and a commodity. The unit is the only object that consumes commodities, it also stores commodities waiting to be consumed, and it is itself a commodity during the surge portion of the simulation because it needs to be transported into theater. These three points will be mentioned again below in detail, but it is critical they are understood in order to comprehend the function of the unit in S3.

The following are descriptions of each of the six building blocks.

### 1. Commodities

Commodities are the simplest elements of S3. They do not perform any actions except to exist and flow through the logistics network until they are consumed by units. Each named commodity is defined by its class, dimensions, weight, production rate, and movement priority.

The commodity classes determine the type of transporter required for cargo movement and the priority of assignment to the same. The classes are:

Fuel : Petroleum, Oil , Lubricants and Water[1]
Subsistence: Food stuff of all kinds
Ammunition: Munitions of all kinds
Spares: Parts, maintenance items and consumables
Personnel
Medical
Major Equipment: Large equipment such as helicopters and tanks
Other Equipment

Note 1: This class should really be called liquids or Class I.

Dimensions and weight of commodities are given in inches and pounds respectfully. They are used to ensure the proper loading constraints of each transporter are met. Production rate refers to the industrial base daily production capacity for each particular commodity. The Supply base in the S3 logistics network uses this rate to create new commodities every twenty-four hour simulation period.

The last item, shipment priority is used to determine the transportation class used to move a particular commodity in the logistics network. The priority is an integer

**TABLE I** - COMMODITY PRIORITY OF SHIPMENT

| Priority | Ranking of Preferred Shipment |
|---|---|
| 1-3 | Air, Rail, Truck, Ship |
| 4-6 | Rail, Truck, Ship, Air |
| 7-9 | Truck, Ship, Rail, Air |
| 10-12 | Ship, Truck, Rail, Air |

from one to twelve. There are four different transportation selection schemes based upon this number. Within each of these schemes there are an additional three subrankings. The subranking give priority to commodities beyond the order in which transportation is selected. See Table I above for a complete listing of the priority schedule. As an example, a tank with priority four will be granted transportation ahead of a jeep with priority six, but both will search the list of available transporters in order of rail, truck, ship and finally air.

### 2. Transporters

Transporters are the arcs of the logistics network in S3. There are four major classes of transporters. These classes are Aircraft, Rail, Truck, and Ship. Each of these classes in turn has a subclass associated with it. The subclass defines the particular cargo that a transporter is able to carry. The subclasses are Liquid, RoRo, Breakbulk, General, and Pax.

A transporter has several performance charachteristics, chief among these are speed and cargo capacity. Transporters are either moving between bases, unloading at a port berth, or waiting for assignment at the last port entered. For example a commercial tanker could be modeled as a transporter in S3 as follows. Its class would be ship and its subclass liquid. The length may be 800 feet with a cargo capacity of six to nine million gallons and a max speed of twenty knots.

There are two special categories of transporters, prepositioned ships and ready reserve ships. Prepositioned ships are transporters that begin a simulation with a starting location, a destination and cargo. The assignment of the initial ship location, it's destination unit and cargo is done by the user in the scenario setup section, before the simulation is run. The prepositioned cargo is earmarked for one particular unit, the cargo cannot be rerouted after the simulation has begun. Prepositioned ships will stay at their initial locations until they are activated by the user and proceed to their destination. The ready reserve ships are transporters that, like the prepostioned ships, begin the simulation in a non-active status. Reserve ships are different than the prepositioned ships in three ways. First, they have no preassigned cargo; second, they have no preassigned destination; third, they are not available immediately after activation. The ready reserve ships fall into one of four categories of Reduced Operating Status(ROS); ROS-4, ROS-5, ROS-10, and ROS-20. The hiphenated number in each category indicates the number of days required to place that unit into operation. In S3 a ROS-4 ship told to activate by the user would wait four days before entering the network as an active transporter. These ships are critical to the surge portion of this model. Today's Ready Reserve Force (RRF) and Afloat Prepositioning Forces (APF), APF including both Maritime Prepositioning Ships(MPS) and Army Prepositioning Ships(APS), can be modeled using these special transporters.

### 3. Subunits

Subunits are the building blocks for units. Each subunit emulates a particular capital item, such as a truck or an M-1A1 tank, and the consumption information particular to that platform. Subunits do not perform any active role in the S3 logistics network, they are really individual records that simplify the unit building process. The sum total of all individual subunit consumption and inventory data within a unit gives that unit its total consumption and inventory information. For instance, an armored calvary regiment is represented as a unit comprised of several M-1A1 tank subunits. If each tank consumes five units of ammunition every twenty-four hours and has a fifty unit ammunition holding capacity, a unit with five tanks then consumes twenty-five units of ammunition per day and requires a total inventory of two hundred and fifty units of ammunition. Thus subunits store the information that enables units to act as the consumers in the S3 logistics network. Each subunit has a list of commodities in its inventory, the total unit inventory is derived from the sum of its subunit inventories.

### 4. Ports

Each base and unit has one or more of four transportation port types. These are truck stops, railyards, seaports and airports. For example, Corpus Christi, Texas could be represented as a base with a railyard, an airport and a seaport. Each port has a specified maximum capacity as well as limits on the size of transporters that may pass through it. For instance, Corpus Christi may only have two piers and limit the size of ships at those piers to five hundred feet. All loading and unloading of transporters takes place at port unloading spaces; ie. for ships at seaport berths. After transporters are unloaded they are removed from their berth assignments and placed in a parked position until further assignment is given.

### 5. Units

Units are the most important piece of the logistics network in S3. They, like bases, act as nodes in this network. Units are positioned with a latitude and longitude, they act as storage sites for commodities and have ports for the movement of commodities to and from transporter arcs. The properties mentioned so far are common to both units and bases, but they are not the same. Units differ because in the logistics network they are the sole exit nodes. Unit actions to this end produce the surge and sustainment flow of commodities that are the objective of the simulation. The way that units do this is by playing two distinct roles in S3. These roles are sequential and concurrent with the two phases of the simulation, surge and sustainment.

The first role of the unit occurs during the surge phase. When the simulation starts a unit can begin in one of two states, deployed or non-deployed. If a unit is not deployed it must be moved into theater before it can be activated. In this phase of S3 the unit acts as a commodity which must be transported. The unit begins the simulation as merely a latitude and longitude with nothing else. Upon activation the unit simply awaits delivery of itself from its home base or origin. The unit is activated by the user in one of two ways, either automatically or instantly upon demand. Automatic deployment occurs at a predetermined time that the user has entered when building a unit prior to execution of the scenario. Instant deployment can be ordered by the user interactively at any time in the simulation prior to automatic deployment.

Once being tasked to deploy, a unit will order itself, its entire required inventory including personnel, from its origin base. Upon transportation of that inventory to the unit it becomes active and enters the sustainment phase of S3. In the sustainment phase the unit shifts from being a commodity and begins performing a second role, the unit now becomes the consumer in the logistics network

Consumption is a continual process that takes place every twenty-four hour period and is based on the daily consumption factors for each individual commodity held in a unit inventory. These consumption factors reflect three different combat intensity levels; high, medium, and low. Each item in unit inventory is maintained at a particular stocking objective. This stock level is maintained by using mandatory ordering points. If the onhand quantity of any item drops below its ordering point the unit will make a request for relief supplies. Thus, the unit nodes of the S3 logistics network *pull* supplies into theater. For instance, a calvary unit may require an inventory of twenty thousand rounds of 20 mm shells with an ordering point of fifteen thousand. Initially the unit is stocked to one hundred percent, but after a sustained period of high intensity combat the unit is reduced to fourteen thousand rounds. The next daily inventory check after the ammunition inventory dropped below the ordering point of fifteen thousand rounds the unit will automatically order difference between its on hand stock and the stocking objective. In this case the unit would place an order for eight thousand rounds of 20mm shells. Simply put, in the sustainment phase units act as the exit nodes of the logistics network consuming goods and pulling follow on supplies into theater.

In summary, units play two separate and equally important roles in S3. First, they act as the commodities in the logistics network during the surge phase of the simulation. Second, units act as the network consumers during the sustainment phase. The actions of units drive the actions of all other objects in S3.

### 6. Bases

Bases are the suppliers for units in S3. Base inventories are pulled by unit consumption through the logistics network. In addition, base inventories depleted due to the filling of a unit order are replenished to normal when the base places a resupply order of its own. Any time a base places an order that cannot be filled by any other active base in the game it is forced to order from the supply base. This is the industrial base which introduces commodities into the network, continuously producing goods at a fixed rate on a daily basis. An example of a base would be a Naval Weapons Station. Let's say this base has a seaport with ammunition ships and an inventory of commodities being used in the theater to enable the resupply of units consuming items in combat and that the unit is a carrier battle group with the commodity in question is a Laser Guided Bomb(LGB). If the carrier battle group requested one thousand LGBs the Naval Weapons Station could fill that order, loading an ammunition ship with LGBs and any other ammunition destined to that theater, and sending the ship directly to the carrier for transfer to the carrier unit. Upon loading the ammunition ship the base would order an identical amount of ammunition from the logistics network. If there is an insufficient supply of LGBs to fill this order, the quantity is placed on backorder and the base is forced to wait for the supply base to produce the LGBs, thus entering additional commodities into the network.

## C. THE SCENARIO

The scenario in S3 is a user defined logistics network. Creating a scenario is much like building a pyramid. The initial work is hardest, and it gets easier as each successive level is built. In S3 the scenario can be thought of as a seven level pyramid. The first level involves the building of the scenario commodities. In similar fashion, the next five levels involve creating objects; transporters, subunits, bases, units, and prepositioned transporters. It is important to note that the sequence of these steps must be followed. A look at Figure 2-3 should make this sequence clear. Following the creation of all the required pieces of the scenario, the seventh and final level involves selecting the bases and units for inclusion in the scenario and creating the ground transportation networks for any train or truck routes from PODs or POEs to bases or units. The same is not required for ships and aircraft since the mere existence of airports and seaports is sufficient to allow arcs between bases and units. The completion of these seven steps leaves a scenario ready to be wargamed.

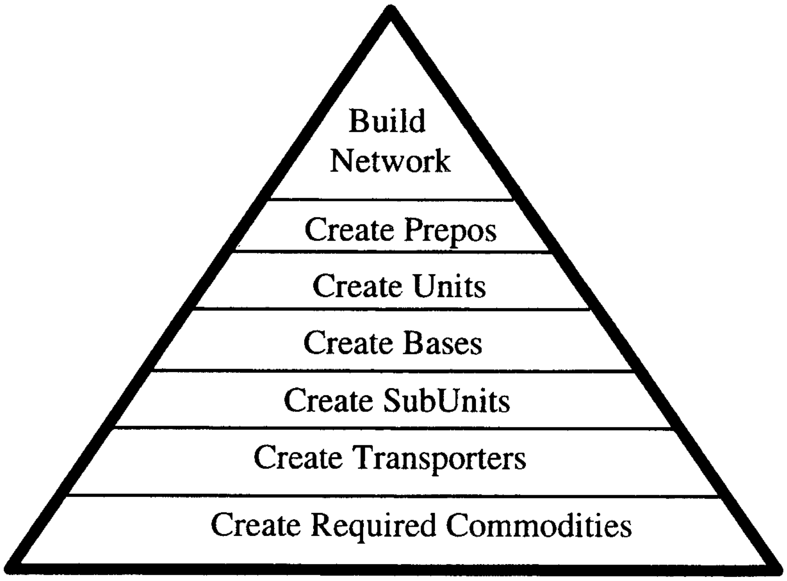


**Figure 2-3 Scenario Construction Pyramid**

Scenario building in S3 is designed to prevent redundancy. The commodities, transporters, units and bases entered as part of one scenario are available to be used in the building of any subsequent scenarios. S3 maintains a database which holds the required information to build all previously designed bases, units, commodities, subunits, and scenarios. This is an important feature of S3. Several steps of the scenario building pyramid can be averted if the required building blocks for a scenario already exist in the S3 database. This shortcut, however, presents two important pitfalls to the user. First, the locations of bases and units in the scenario should be verified since the locations of units and theater bases are sure to change from scenario to scenario. Second, the deployment times of all involved units should also be updated as this too should change for each scenario.

Each scenario is saved in the S3 database after completion. This allows the scenario to be recalled and wargamed by the user repeatedly. In addition, any existing scenario can also be modified by the user to reflect any desired changes. Changes can be entered at any point in the scenario construction pyramid.

In summary, an S3 scenario is a user defined logistics network required to wargame surge and sustainment for a given situation. A scenario is built following the steps of the scenario construction pyramid. Each user defined scenario is saved in the S3 database and available for selection by the user during the execution mode of the simulation.

## D. THE LOGISTICS NETWORK MANAGERS

### 1. Overview

The logistics network in S3 is run by two special Modsim objects that manage commodity and transporter requests. They are the logistics manager and the transportation manager. Each of these objects play a special part in determining the flow of commodities through the network. A brief description of the logistics manager and transportation manager follows.

### 2. The Logistics Manager

The logistics manager handles requests from scenario units and bases for commodities. It does the model *thinking*, deciding which base will fill a particular order. After deciding who will fill an order the logistics manager next builds a shipment, a route for the shipment to follow, and assigns the chosen supply base to fill the order. In the event one base cannot fill an entire order the logistics manager will continually break the original order into smaller shipments until the complete order is filled. If the sum of available commodities in scenario bases are still unable to fill the order the remainder is back ordered from the supply base.

### 3. The Transportation Manager

The transportation manager handles transportation requests from bases that are attempting to fill orders tasked to them by the logistics manager. Upon receiving an order each base will send the required items to its port for shipment. The port pools all commodities bound for the same destination, determines the amount of transportation required and the highest priority item in the pool. Once this is done a request holding this information is placed with the transportation manager.

The transportation manager maintains two lists, the request list and the available transporters list. Each time a new request or new available transporter enters one of these lists the transporter manager will attempt to find a match between the two. Once a match has been made the chosen transporter is told to go to the appropriate base. Upon arrival at the base the port will load the transporter and send it to the first base in the cargo shipment's route.

## E. SIMULATION AND THE HUMAN FACTOR

All of the computer elements of S3 have now been introduced, but looking back to our earlier discussions on why S3 is needed, it should be clear the most important part of S3 has yet to be covered. It is human interaction. Although not a part of the inner workings of the model the human user is the cog that drives S3. The logistics manager and the transportation manager take care of the routine number crunching movement of cargo, this follows in suit with other models that are currently used in wargaming. S3, however, is more than just a model; it is also a wargame. In S3 there is a higher level of control than the logistics and transportation manager. The primary controller in S3 is the human being behind the keyboard.

User interaction is the overriding force that drives unit activity, and hence all consumption, and movement of goods in S3. The S3 computer interface allows user interaction in the creation of the scenario building blocks, in tying them together into a scenario, and finally in the actual wargaming of that scenario. Many logistics models allow custom scenario building and deployment timelines, but few if any allow human interaction between the start and finish of a simulation run.

Human interaction in S3 is done through the six user control points of S3. These control points have significant impact on the surge and sustainment of supplies into the scenario theater. The six control points are: time steps, stock levels, combat intensity, port existence and capacity, the existence of transporters, and the deployment of units. A brief explanation of how each of these controls are exercised follows below.

### 1. Time Steps

The opportunity for user interaction during the running of the simulation is where S3 departs from being just a logistics model and becomes a logistics wargame. This interaction is possible because S3 is run in **time steps**. Each time the user tells the simulation to run there is a preset time at the end of which the simulation will stop and wait for further user direction. The length of each time step is originally set at twenty-four hours, but is variable by the user up to any length. Time stepping is important, because it determines when the remaining five control points can be exersized.

### 2. Stock Levels

All stock levels for units and bases can be altered through the computer interface. Items can be added to inventories and the associated stocking information can be set to the user's needs. This is an important feature allowing the user to push supplies into a theater. In addition it allows the user to create consumption rates that differ from the daily percentages built into subunits. For example an item maintained in a theater logistics site inventory could have its stocking objective raised by the user forcing the base to order more of that item from the logistics manager, likewise an enemy raid on the base could cause a reduction in inventory, also resulting in an order placement.

### 3. Combat Intensity

Combat intensity for all units in a scenario is determined by user interaction and is set at either high, medium, low, or none. This has a direct impact on consumption. For example, a unit in a defensive pre-conflict posture is consuming at a low level and falls under a surprise attack from the enemy. The S3 interface allows the user to immediately raise the unit combat intensity to high to reflect the increased level of combat. As a result the unit will begin to consume more of each item in its inventory during each daily consumption period.

### 4. Port Existence and Capacity

Port existence and capacity is also variable through the user interface, this has a direct impact on port throughput and the supply rate into theater. The existence of ports can be toggled on or off for each base or unit at any time step. In addition the size of transporters allowed into a port and the number of unloading spots are variable. An example of this feature in use could be the entry of forces into theater with only one Port of Debarkation(POD). The port could begin the simulation completely closed. Perhaps in the wargame it is determined only a pier a week can be readied for use. This can be modeled in S3 by starting the simulation with the port closed. After a seven day time step the user could put the port into existence and create a single pier. Every additional seven day time step the user could add a new pier until the maximum number of piers for that particular port have been created.

### 5. Transporter Existence

The life of transporters can also be controlled by the user. For example, in a wargame scenario there is an action in which enemy submarines sink a breakbulk ship. S3 allows the user to reflect this shock to the logistics system by simply removing the ship from the scenario along with its cargo. The unit or base that the cargo was destined for will then proceed to completely reorder the items in question. In addition to destroying transporters the user can also add transporters at any time step.

### 6. Unit Deployment

Unit deployment is the most important control point of the six that have been mentioned. The activation of units, or the surge portion of this model provides the largest impact of any possible action on the logistics network in S3. Each unit can be activated in one of two ways specified by the user, by a preset time or by user interaction during simulation. Each way requires the user's input. For example, when creating a unit the activation time is set by the user at one hundred hours. After the simulation is started, at any time up to one hundred hours the user may prompt that same unit to activate at any time step.

## F. SUMMARY AND REVIEW OF S3(VERSION 1.0)

Up to this point we have discussed the pieces of S3, the logistics network they form when placed together, and the logic behind their interaction in that logistics network, but this does not completely cover Version 1.0. What remains to be done is to introduce the user to the actual computer menus through which S3 is operated. Because S3(Version 2.0) is a continuation of S3(Version 1.0) the user interface has many similarities, so in the interests of space and to prevent redundancy this introduction will be left for later in this thesis. The next step will be to present the requirements for S3(Version 2.0) in Chapter III. Chapter IV will follow with an introduction to using S3, both the old and new elements.

Despite not fully having covered S3(Version 1.0) now is the appropriate time to summarize its capabilities and to match these with the original requirements put forth in chapter one. First and most importantly the logistics network has been successfully modeled. All of the required building blocks have been programmed. Second, the user interface is in place and operational allowing the user to build specific scenarios and to interactively time step through their execution. Finally, the arduous task of entering the entire U.S. military force into the S3 database has begun. Together these accomplishments have produced the first running version of an interactive logistics wargaming tool.

S3(Version 1.0) was a step forward in the right direction, but looking closely at S3(Version 1.0) one can see that there is still room for growth. Though having satisfied many of the requirements originally set forth, there are still areas of S3 that need improvement and many more that need to be added completely. The next chapter covers those problems identified for this thesis.

# III. S3 (VERSION 2.0)

The first release of S3 did not completely fill the requirements originally put forth by the Naval War College for a logistics wargaming tool. What Version 1.0 *did* accomplish is to create an engine for the simulation, with all of the inner workings and database manipulations required to build and operate a complex logistics network. Many of the things it did not do have been accomplished in this thesis. First, the simulation engine has been fine tuned to further increase model fidelity. Second, the desired output of the model, information providing insight about what the designed logistics network can do, has been made more visible.

Several additions or changes to S3 have been incorporated into Version 2.0 that provide improvements in both areas identified above. For this thesis five separate tasks have been identified and completed, the first two concern the availability of information about the logistics network while the remaining three concern model fidelity.

The improvements in the accessibility of information are as follows. First, an order tracking system has been added to allow better network visibility. In Version 1.0 the only information available about unit or base commodity orders was the fact that they had been made. Only through a great deal of effort could the user determine where supplies were, and even then the user could not determine which unit the supplies were destined for or in what time frame they would arrive.

Second, a diagnostic module has been created to provide information about the effectiveness of the logistics network in a given scenario. This module collects and presents data about unit and theater closure, transporter usage, and port throughput. The first version of S3 did not provide any data output whatsoever about theater closure giving only snapshots of unit inventory data. The combined effect of the improvements is that the user has been given a much better picture about what the logistics network is doing, has done and what changes to the network would have the biggest impact (i.e. insufficient transporters, ports, etc.).

The last three changes to S3 concern model fidelity. First, the old version of S3 did not allow the movement of units. Current military planning involves up to two theaters of operations at one time (two nearly simultaneous Major Regional Conflicts) and often mentions moving or "swinging" units in-between those theaters. This version of S3 allows the user to move units from one theater to another.

Second, the modeling of transporters in S3 in Version 1.0 did not reflect the real world. Travel distances for shipping were calculated with no regard to obstructive land masses. A ship simply took the shortest path between two ports. This version of S3 reflects the realities of the globe and uses voyage waypoints to provide routing that uses such unavoidable ocean connections as the Straits of Gibraltar and the Suez Canal.

Last of all, numerous computer bugs were identified in the computer code released with S3(Version 1.0). This is a common occurrence in the progression of large computer programs like S3 and these problems were not unexpected. Any errors *identified* in the Modsim code for S3(Version 1.0) have been corrected. This task is by far the most ambiguous of the five mentioned, and it has been the most difficult to accomplish.

The rest of this chapter provides an in-depth list of the additions that have been made to bring about the five improvements mentioned above.

## A. ORDERS TRACKING

An important difference between S3 and other logistics models is the fact that the user interacts throughout the running of the model. In order for this interaction to be meaningful the user needs to have the best picture of what the logistics network is *doing* in order to make timely decisions during wargaming. In the original S3 there were no action pictures, no feel for what was going on. There was an idea of what the network had *done*, but this picture was incomplete in that it only provided this information at the unit level.

An example to show the above shortcomings in S3 (Version 1.0) follows. Suppose that S3 is being run as the logistics constraint to a concurrent ENWGS wargame. In the ENWGS game a tank battalion wishes to engage the enemy and the game umpire has decided to check the logistics status of the tank unit using S3 before approving the action. To get this information in S3(Version 1.0) the umpire would look at the tank battalion's unit display screen to see a listing of the current inventory of that unit. This listing would tell him whether the unit is at full strength, or if it is not, that a certain percentage of that inventory is on order. This information is acceptable to the umpire for the task of determining whether the tank unit is capable to fight, but it is unacceptable to the game players in the team cell who control the tank unit. With Version 1.0 a negative answer would be given with none of the information the tank unit owner would need to plan further actions. That team cell might also want to know where the current orders of commodities are that would allow the unit to become active, and when they would arrive at the unit. The problem we are discussing has its counterpart in the real world called intransit visibility. Without some way of *seeing* goods in transit, the cell players are handcuffed from making decisions based upon the readiness of the unit in question stealing from the effectiveness of the wargame. The net effect of this lack of information is to waste the time of both the players and the umpire.

To curtail this problem each order placed by a unit or base must be visible from the time it is delivered to the logistics manager to the day that it arrives. The order tracking system provides the required information. This system tracks each order from start to finish; to include order placement, order filling at the shipment origin, transit to the requester, and finally the delivery of the shipment. Several quirks of the ordering process are also accounted for in this system to include backorders and the splitting of individual orders into several different shipments coming from several sources.

The information held in the order tracking system is made available to the user through the unit (or base) inventory display. It indicates the status of the order, including all shipments associated with the order and the transporters those shipments are assigned to or the ports they are currently waiting in. With this information a wargamer will not only be able to determine unit supply status at one moment in time, but also at some time in the future.

## B. THE DIAGNOSTICS MODULE

The first version of S3 did a fair job of showing unit level supply status, but it did not provide any performance measures for the overall logistics network. Data on force closure, port throughput, transportation availability and usage were not collected. The Surge and Sustainment Simulation should provide this data to wargamers, especially for use with seminar wargames. The "what ifs" of logistics wargaming require that the above measures be taken in order to begin to make comparisons between different logistic situations.

A diagnostics module has been added to S3 that provides data on the operational efficiency of the logistics network. This module provides theater closure data, and port and transporter usage information. The information is available at each time step during the running of a simulation. This module has been broken into three parts: closure, transporter, and port information. The requirements met for each of these follow in the paragraphs below.

Closure data is the most important of the three information groups mentioned. The most frequently asked logistics question at wargames is "when can we get the troops into theater." The diagnostics module should provide the answer to this question. To do this it must produce two types of information, theater closure and unit closure data.

The theater closure data collected should provide a daily update to the aggregate total of commodities received in theater. This information should break down the commodities into four groups; petroleum oil and lubricants (POL), ammunition, unit equipment, and personnel.

Unit closure data should include the recorded time of occurrence for important events in the unit deployment cycle. Significant dates such as when a unit began arriving in theater, when unit strength reached twenty-five, fifty, and seventy-five percent, and the time a unit is considered active (ninety percent arrived), and finally the date the unit reaches full strength all should be recorded for later analysis.

The second group of information mentioned concerned transporter usage. The user should know how much a particular transporter or transporter type is being used. This should include the number of trips made by a transporter, the average cargo it carried, and a comparison of this figure to the transporters maximum capacity. This data could be helpful in giving insight into any number of questions from where transportation should be located, to what size transporter would best suit a particular scenario. For example, if a transporter group is found to have a high instance of idle time that indicates that there are too many of those transporters and vice versa if the transporter group is being used all of the time.

The third and final information group is port data. Port figures should include the average number of transporters in port, the number currently in port, and the average time transporters spent waiting for berths at that port, and an idea of what the port throughput is. The data should be presented using the same cargo groupings planned for unit closure. This information should allow the user to determine whether the port has become a bottleneck to cargo passing through the logistics network, and also allow some degree of sensitivity analysis of the affects various numbers of unloading spots have on port throughput.

In summary, all three of these information areas are accessible in S3 through the diagnostics module. The diagnostics module provides an interface for display of this information at each time step. There is a synergy between all three information types, if closure times are not satisfactory an examination of both transporter usage and port throughput should shed light on where the logistics system can be modified to improve system performance. The introduction of this diagnostics module was critical in order for S3 to fulfill its role in logistics wargaming.

## E. DUAL THEATER CAPABILITY

The Department of Defense currently feels it is possible that U.S. forces may become involved in two nearly simultaneous major regional conflicts (MRCs). A logistics model for today's wargamer must be able to handle this situation. One hurdle stands in the way of S3 capability to do so. In most two MRC scenarios discussed at the Pentagon certain units are tasked with participation in both theaters. To do so any unit which is designated for dual use must be transported between theaters. S3 needs the ability to move units, particularly from one theater to another.

To successfully move a unit in S3 four things must be accomplished. First, the unit's position must be changed to reflect the new location. Second, the current unit commodity orders must be rerouted to the new location. Third, the on hand inventory at the old location must be routed and transported to the new location. In other words the unit must become a commodity and be redeployed. Finally, a new transportation network for the unit's new location should be constructed if the unit has a truck stop or railyard. The new dual theater capability in S3 meets all of these requirements.

## F. TRANSOCEAN WAYPOINTS

Transocean waypoints is the last completely new addition to S3 accomplished in this thesis. The modeling of shipping routes in S3 now takes land mass into account. Previously in Version 1.0 ships were assigned routes from port to port using great circle route courses. This method is gross and inaccurate. For example, a ship going from Los Angeles to Boston must either go through the Panama Canal or around the southern tip of South America. It should not go directly from Los Angeles to Boston via a course that coincides with the interstate highway network. S3 required some mechanism to minimize land mass encounters by shipping. Important waypoints such as the Straits of Gibraltar or the Panama Canal have become part of the routing used by S3's sea transporters. The introduction of these waypoints was achieved by adding additional nodes to the logistics network. These nodes are positioned using latitude and longitude, but have none of the other features exhibited by units and bases. These new objects are called waypoints in S3 (Version 2.0) and they provide more realistic routing that follows the actual shipping lanes used by today's merchant shipping and will produce more truthful outcomes for unit surge and sustainment.

## G. DEBUGGING AND UPDATING EXISTING CODE

The last work accomplished by this thesis involved taking a second look at the Version 1.0 computer code. Any identified problems or bugs that existed in the original S3 Modsim code have been eliminated. Taking into account the short time frame given for thesis work and the large effort required to build a program like S3 it should be no surprise that the debugging effort following S3(Version 1.0) was quite large. For those familiar with computer programming a majority may agree that debugging was the most difficult of the tasks completed for this thesis. "Debugging and improving code efficiency" is a general statement that covers a large number of individual tasks. Although this list may for the most part only be a series of small jobs, the sum of the work is important and this thesis could not disregard it. The correction of identified errors and the improvement in efficiency of existing code, wherever possible, has been completed and was a major portion of this thesis.

## H. SUMMARY

With the commonness of software upgrades in today's computer market it is only logical that the Surge and Sustainment Simulation (Version 1.0) should also be improved over time. This thesis was written based upon the premise that S3 Version 1.0 had room for growth and it is hoped that the work done in it will have filled much of that room. Chapter III has described a number of improvements in the S3 model that have been incorporated into the next generation of S3, Version 2.0. The next chapter addresses the transition of the above requirements into functioning Modsim code. Chapter IV will introduce S3(Version 2.0). Not only will the new features of S3 be demonstrated, but the entire simulation will be reviewed as promised in Chapter II, including the remaining Version 1.0 features.

# IV. WARGAMING S3 (VERSION 2.0)

## A. INTRODUCTION

This chapter is written to provide the reader with a demonstration of S3(Version 2.0). The purpose of the demonstration is to show that the goals of this thesis have been met and also to serve as a user's guide for future wargaming of the model. A scenario has been created for this demonstration that will exercise all aspects of the simulation and give the user insight into how to use the model.

The chapter is organized as follows. S3 will be covered in two sections, SetUp and Execution. The SetUp discussion is broken into three parts. First, the scenario will be introduced along with the U.S. forces that will be employed. Second, the preparation work required to be done before building a Scenario in S3 will be covered, Third, the actual building of the of a scenario using the Surge and Sustainment model will be walked through. The Execution section discussion is in three parts. First, the demonstration scenario event timeline will be described. Second, the six human control points from chapter two will be demonstrated. Third, the new additions to S3 introduced for Version 2.0 will be reviewed and their impact demonstrated.

## B. SETUP

### 1. The Demonstration Scenario

The background behind the conflicts in this scenario and a listing of the forces to be supported logistically will be given in this section. Before that is done it is important to note two points about this scenario. First, this scenario is not meant to represent any real crisis, the scenario is simply a hypothetical vehicle for showing S3. Second, the types of forces and the actions they are involved in during the model demonstration have been chosen to display the flexibility of the simulation and not to reflect any real capabilities or tactical ideas about the employment of those forces. Keep these points in mind throughout the rest of this chapter.

In order to ensure a complete demonstration of the capabilities in S3 a robust scenario has been created. The scenario is designed to model two nearly simultaneous regional conflicts. The first conflict takes place in the Middle East, where an expansive nation, country Brown, has invaded neutral country Blue threatening regional stability and the production of oil. The United States has chosen to intervene in country Blue's behalf. A second smaller conflict takes place in country Gold, a small peninsula nation of Southeast Asia. This operation begins somewhere near the end of operations in country Blue. Due to an area peace agreement between the United States and several regional countries including country Gold, the United States has stepped in to restore order. The United States stated goals in country Gold will be to strike down a self imposed military regime and install a previously exiled and democratically elected government.

The enemy forces to be faced in the above conflicts are generally small. The Brown forces are made up primarily of ground forces, some of these forces mechanized, which are supported with limited air power. In addition Brown has a small navy comprised of coastal patrol boats with mine laying capability and several dated diesel submarines. Brown also has a relatively sophisticated command and control capability. In country Gold the enemy forces are more primitive, and made up entirely of ground forces which maintain a low state of readiness. Intelligence shows these forces have purchased several cheap surface to air and surface to ship missiles.

The forces to be used by the United States to counter the above threats have been selected from those currently available from the Army, Navy, Marine Corps, and Air Force toolkits. The list of forces to be provided to the Joint Commander for each conflict follows below:

## MIDDLE EAST REGIONAL CONFLICT

### NAVAL FORCES

Commander, Middle East Force

- USS LaSalle (AGF-3)
- USS Stout(DDG-55)
- USS Jarret(FFG-33)
- USS Samuel B. Roberts ( FFG-58)
- USS Avenger (MCM-1)
- USS Patriot (MCM-7)

Commander, Carrier Strike Force 50 (CTF-50)

- USS Abraham Lincoln ( CVN-72)
  - Commander Carrier Group One
  - Commander Destroyer Squadron TwentyThree
  - Commander Carrier Airwing One
    - 24 F-14D
    - 24 F/A-18C
    - 12 A-6E
    - 4 EA-6B
    - 5 E-2C
    - 8 S-3B
    - 2 ES-3
    - 6 SH-60F
    - 2 HH-60H
- USS Chancellorsville (CG-62)
- USS Shiloh (CG-67)
- USS John S. McCain ( DDG-56)
- USS O'Brien (DD-963)
- USS Rentz ( FFG-46)
- USS Vandergrift ( FFG-48)

Commander, Amphibious Task Force 51 ( CTF-51)

Amphibious Ready Group Alpha ( CTF - 51.1A)

- USS Essex ( LHD -2)
  - Commander Amphibious Group Two
    - 6 AV-8B
    - 12 CH-46E
    - 4 CH-53E
    - 3 UH-1N

4 AH-1W
3 LCAC
USS Germantown ( LSD - 42)
3 LCU
USS Harpers Ferry ( LSD - 49)
2 LCAC
West Pac MEU SOC
6 155MM Howitzers
2 105MM Howitzers
1 LAV-C2
1 LAV-L
2 LAV-M
7 LAV-25
2239 Marine Corps Personnel Embarked

Amphibious Ready Group Bravo (CTF.51.1B)

USS Saipan ( LHA -2)
6 AV-8B
12 CH-46E
4 CH-53E
3 UH-1N
4 AH-1W
4 LCU
USS Whidbey Island ( LSD-41)
4 LCAC
USS Carter Hall ( LSD - 50)
2 LCAC
Med MEU SOC
6 155MM Howitzers
2 105MM Howitzers
1 LAV-C2
1 LAV-L
2 LAV-M
7 LAV-25
2274 Marine Corps Personnel Embarked

Commander, Submarine Task Force 52 (CTF-52)

USS Helena ( SSN-725)
USS Chicago( SSN-721)

Commander, Maritime Patrol & Reconaissance Force 53 ( CTF-53)

VP-1 ( 8 P-3C) Diego Garcia
VP-8 ( 8 P-3C) Masirah
VQ-1( 2 EP-3) Masirah

## AIR FORCE

Country Blue
6 F-15C
12 F-15E
5 E-3B ( AWACS)
6 KC-135R

## ARMY FORCES

82nd Airborne Division, Ft Bragg, NC
3rd Armored Cavalry Regiment, Ft Bliss, TX
10th Mountain Division, Ft Drum, NY

## SOUTHEAST ASIA REGIONAL CONFLICT

## NAVAL FORCES

Commander, Carrier Strike Force 70 ( CTF - 70)

USS Nimitz ( CVN - 68 )
Commander Carrier Group Three
Commander Destroyer Squadron Twenty One
Commander Carrier Airwing Three
24 F-14D
24 F/A-18C
12 A-6E
4 EA-6B
5 E-2C
8 S-3B
2 ES-3
6 SH-60F
2 HH-60H
USS Antietam ( CG - 54)
USS Monterey ( CG -61)
USS Curtis Wilbur ( DDG - 54)
USS Caron ( DD - 970)
USS Taylor ( FFG - 50)
USS Elrod ( FFG - 55)

Commander, Amphibious Task Force 71 ( CTF -71)

Amphibious Ready Group Alpha ( CTF - 71.1A)

USS Wasp ( LHD -1)
- 6 AV-8B
- 12 CH-46E
- 4 CH-53E
- 3 UH-1H
- 4 AH-1W
- 3 LCAC

USS Tortuga ( LSD-46)
- 3 LCAC

USS Pearl Harbor (LSD-52)
- 3 LCU

East Pac MEU SOC
- 6 155MM Howitzers
- 2 105MM Howitzers
- 1 LAV-C2
- 1 LAV-L
- 2 LAV-M
- 7 LAV-25

2239 Marine Corps Personnel Embarked

Amphibious Ready Group Bravo (CTF.71.1B)

USS Belleau Wood ( LHA -3)
- 6 AV-8B
- 12 CH-46E
- 4 CH-53E
- 3 UH-1N
- 4 AH-1W
- 4 LCU

USS Ashland ( LSD-48)
- 4 LCAC

USS Oak Hill ( LSD - 51)
- 2 LCAC

YOKUSKA MEU SOC
- 6 155MM Howitzers
- 2 105MM Howitzers
- 1 LAV-C2
- 1 LAV-L
- 2 LAV-M
- 7 LAV-25

2274 Marine Corps Personnel Embarked

MARINE CORPS ( Land Based)

7th MEB, Okinawa
MPS 3, Diego Garcia
15,000 Marine Corps Personnel

AIR FORCE

Country Gold
5 E-3B ( AWACS)
6 KC-135R
6 F-15C
12 F-15E

ARMY FORCES

25th Light Infantry Division,Ft Shafter, HI (Swung from first conflict)
101st Airborne Division, Ft Campbell, KY

Remember that the forces provided for this scenario are chosen to flex the simulation and exist for demonstration purposes only. Any lack of realism in how forces are employed should be ignored for our purposes here. The remainder of this chapter will deal with building this scenario and wargaming it using S3.

### 2. Setup Preparation

Getting ready for a wargame has always been and always will be the most time intensive period of the wargaming process. When preparing to wargame S3 it is imperative to collect the data required to build a scenario *before* the user sits down to enter it into the computer. This means not only deciding what forces will be used by S3, but also what bases, commodities, transporters, subunits, and ports will be available. In addition, the specific data that goes with building each object also must be collected and verified with the appropriate agencies.

The rest of this section explains the S3 database, the information required to build the objects for a scenario, and the need for a scenario deployment timeline. An understanding of each area is essential for a smooth scenario setup.

### *a. The S3 Database*

The S3 database is designed to minimize the overall scenario SetUp workload. When objects have been created in S3 once, they become available for use in all future scenarios. Thus, the database is the memory of S3 that prevents the user from having to "recreate the wheel" and start from scratch in creating objects for each new scenario. Over time, S3 should develop a complete database of forces and commodities that will require only periodic updates. This feature is very useful considering the large amount of effort required to build even the simplest of scenarios.

The scenario that will be created later on in this chapter will make use of this database. Many of the forces listed in the preceding section already existed as objects in the S3 database and required only minor modifications in order to fit the demonstration scenario.

The organization of the S3 database is on three levels. The first level is generic data, the second level is prepositioned ships, base or unit data, and the third level is scenario specific data. The generic data consists of commodity, transporter, and subunit data files. All generic datafiles are scenario independent. The commodity and transporter files hold all the basic information to create any commodity or transporter ever built over the lifetime of the S3 program. The subunit datafiles hold the inventory data required to support that subunit. The objects built using these datafiles can be duplicated many times for the running of any scenario. For example, if a scenario calls for ten B747 aircraft S3 will pull the data to build the B747 from the generic level datafile and make ten identical transporters. The current S3 database holds seventy-five commodities, forty-four subunits, and eighteen transporter types. None of the objects in these data files will be used in any scenario, however, unless they are called for in a prepositioned ship, unit or base datafile.

Second level datafiles specify information relating to the generic level datafiles. Prepositioned ship datafiles specify two things, a ship type from the transporter data files and an inventory of cargo from the commodities file. Base datafiles specifies the transporter types that will begin at its ports and the commodities that will be held in its inventory once the simulation begins. Unit datafiles specify the subunits that will be used to build the unit inventory, which indirectly specifying the commodities required for that unit, and the transporters that will begin simulation in the unit's ports. These files are scenario dependent and they are not able to be replicated. The numbers held in these datafiles have to be confirmed prior to scenario execution, because they ultimately determine the quantities and types of commodities and transporter that will be used in any scenario.

The third level datafiles are scenario specific. Each scenario has several files that are used to build the scenario each time it is being executed. The scenario execution data files hold information on the particular bases, units, and prepositioned ships that will be used in the scenario. These datafiles specify the second level datafiles which specify the generic level datafiles that will be used to construct the scenarios logistics network. Ultimately the aggregate total quantities and types of commodities and transporters that will exist at the beginning of any simulation is determined by the scenario datafiles.

### *b. Object Building - Preliminary Information*

As noted in the scenario introduction the most important work for setting up an S3 simulation is done beforehand. Not only should the forces be laid out, but the exact commodities, transporters, subunits, units, and bases must also be identified prior to using the Scenario Setup. The demonstration scenario forces have been listed above.

What follows below is a listing of all the objects required to model those forces and the logistics netork required to support them. There is additional required information beyond the simple naming of these objects that also needs to be collected but has not been displayed here for reasons of brevity. Later, the SetUp portion of this chapter will provide examples of this additional information as the steps of the S3 Scenario Construction Pyramid are followed. A listing of the required objects follows :

## COMMODITIES

The following commodities will be used and are listed by classes:

Commodity Class: Fuel

JP-5
DFM
Water
DIESEL
MOGAS

Commodity Class: Subsistence

FFV
Frozen Goods
Dry Stores

Commodity Class: Ammo

Missiles:

TLAM-N
TLAM-C
TLAM-D
TASM
Harpoon ( Air)
Harpoon ( Sea)
ASROC
Sea Sparrow
SM-1MR
SM-2MR
SM-2ER
HARM
AIM-54C
AIM-9M
AIM-9L
AIM-7M
AMRAAM
AGM-65
AGM-62
Penguin
TOW
TOW II
HELLFIRE
DRAGON

Guns:

20MM
20MM/76
25MM
40MM Grenade
76MM/62
76MM/50
81MM Mortar
105MM
127MM/54
155MM
120MM
7.62MM
.50CAL
M16A1

Bombs:

Mk-80 Series
Paveway II
APAM
Rockeye
FAE

Misc:

Sonobouys
Mk-46 Torpedo
Mk-48 ADCAP
Mk-54 Depth Charge
SRBOC

Commodity Class: Major Equipment

M-1A1
AH-64
M-109
105HOW
UNITEQ

Commodity Class: Other

Mail

Units of measure for the above categories follow these general guidelines:

| | |
|---|---|
| Missiles, bombs, and misc. | Individual Rounds |
| Gun Ammo, UnitEq | 48 x 40 x 72 Pallets |
| Liquids | 42 gallon barrels |
| Major Equipment | Individual Units |

## TRANSPORTERS

The following Transporters have been built for the demonstation scenario:
- C-5A Galaxy aircraft
- C-17 aircraft
- C-141B Starlifter aircraft
- B-747 CRAF aircraft
- Tanker ( 200,000 bbls.)
- Comet Class RoRo cargo ship
- SL-7 class RoRo cargo ship
- C- 4 - S - 1H breakbulk cargo ship
- Generic breakbulk cargo ship
- Large Medium Speed RoRo ( LMSR)
- Hauge Class prepositioning ships
- AE26 class ammunition ship
- A0177 class oiler
- Tank Truck Convoy ( Ten 5000 Gal. vehicles)
- Breakbulk Truck Convoy ( Ten 5-ton vehicles)
- Unit Equipment Train ( Fifty 54' flatcars)

- Passenger Train ( 7 coaches, 14 sleepers, 1 baggage car)
- Tank Train ( Forty 20,000 Gal. tank cars)

## SUBUNITS

The following SubUnits will be required to build the scenario Units:

- Ships:

CVN68
CG47
CG52
DDG51
DD963
FFG7
LHD1
LSD41
MCM1
AE27
AFS1
AO177

- Aircraft:

F-14D
F/A-18C
A-6E
EA-6B
E-2C
S-3B
ES-3
SH-60B
SH-60F
HH-60H
P-3C
EP-3
KC-135
E-3B
F-15C
F-15E
AV-8B
UH-1N
CH-46E
CH-53E
AH-1W
AH-64

- Ground:

155MM Howitzer
105MM Howitzer
LAV-C2
LAV-L
LAV-M
LAV-25
M-1A1
M-109
InfantryBattalion
M1Platoon

## UNITS

For the demonstration the following Units were required:

REGIONAL CONFLICT ONE:

- Middle East Force

1 AGF3
1 DDG51
2 FFG7
2 MCM1
2 SH-60B

- Task Force 50 ( CTF 50)

1 CVN68
2 CG52

1 DDG51
1 DDG963
2 FFG7
24 F-14D
24 F/A - 18C
12 A-6E
4 EA-6B
5 E-2C
8 S-3B
2 ES-3
8 SH-60F
4 SH-60B

- Task Group 51.1A (CTG51.1A)

1 LHD
2 LSD41
6 AV-8B
12 CH-46E
4 CH-53E
3 UH-1N
4 AH-1W
3 LCAC

- Task Group 51.1B(CTG 51.1B)

1 LHA
2 LSD41
6 AV-8B
12 CH-46E
4 CH-53E
3 UH-1N
4 AH-1W
4 LCU

- Air Force, Country Blue

5 E-3B
6 KC-135R
6 F-15C
12 F-15E

- VP-8, BlueAlpha, Country Blue ( VP-8)

8 P-3C

- VQ-1, BlueBravo, Country Blue( VQ-1)

2 EP-3

- VP-1, Diego Garcia ( VP-1)
8 P-3C

- 3rd Armored Cavalry Regiment ( 3ACR)
116 M-1A1
18 M-109
9 AH-64

- 82nd Airborne Division ( 82ABNDIV)
33 AH-64
54 105HOW
- 25th Light Infantry Division (25THLtInf)
50 UH-1H

REGIONAL CONFLICT TWO:

- Task Force 70 ( CTF 70)
1 CVN68
2 CG52
1 DDG51
1 DDG963
2 FFG7
24 F-14D
24 F/A - 18C
12 A-6E
4 EA-6B
5 E-2C
8 S-3B
2 ES-3
8 SH-60F
4 SH-60B

- Task Group 71.1A (CTG71.1A)
1 LHD
2 LSD41
6 AV-8B
12 CH-46E
4 CH-53E
3 UH-1N
4 AH-1W
3 LCAC

- Task Group 71.1B(CTG 71.1B)
1 LHA
2 LSD41

6 AV-8B
12 CH-46E
4 CH-53E
3 UH-1N
4 AH-1W
4 LCU

- Air Force, Country Gold

  5 E-3B
  6 KC-135R
  6 F-15C
  12 F-15E

- Seventh Marine Expeditionary Brigade ( 7MEB)

  12 M-109
  4 105HOW
  2 LAVC2
  4 LAVL
  4 LAVM
  14 LAV25

- 101st Airborne Division ( 101ABNDIV)

  33 AH-64
  54 105HOW

- 10th Mountain Light Infantry Division (10MtnInf)

  50 UH-1H
  109 155HOW

## BASES

The following bases are used in the S3 demonstration scenario:

- Theater Bases

BlueAlpha, Country Blue: Airhead, no stocked Commodities
BlueBravo, Country Blue: Commodities stocking point
BlueFoxtrot, Country Blue: POD for Country Blue Units
GoldAlpha, Country Gold : POD for Country Gold Units

- Intermediate Locations (ILOC)

BlueCharlie, Country Blue: Fuel stock point
BlueDelta, Country Blue: Fuel stock point

| | |
|---|---|
| BlueEcho, CountryBlue: | Fuel stock point |
| Diego Garcia: | Fuel and ordnance stock point, MPS homeport |
| Sigonella: | Fuel stock point |
| Rota: | Fuel stock point |
| Singapore: | Fuel stock point |
| Okinawa: | Fuel stock point |
| Guam: | Fuel and ordnance stock point |

- U.S. Bases (CONUS)

| | |
|---|---|
| NWS Earle, NJ: | Ordnance stock point |
| NWS Concord,CA: | Ordnance stock point |
| MOT Bayonne,NJ: | Port of Embarkation |
| MOT Oakland, CA: | Port of Embarkation |
| NSC San Diego,CA: | Port of Embarkation |
| NSC Norfolk,VA: | Port of Embarkation |
| Ft Bragg, NC: | Origin, 82nd Airborne Division |
| Ft Bliss, TX: | Origin, 3rd Armored Cavalry Division |
| Ft Campbell,KY: | Origin, 101st Airborne Division |
| Ft Drum, NY: | Origin, 10th Mountain Infantry Division |
| Ft Shafter, HI: | Origin, 25th Light Infantry Division |
| MARDETOkinawa: | Origin, 7th Marine Expeditionary Brigade |
| PopeAFB, NC: | Port of Embarkation |
| HollomanAFB,NM: | Port of Embarkation |
| Corpus Christi, TX: | Port of Embarkation |
| Wilmington, NC: | Port of Embarkation |
| Pearl Harbor,HI: | Port of Embarkation |
| Philadelphia,PA: | Port of Embarkation |

## PREPOSITIONED SHIPS

The following prepositioned transporters will be used:

- MPS Squadron 2, Diego Garcia

Bonneyman
Hauge
Baugh
Philips

### *c. The Unit Deployment Timeline*

Another important item of information required before building a scenario is the unit deployment timeline. Most military planners maintain deployment timelines that can readily be adapted to similar sized and similarily located wargame scenarios. This is not a requirement, as the units may be activated manually by the user, but if the wargame scenario permits preconceived deployment schedules it is highly encouraged that it be done. The scenario created for this demonstration involves two regional conflicts. The forces in the first conflict will be given a deployment schedule while those in the second will be left unscheduled to allow for uncertainty in their deployment and/or redeployment. The unit deployment timeline created for the demonstration scenario is shown below in Table II. Note that some of the units are beginning the simulation already deployed. This is done for two reasons. First, the navy is assumed already deployed to theater and is completely loaded out with initial inventories. Second, the air forces are currently treated as self deployable by S3, they will fly to the theater and therefore begin active in theater although at a minimum combat intensity level.

TABLE II

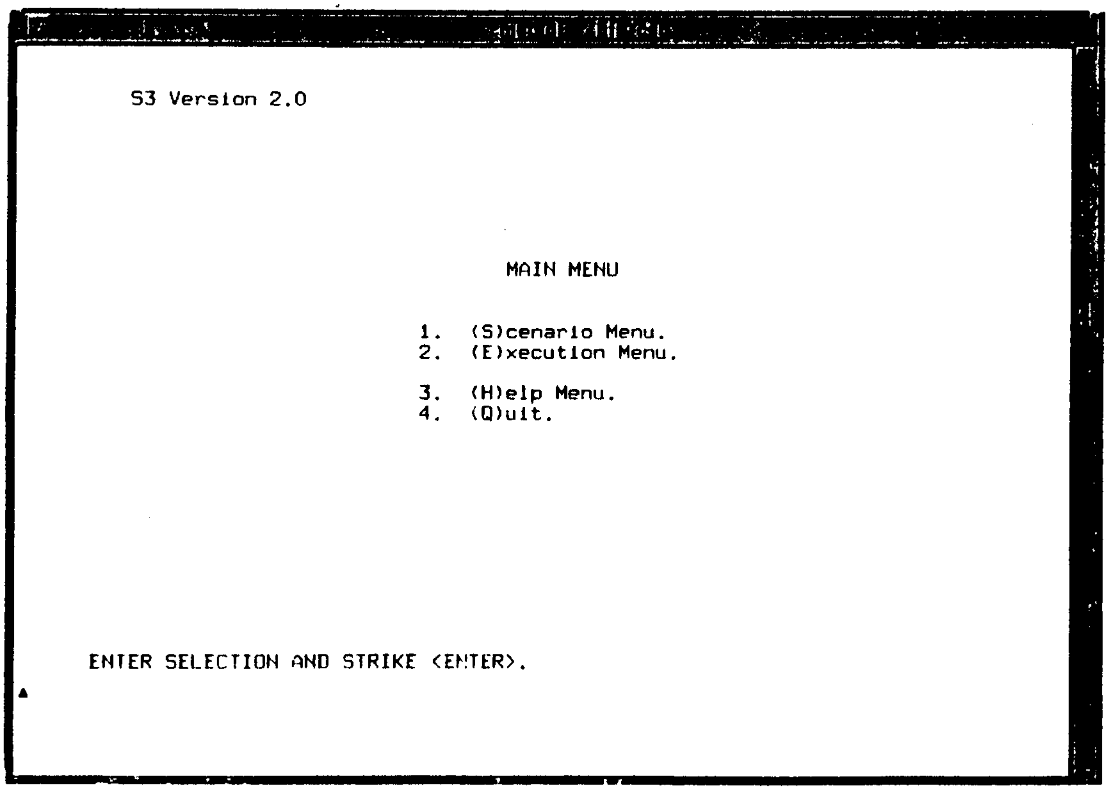
| **Unit Name** | **Deployment ( Days after C-day)** |
|---|---|
| CTF50 | Active Immediately |
| CTF51.1A | Active Immediately |
| CTF51.1B | Active Immediately |
| MEF | Active Immediately |
| CTF70 | Active Immediately |
| CTF71.1A | Active Immediately |
| CTF71.1B | Active Immediately |
| VP-1 | Active Immediately |
| VP-8 | Active Immediately |
| VQ-1 | Active Immediately |
| AF-CountryBlue | Active Immediately |
| AF-CountryGold | Active Immediately |
| 25THLtInf | +4 |
| 10MtnInf | +4 |
| 82ABNDIV | +7 |
| 3ACR | +7 |
| 7MEB | TBD |
| 101ABNDIV | TBD |

### 3. Entering A Scenario in S3

After all of the preliminary information has been collected for a scenario the next step is to create that scenario in S3. It is important to note that this step is not taken at the beginning of the wargaming, but rather at the end of wargame preparation. To build a new scenario is as simple as following the S3 Construction Pyramid from Chapter II. It is important to follow the order set out in the pyramid. All commodities must be entered first, then the transporters, then the subunits, etc. The next section attempts to explain and present examples of each step in the pyramid. By walking through the creation of several of the objects required for the demonstration scenario it is hoped that the construction process will become clear. Considering the size of this scenario only one example will be given for each step of the pyramid. The scenario entry walkthrough starts off with an introduction to the S3 Menu system and then works through the seven steps of the S3 Construction Pyramid.

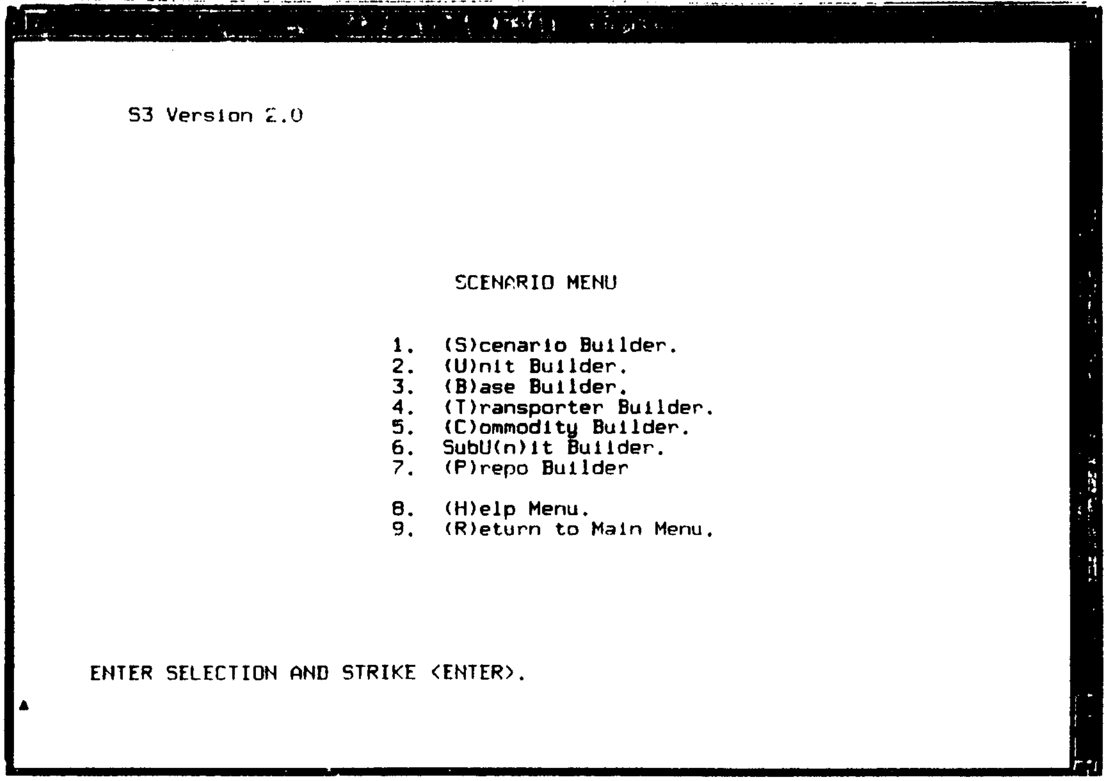
#### *a. Using S3 Menu Screens*

Before introducing the scenario setup section the menu system used in S3 (Version 2.0) must be explained. All user interaction in S3 takes place through a system of menu screens. Upon start-up of the S3 program the first menu displayed is the Main Menu (See Figure 4-1). This menu screen prompts the user to choose between four different actions; Scenario Building, Scenario Execution, Help, and Quitting the program. Note that the Help section has not been added to the simulation, so selecting Help currently gives no response. To make a selection from the menu screen the user should type the encapsulated letter or number concurrent with his choice. For example, the Main Menu encapsulates the letter S for choosing the scenario building section of S3. If the user types " **S** " or "s" the simulation will move into the Scenario Building menu screen. Similarly, all required user interaction in S3 is accomplished in this fashion, by moving



```
S3 Version 2.0



                              MAIN MENU

                         1.   (S)cenario Menu.
                         2.   (E)xecution Menu.

                         3.   (H)elp Menu.
                         4.   (Q)uit.




ENTER SELECTION AND STRIKE <ENTER>.
```

Figure 4-1 Main Menu Display



```
S3 Version 2.0



                              SCENARIO MENU

                         1.   (S)cenario Builder.
                         2.   (U)nit Builder.
                         3.   (B)ase Builder.
                         4.   (T)ransporter Builder.
                         5.   (C)ommodity Builder.
                         6.   SubU(n)it Builder.
                         7.   (P)repo Builder

                         8.   (H)elp Menu.
                         9.   (R)eturn to Main Menu.




ENTER SELECTION AND STRIKE <ENTER>.
```

Figure 4-2 Scenario Menu Display

to the appropriate menu screen and making the desired inputs. From most menuscreens there are three possible actions; moving forward or moving backward to another menu screen or entering data following a menu screen prompt.

Entering data occurs after the program queries the user for specific information such as the name of a base or the latitude of a unit. In these cases S3 will either provide an entry format to follow or a list of choices from which the user response should be copied. The user should type the entire entry followed by the **<enter>** key. It is important to note that Unix is case sensitive and any typing should be done exactly as the program asks or as the choices are displayed on the menu screen. An example of this would be selecting a base for entry in a scenario. The menu screen will ask for a choice from a list of available bases that are displayed on the screen. If the choice desired is "**FtBragg**" the user should type "**FtBragg<enter>**" and not "**FTBragg**" or "**FTBRAGG**" as these entries will not be recognized by the simulation.

### *b. Beginning the S3 Construction Pyramid: Building Commodities*

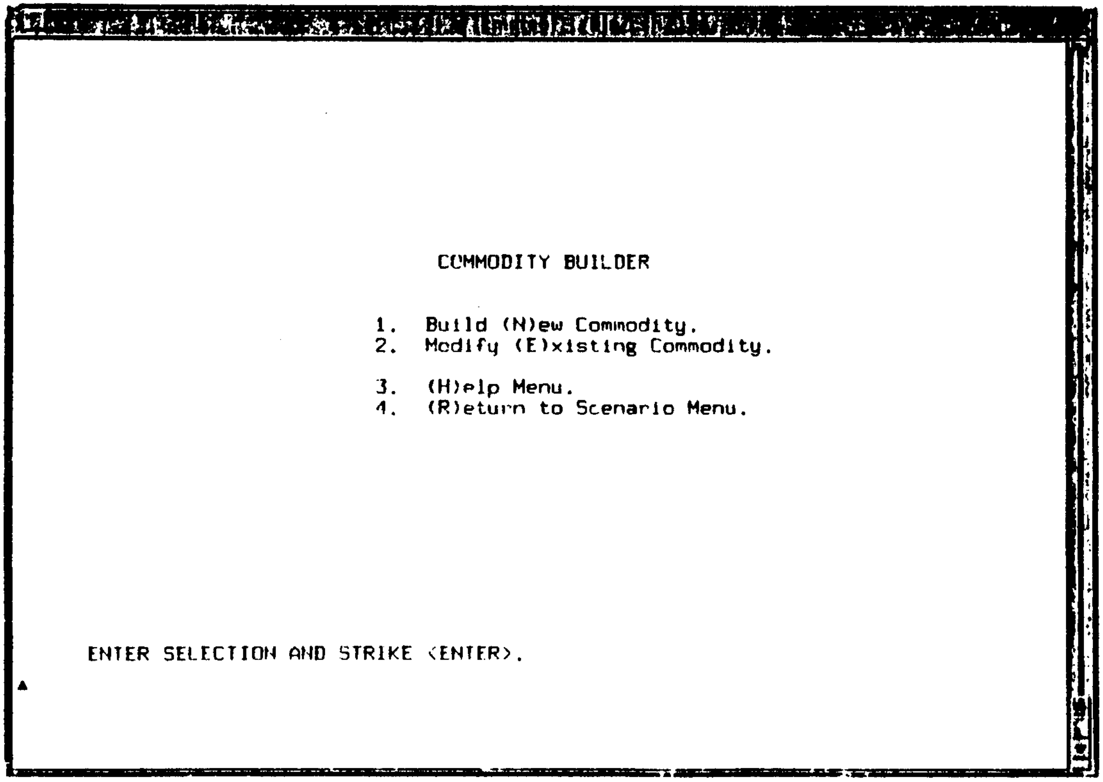
Armed with the knowledge about how to use S3 menu screens it is time to begin the task of learning how to set up a scenario in S3. From the main menu screen shown in Figure 4-1 choose "**S**" for the Scenario Menu. This choice will bring up the Scenario Menu.

The Scenario Menu (Figure 4-2) provides the user access to object building sequences for all steps in the S3 Construction Pyramid. Starting at the bottom of the pyramid commodities are the first objects that should be built. For the purposes of the S3 simulation all commodities used by the forces in any given scenario need to have been entered at this level. Remember that commodities and transporters need to be created in S3 and entered into the database only once. After that initial entry they will be available for the building of units and bases indefinitely. For the demonstration scenario most of the required commodities already existed in the S3 database. One new commodity called for in the scenario that was not in the database was the Dragon missile. The following steps trace the construction of this commodity in S3.

From the Scenario Menu choose "**C**" for Commodity Builder. The Commodity Builder screen ( Figure 4-3) will appear. To build the Dragon missile a new commodity needs to be created, so begin by selecting "**N**" to build a new commodity. After this selection is made S3 will prompt the user to enter all of the required information for building a scenario commodity. Simply follow the instructions provided on the display screen. First, enter the name of the commodity, "**DRAGON**". Second, enter the class of commodity from the list in Chapter II, this is "**Ammo**" for the Dragon missile. Third, enter the items production rate, for the Dragon "**10**" was entered to indicate a production rate of ten missiles per day by the industrial base. Next enter the dimensions of the commodity. This can be entered in individual rounds or in aggregate terms. Enter dimensions in inches. For the Dragon missile the dimensions entered should be **12**" X **12**" X **46**". Next, enter the weight in pounds; **67** pounds for the Dragon. Finally, enter the shipment priority for the commodity. For the Dragon the normal priority is "**4**" and emergency priority is "**1**". These numbers should be in a range from 1-12 from the priority chart given in Table I. Figure 4-4 gives the on screen picture of the entries made for the Dragon missile.

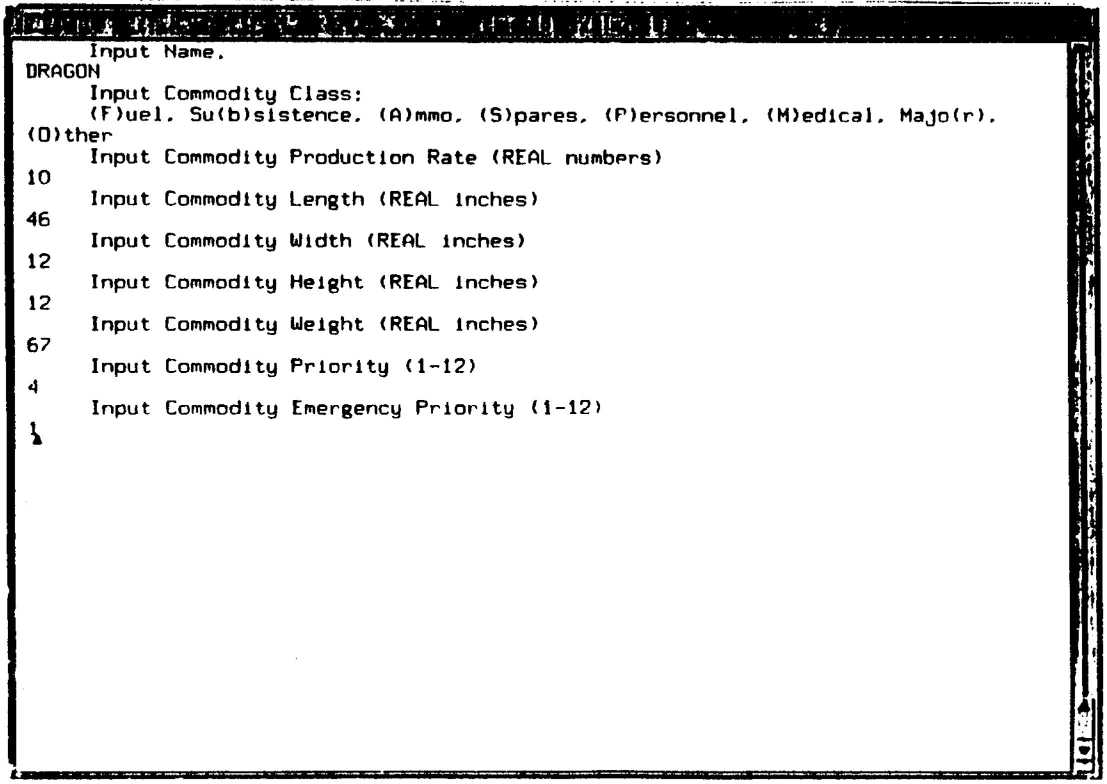
At this point S3 will ask if the commodity has been entered correctly and if it should be saved, enter "**Y**" for yes. Once a commodity has been saved it can be modified from the Commodity Builder screen by choosing "**E**" for enter modification. Any field previously entered can be changed from this screen. This feature holds true for all objects created using the construction, they all may be edited after they have been created.

An important point that is worth a second mention is when building a scenario all required commodities must exist before building any other required objects.



```
                         COMMODITY BUILDER

                     1.  Build (N)ew Commodity.
                     2.  Modify (E)xisting Commodity.

                     3.  (H)elp Menu.
                     4.  (R)eturn to Scenario Menu.




     ENTER SELECTION AND STRIKE <ENTER>.
```

Figure 4-3 Commodity Builder Menu



```
     Input Name.
DRAGON
     Input Commodity Class:
     (F)uel, Su(b)sistence, (A)mmo, (S)pares, (P)ersonnel, (M)edical, Majo(r),
(O)ther
     Input Commodity Production Rate (REAL numbers)
10
     Input Commodity Length (REAL inches)
46
     Input Commodity Width (REAL inches)
12
     Input Commodity Height (REAL inches)
12
     Input Commodity Weight (REAL inches)
67
     Input Commodity Priority (1-12)
4
     Input Commodity Emergency Priority (1-12)
1
```

Figure 4-4 Dragon Missile Entries

Commodities that do exist in the S3 database are the only ones available for new scenarios. Commodities do not have to be rebuilt for each new scenario. Any commodity that does not previously exist in the S3 database must be created before any further progress is made. Once all scenario commodities have been confirmed to be part of the database the user is ready to move up one step on the construction pyramid. By entering "**R**" for return, back up one menu screen and return to the Scenario Menu screen.

Before continuing up the scenario building pyramid here is a quick word on menu screens. After the Scenario Menu the entry screens used for the scenario building process are very similar for each object. Figure 4-3 is identical for them all except that the object name in the screen heading changes. Figure 4-4 is just a list of questions written to the screen by S3 and does not give any additional insight into how to use the program, although the types and number of questions change the general format is used to enter data when building all types of objects. For the remainder of this discussion on object building the specific menu screens will not be shown, and the reader should look back to the commodity builder screens for reference.

### *c. Building Transporters*

To build a transporter, select "**T**" for the Transporter Builder menu from the Scenario Menu. In the transporter builder menu select "**N**" to build a new transporter. The remaining steps all involve entering the requested data. For example, in the demonstration scenario a new ship object was created, the Large Medium Speed RoRo(LMSR). The first step in the transporter building process was to enter "**LMSR**' as the transporter name. Next, the class of transporter, "**ship**", and the transporter subclass, "**RoRo**" were entered. Following this step the transporter movement charachteristics were entered; the ships maximum speed, "**24**" knots, and the ships maximum range, "**10000**" nautical miles. Next, the ships dimensions; length, "**850**" feet, and width, "**106**" feet were entered.

The next group of fields to be entered concerned the transporters ability to carry cargo. To set the cargo limits on the LMSR the following data was entered; Cargo Area "**380000**" square feet, Cargo Cube "**3500000**" cubic feet, Max Cargo Dimensions "**600**"X"**600**"X"**120**"inches. The last entries were the max transporter capacities for personnel and liquid cargo. For the LMSR the fields were "**10**" pax and "**100**" forty-two gallon barrels of liquids. At this point S3 asked if the new object should be saved. Again as in building commodities, type "**Y**" for yes to save the new transporter. With the completion of this entry the information to build an LMSR exists in the S3 database and subsequent unit, base and prepositioned ship objects can call for the creation of multiple copies of this transporter class.

### *d. Building SubUnits*

The next step in the S3 Construction Pyramid is building subunits. As a reminder, subunits are used as building blocks for units. The sum of a units subunit inventories becomes part of the units inventory. The demonstration scenario required an Infantry Battalion subunit to be used in later construction of a new Light Infantry Division. To enter a new subunit choose "**N**" from the Scenario Menu. The next screen will be the Subunit Builder menu and as in all builder menus enter "**N**" to begin entering a new object. First, enter the subunit name; for the Infantry Battalion the name is "**INFBN**". Next, enter the type of subunit, "**Land**" for the Infantry Battalion.

Finally, the subunit inventory must be entered. A list of all the commodities built in step one of the Construction Pyramid will be displayed from which any commodities desired for a subunit can be chosen. Once a commodity is entered the stock level is entered, the commodity is identified as either a deployment item or not, and the various consumption factors are entered for each level of combat intensity. For the

Infantry Battalion four commodities were required; Personnel, Dragon missiles, TOW missiles, and M16A1 ammunition. The entry process for the Personnel went as follows: "**Y**" was entered to indicate a new commodity was desired to be added, from the displayed list "**Personnel**" was entered, the stock level was set at "**559**" and "**Y**" was entered to indicate the item should be counted for deployment. The last entry is very important, it drives routine consumption for activated units. The daily consumption rates were entered for high, medium, low and none combat intensity levels as "**5**","**2**","**.1**",and "**0**" respectively. The same figures were entered for the other three commodities in similar fashion. Upon completion of these entries the program will ask if the subunit should be saved and "**Y**" may be entered to make the object permanent.

### *e. Building Bases*

Creating bases in S3 is done by selecting "**B**" from the Scenario Menu. From the Base Building screen select "N" to build a new base. For the demonstration scenario GoldAlpha was added as a new theater base for forces in Country Gold. Following the on-screen prompts for building a new base, the following entries were made to build the GoldAlpha base. First, the scenario builder asks for a base name; "**GoldAlpha**" was entered. Next, the base group must be entered; "**Theater**" was entered for the Country Gold base. Position data is asked for next, the latitude and longitude of GoldAlpha was entered; "**03 00 N**" and "**128 00 E**". In addition, the continental location of the base is asked for from a list of choices, "**S**" was entered to indicate Southeast Asia for GoldAlpha.

The next group of data that S3 asks for is port and transporter data. The builder individually asks whether any of the four port types will be part of the new base. For GoldAlpha "**Y**" for yes was entered for seaport, airport, and truck stop existence. Alternatively, "**N**" for no was entered for railway existence. Next, for the port types answered in the affirmative the base builder will ask for two port constraints, Maximum

Capacity and Maximum Vehicle Size. For GoldAlpha the following entries were made for the seaport; "**10**"for ten berths and "**1000**" for the maximum ship length allowable in the port. Similar entries were made for the airport and truck stop. Next, the transporters that will start the simulation available at this port must be entered. For GoldAlpha, only the truck stop will begin the simulation with transporters available. The following transporters were entered; "**TRUCKCONVOY**" and "**10**" to indicate that ten truck convoys should begin the simulation at the GoldAlpha truck stop. These transporters are not currently treated as commodities that need to be themselves transported into theater. This is a known shortcoming of this program. Currently theater land transportation is treated modeled as host nation support. This is a big assumption and one that will be taken out in future versions of S3.

Next, the starting base inventory must be entered. For GoldAlpha the list is short. Only a small amount of diesel fuel will exist in GoldAlpha's inventory at the beginning of the simulation. When building a base each required commodity must be added to the inventory list individually. The only available commodities will be those commodities that have at one time been entered into the S3 database under the commodity building step. To add a commodity type "**Y**" and enter the commodity name from the list displayed by S3. For the GoldAlpha diesel fuel "**DIESEL**" was entered. Next, the stocking objective , the amount to be on hand at simulation time zero, and the order point for the new commodity will be asked for. In GoldAlpha's case they were entered as follows; "**20000**", "**20000**", and "**18000**" to indicate a stocking objective of twenty thousand barrels of diesel fuel, an on hand amount of the same quantity and an ordering point of eighteen thousand barrels.

At this point the base building steps are finished and the program will ask if the new base should be saved. Type "**Y**" for yes and the new base will be available when the scenario base list is made later on.

### *f. Building Units*

Building new units in S3 begins at the Scenario Menu. Enter "**U**" to move to the Unit Building menu. From this menu enter "**N**" to build a new unit. The next screen to appear will be the Unit Building screen. For the demonstration scenario a new light infantry division was entered. The first step is entering the unit name; "**25THLtInf**" was entered for the Twenty-fifth Light Infantry Division from Fort Shafter, Hawaii.

Next, the Unit Builder will ask if the unit activation should be delayed; "**Y**" was entered for yes concerning the new infantry division. If no had been entered the unit will begin the simulation in an active state. If the unit will be delayed, the builder will prompt the user to give the delay time in days; "**4**" was entered for a four day delay for the light infantry division.

The unit position data is entered next. First, the unit type is called for; "**Land**" was entered for the 25TH Infantry Division. Next, the unit location is entered; "**36 00 N**" and "**128 00 E**" was entered for the deployed position of the 25TH Infantry Division. Following the latitude and longitude entry the unit builder will ask for the continental location of the unit; "**S**" was selected from the displayed list for Southeast Asia for the 25TH.

The next series of questions from the unit builder deal with ports. The unit builder will ask for the port types that will be available at the new unit and the transporters that will begin at those ports. For the 25TH Infantry the only available port entered was a truck stop at which ten truck convoys and five tank trucks were entered for positioning at the simulation start.

After the port information is entered the unit builder next asks for the unit inventory list. This entry is done the same as in the base builder. Though the steps are the same there is one difference. The on hand entry for each commodity is entered as zero for units. The concept of units and their origin base drives this entry. For instance the 25TH Light Infantry Division begin the simulation with an on hand inventory of zero while its origin base, Ft. Shifter, should have an identical inventory list with a full on hand inventory. In other words the division begins the scenario as a commodity held in inventory at the division origin.

The last step in building a new unit is to determine if that unit will be a rapid deployment unit. If that unit is designed to be moved primarily by air and quickly such as a Light Infantry Division, that unit should be entered as a rapid deployment unit by typing "**Y**" for yes. Non-rapid deployment units will have there personnel time delayed to coincide with the arrival of unit equipment. Once all of this information is entered the unit is complete and the user should enter "**Y**" to permanently save the entered data.

### *g. Building Prepositioned Ships*

A prepositioned ship is really just a transporter object which begins the simulation at a predefined position with cargo already on board earmarked for a predefined destination. The prepositioned transporter will not automatically move to this destination, but waits at the entered position until it is activated by the user. To build a new prepositioned ship enter "**P**" at the Scenario Menu screen. This will bring up the Prepo Builder screen. Enter "**N**" to build a new prepo ship.

The first entry required is the ship name; for the demonstration scenario "**Bonneyman**" was entered to build one of the ships in Maritime Prepositioning Squadron Two. The next entry is the ship type of the prepositioned transporter. From the displayed list choose a ship type; for the Bonneyman "**Hauge**" was entered for the Hauge class.

Next, the prepo builder will ask for the ship's destination. For the Bonneyman "**7MEB**" was entered for the Seventh Marine Expeditionary Brigade. After the destination is entered the prepo builder will ask for the names of any intermediate bases that should be used for cargo routing. For the Bonneyman cargo there was one intermediate stop at GoldAlpha, the Country Gold port of debarkation.

After the cargo routing and the destination of the ship is entered the prepo builder will ask for the ship's starting location. For the Bonneyman the ship will begin the simulation at Diego Garcia as a member of MPS 2. The entered location was "**03 00 N**" and "**075 00 E**".

The next step is to enter the prepositioned ships cargo list. This list is made in the same light that the base and unit inventories were created. First, the prepo builder will ask if any cargo will begin aboard the transporter; for the Bonneyman "**Y**" was entered. If yes then the commodity types available for the scenario are listed on the screen. Choose the commodity types desired for the prepositioned ship one by one and enter them. The prepo builder will ask for the amount of each commodity to be held on board the transporter at the beginning of the simulation. Once the cargo has been entered the prepo builder is finished and typing "**Y**" for yes will save the new prepositioned ship in the S3 database.

### *h. Building A Scenario*

The last step in the S3 Construction Pyramid is building the scenario. Again, as in each step of the pyramid, start from the Scenario Menu. Select "**S**" to move to the Scenario Builder menu. Choose "**N**" to create a new scenario. Next, enter the name of the new scenario; "**S3DEMO**" for the demonstration scenario. After giving the scenario a name there are three major steps left to building a scenario, selecting the units and bases, selecting the unit origins, and building the ground transportation networks.

To add bases and units to the scenario choose "**A**" for add and S3 will display a list of all bases and units in the S3 database. Enter the name of a base or unit and it is added to the list. Repeat this step until all of the bases and units required for the scenario have been added.

After the bases and units are entered the unit origins can be specified. Select "**O**" and the Scenario Builder will ask for the origin base for each unit in the scenario. For example, for the 25TH Light Infantry Division "**FtShafter**" was entered as the origin. During a simulation run if the 25TH were activated the logistics manager would look to Ft. Shafter for the 25TH Light Infantry Division commodities.

After the unit origin has been entered for each unit, the rail and truck networks are the last things to be constructed. To build transportation networks from the Scenario Builder menu type "**D**" to enter roads, or "**R**" to enter rail links. The next step is to identify reachable bases or units from each base and unit in the scenario. For example, in the S3DEMO scenario the 3ACR is connected to a theater POD base BlueFoxtrot by road. The following steps would create a road link between the two. From the Scenario Builder "**D**" would be entered to enable the building of a road network. S3 would then run through the list of bases and units with truck stops. At 3ACR, the user would stop and type "**A**" to add a base to the list of bases accessible by truck. Next, "**BlueFoxtrot**" would be entered as a member of this list. To confirm this entry type "**N**" for network and the names of the bases or units currently accessible from 3ACR will appear on the screen. BlueFoxtrot should appear on this list. To enable two way traffic a similar entry must be made at the BlueFoxtrot display to put 3ACR on the list of bases or units accessible from BlueFoxtrot.

To build a complete scenario each base or unit with either a truck stop or railway must have the accessible bases and units entered correctly in this fashion. Failure to enter rail or road links for a land based unit or base without an airport will result in that node being disconnected from the logistics network. If the scenario is satisfactory "**Y**" can be entered to save the scenario in the S3 database. At this point the scenario building process is complete. The work required to input a scenario is demanding and admittedly harder than it should be. Nevertheless, when using a simulation model, the quality of the input data reflects the quality of the model results. The large effort needed to build S3 scenarios is a must if the results are to be taken seriously.

#### *i. Object Modifications*

Once a scenario has been saved any of the objects used to build it can be modified from the Scenario Menu. Simply access the correct builder screen and enter "**E**" for enter modification. Next, all of the objects of that type will be displayed by name on the screen. Enter the name of the specific object that needs to be changed. A new screen will display the current status of the fields for the object named. Using the appropriate menu screen prompts any of these fields can be altered and the new modified object saved.

#### *j. Summary*

This demonstration of scenario building using S3 presents a good picture of what types of information should be gathered beforehand. If this is done, the transportation networks, commodity consumption rates, etc. can easily be added to a scenario. Not fully collecting this data means working backwards in the pyramid. This duplicated effort is a wasteful use of time, especially if the scenario building effort takes place under time pressure. After the seven steps of the S3 Construction Pyramid have been finished the scenario is complete and is ready for use in the execution section of the program. The next section will demonstrate the execution phase of S3 using the S3DEMO scenario.

## C. EXECUTION

The execution section of S3 is where logistics scenarios are wargamed. A description of how to operate S3 using the S3DEMO scenario follows below. This demonstration is an atypical use of S3 in that it merges both seminar and computer wargaming styles. The organization of this discussion will be as follows. First, there is a general overview of the S3DEMO wargame timeline. Second, an introduction to the Execution Menu and Main Execution Display. Third, examples of how to exercise all six of the S3 user control points. Fourth, there is a demonstration of how to access important information about the performance of the logistics network.

### 1. The S3DEMO Wargame Timeline

For this demonstration of the execution section of S3, the S3DEMO wargame event timeline is introduced first as is often done for seminar wargames. This timeline is fictitious and involves the forces outlined in the previous section. The S3DEMO timeline is necessary to provide a reference point for examples in the following sections where timeline events are transformed into actions in S3.

The timeline for the S3DEMO scenario outlines a one hundred and forty day two theater campaign. The first eighty-five days of the wargame deal with events in the Middle East regional conflict. At day eighty-five the Southeast Asia conflict begins and the two run concurrently for ten days. After day ninety-five the Southeast Asia conflict then runs an additional forty-five days. The complete timeline with specific events is listed below in Table III.

**TABLE III - THE S3 DEMO TIMELINE**

| Day | Event |
|---|---|
| +4 | Activate10th Mountain Light Infantry |
| +5 | Activate 25th Light Infantry |
| +7 | Activate 3rd Armored Calvary Regiment |
| | Activate 82nd Airborne Assault Division |
| +26 | Missile Attack On CTF50 |
| +45 | Middle East Ground Forces Put On Alert |
| +65 | Middle East Ground Campaign Begins |
| +78 | Middle East Victory With Pocket Resistance |

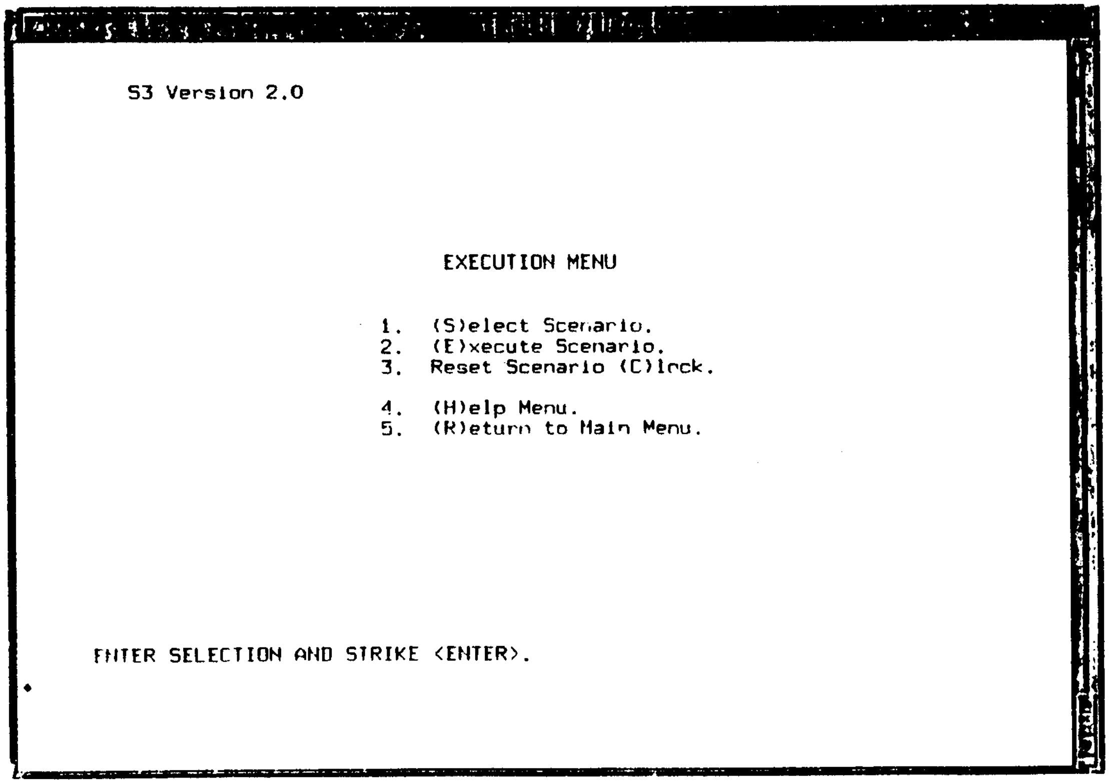
| | |
|---|---|
| +85 | Southeast Asia Forces Alerted |
| | Activate 7th Marine Expeditionary Brigade |
| | Activate Maritime Prepositioning Squadron Two |
| | Activate 101st Airborne Assault Division |
| +90 | Order 25th Light Infantry To Southeast Asia |
| +100 | Southeast Asia Forces Begin Police Action |
| +140 | Democratic Rule Restored to Country Gold |

## 2. The Execution Menu and Display

The primary menu screens used in the execution section of S3 are the Execution Menu and the Main Execution Display(MED). The Execution Menu (Figure 4-5) is where the execution of an S3 simulation begins. Once S3 is in action the Main Execution Display(Figure 4-6) becomes the primary interface screen for the S3 menu system. At the end of every time step the MED appears first. Any further user interaction during execution begins at this screen.

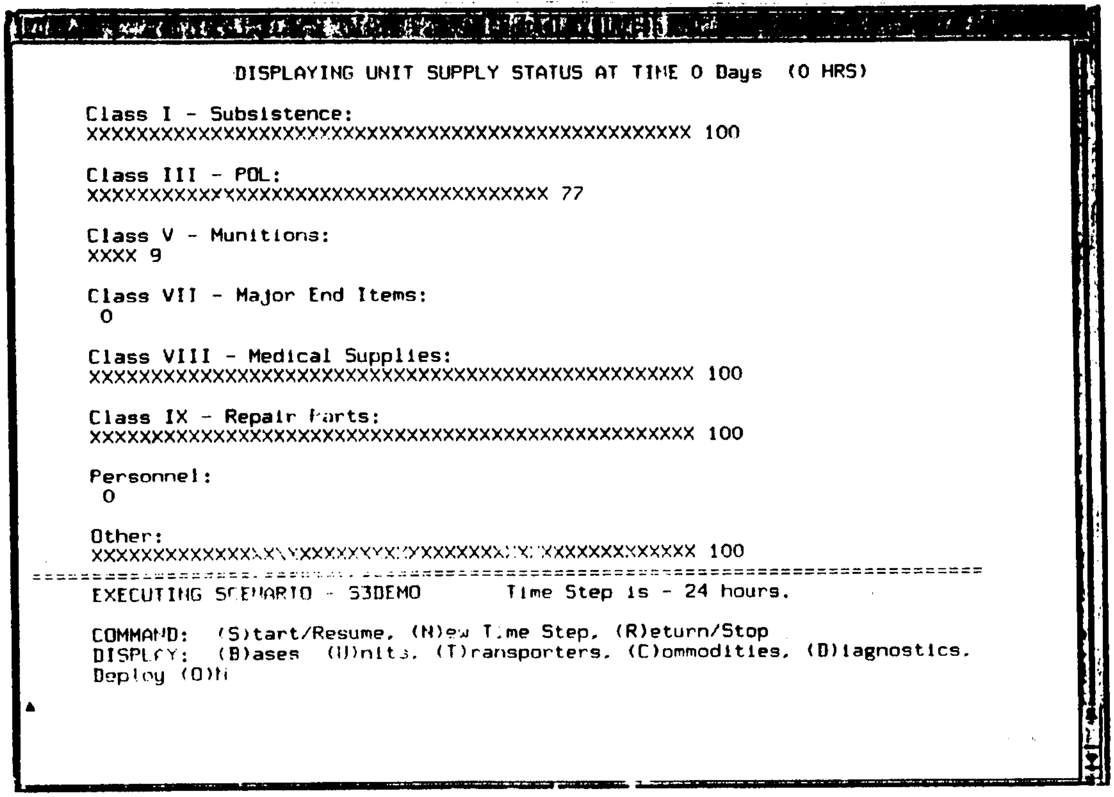
To enter the execution section of S3 choose "**E**" from the Main Menu. The next screen is the Execution Menu. The first selection on this screen is "**S**" for Select Scenario. Making this selection allows the user to choose which scenario from the S3 database will be used for the execution phase. If no choice is made when S3 begins execution it will do so using the first scenario on the list. The second choice on the Execution Menu is "**E**" for execution, which commands the model to begin simulation. Next is a selection to reset the scenario clock. This is available since the scenario clock does not currently reset at the end of each simulation run. By entering "**C**" the clock is reset and a new scenario can be executed from time zero.

Once a scenario has been executed the next screen shown is the Main Execution Display. Two sources of information exist in the MED, the simulation time and a graphic display of the initial scenario unit supply status. Initially, the time should be zero and the supply levels should reflect the current aggregate unit supplies in theater. The initial supply levels are not always zero due to the possibility of forward deployed units beginning a



```
  S3 Version 2.0




                          EXECUTION MENU

                   1.  (S)elect Scenario.
                   2.  (E)xecute Scenario.
                   3.  Reset Scenario (C)lock.

                   4.  (H)elp Menu.
                   5.  (R)eturn to Main Menu.






ENTER SELECTION AND STRIKE <ENTER>.
```

Figure 4-5 Execution Menu Display



```
            DISPLAYING UNIT SUPPLY STATUS AT TIME 0 Days  (0 HRS)

   Class I - Subsistence:
   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 100

   Class III - POL:
   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 77

   Class V - Munitions:
   XXXX 9

   Class VII - Major End Items:
    0

   Class VIII - Medical Supplies:
   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 100

   Class IX - Repair Parts:
   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 100

   Personnel:
    0

   Other:
   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 100
=========================================================================
   EXECUTING SCENARIO - S3DEMO        Time Step is - 24 hours.

   COMMAND:  (S)tart/Resume, (N)ew Time Step, (R)eturn/Stop
   DISPLAY:  (B)ases  (U)nits, (T)ransporters, (C)ommodities, (D)iagnostics.
   Deploy (O)N
```

Figure 4-6 Main Execution Display

scenario in an active status. For example, when running the S3DEMO scenario the initial MED shows several categories with positive deployment levels. These numbers are a result of forward deployed air force and navy units. The supply information shown on the MED is aggregate and only gives the user an initial feel as to how the logistics network is performing at any one time.

This brings up the main purpose of the MED which is to provide the user with control of the simulation and access to specific logistics network performance data. Control of the simulation is accomplished through the six user control points from Chapter II. Specific logistics network performance data is given through the unit and diagnostics module displays. The next two sections deal with these two areas and the paths to take from the MED in order to use them.

### 3. A Demonstration of S3 User Control Points

S3 has six main user control points. To show how to exercise these control points, examples from the wargame timeline in the previous section will be logistically wargamed using S3. The examples follow below:

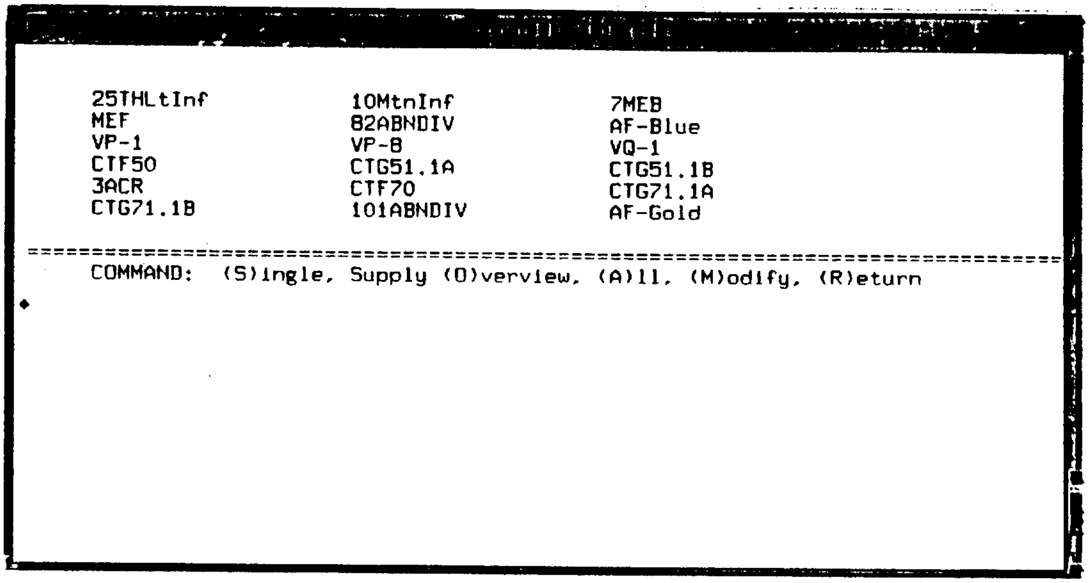
#### *a. Time Steps*

S3 is a discrete event *time step* simulation. As mentioned in Chapter II the time steps selected by the S3 user determine when the other five user control points can be flexed. In the S3DEMO scenario the first events occur on the fourth day of the wargame. In order to enter these events the user needs to time step forward to the fourth day. To do so enter "**N**" from the Main Execution Display to allow a change in the size of the simulation time step. Note that the time step is set at twenty-four hours at the beginning of each simulation run. S3 will ask for the new time step in hours. Enter "**96**" for ninety-six hours (or four days). The new time step will be displayed in the lower right hand corner of the Main Execution Display. To advance the simulation one increment using the new time step, currently ninety-six hours, enter "**S**" for start simulation. The simulation will then step forward ninety-six hours completing all logistics events scheduled in that interval

The user can change the time step at any simulation stop from the MED. It is important to note that the size of the chosen time step controls user access to S3, the larger the time step the more control that is left to the computer.

### *b. Unit Deployment*

Unit deployment times are preset by the user when the scenario is built. Once the simulation begins the user can only move the deployment time forward. For example, in the S3DEMO timeline the 25TH Light Infantry Division (25THLtInf) deploys on day C+5. Suppose that it is desired to speed this activation time up one day to C+4. To do so manually, activate the 25THLtInf at time ninety-six hours. Begin at the Main Execution Display at the ninety-sixth hour (where the last section stopped). Type "U" to move to the Unit Menu Screen (Fig 4-7). From the Unit Menu Screen select "M" for modify. Next, choosing from the provided list, enter the name of the unit to be modified, "**25THLtInf**".



```
    25THLtInf            10MtnInf             7MEB
    MEF                  82ABNDIV             AF-Blue
    VP-1                 VP-8                 VQ-1
    CTF50                CTG51.1A             CTG51.1B
    3ACR                 CTF70                CTG71.1A
    CTG71.1B             101ABNDIV            AF-Gold

==============================================================================
    COMMAND:  (S)ingle, Supply (O)verview, (A)ll, (M)odify, (R)eturn
```

**Figure 4-7 Unit Menu**

Next, a list of unit data areas that can be modified will be displayed. Enter "**S**" to change the unit status. A list of fields that can be changed to affect the overall unit status will be displayed next. Enter "**A**" to activate the unit. No further action is required, the 25THLtInf will begin activation as soon as the simulation begins the next time step. Note, if no action is taken to manually activate a unit it will automatically deploy at the preset deployment time.

### *c. Combat Intensity*

Unit combat intensity can be changed at the end of any time step. In the S3DEMO wargame several events resulted in unit combat intensity changes. For example, the missile attack on day twenty-six commenced hostilities in the Middle East theater and placed the surface ships there under a high combat intensity level. To reflect a combat intensity change in S3, the following steps are required. To raise the combat intensity for CTF50, the main surface ship force in the Middle East theater of the S3DEMO, follow the previous section steps to get to CTF 50's status modifications entry point. Next, enter "**I**" to change the unit combat intensity level. The available choices are high, medium, low and none. Enter the appropriate selection, "**H**" for high in this example, and the new combat intensity level will be displayed on the unit display screen. Upon resumption of the simulation, CTF50 will begin consuming at a high combat intensity level.

Combat intensity drives consumption, and the sum of unit consumption figures produces the network sustainment requirements. It is important to reflect correctly the unit combat intensity during simulation time steps. The quality of user control over combat intensity, especially for long campaigns, will determine the quality of the S3 model sustainment results.

### *d. Stock Levels*

The ability to control unit and base stock levels is an important tool in S3. This is the one means to "***push***" commodities into theater if the user desires to build up supplies at a quicker rate than unit consumption is "***pulling***" them in. To do so the user can raise the stocking objectives of selected units or theater bases. The effectiveness of this method is limited by the overall network supply level of the commodity in question.

As an example, suppose that wargame usage of TOW anti-tank missiles is higher than was anticipated when the logistics network was built. The user has two options at this point. Option one, allow additional TOW missiles to be "***pulled***" into theater by unit consumption resulting in several small shipments and a losing game of catch up supply resulting in a long term shortage of this commodity. Option two, raise the commodity stocking objective "***pushing***" additional TOW into theater in one large shipment with only a short term shortage. Both of these results are conditional upon ample Conus supplies of TOW missiles. Taking this as a given, however, and it becomes obvious that the best choice for the user is to minimize the shortage time by raising the theater stocking objective of the TOW missile immediately in anticipation of the future shortage.

The Middle East theater stock point for TOW missiles is BlueFoxtrot in the S3DEMO scenario. To raise the BlueFoxtrot stocking objective, go to the base display and enter "**I**" to change the base inventory. A new line of selections will appear, providing two choices; enter a new commodity or edit an existing commodity. Enter "**E**" to edit a commodity. Next, enter the commodity name, "**TOW**" for the tow missile. On the next line choices will appear allowing the user to alter the base data for the selected commodity. Select "**S**" to change the base stocking objective. The old stocking objective was two thousand, suppose that new usage figures indicate three thousand is needed, enter "**3000**" to raise the base stocking objective. The new higher stocking objective will "**push**" additional missiles into theater once simulation begins. This example is one of a multitude of possibilities for the user to change the rate of push and pull for theater supplies using stock control at the base and unit level.

Stock control methods are most important for both seminar and computer wargaming. If the theater stock levels of an item are inadequate for a wargame scenario, stock control can be used to "***pull***" or "***push***" in additional supplies. The proper use of this tool could prove to have a major logistic impact on unit readiness in an ENWGS wargame. For strategic mobility questions, the failure to push adequate supplies forward early on may result in a logistics deficit later that may not be overcome.

### *e. Port Existence*

Another simply exercised control point is port existence. At any time during model execution the ports of any base or unit can be modified. Changes of this sort will affect the logistics flow rate for the scenario. In the S3DEMO the GoldAlpha seaport, in the Southeast Asia conflict, begins the simulation with only one pier available. After the build up of forces in this conflict begins at day eighty-five, suppose that the wargame calls for port improvements. Say that these improvements come at a rate of one pier space every additional week after day eighty-five. To add these new port unloading spots (piers for a seaport) move to the Unit Display Menu and enter "**M**" for modifications. Next, enter the name of the unit whose port is desired changed; "**GoldAlpha**" in this case. From the next group of choices enter "**P**" to alter GoldAlpha ports. From the following list of selections choose "**S**" for seaport. The next screen will display the choices of alterable fields for the port. These fields are the number of transporters in port, the port capacity (or number of unloading spots), the max size transporter allowed, and most importantly the very existence of the port. For the S3DEMO example, enter "**C**" for capacity and "**2**" to indicate a new port capacity to unload two ships at once. To add new capacity every week, simply time step forward seven days and repeat the process, entering "**3**" for the new port capacity. To completely shut down the GoldAlpha seaport due to a wargame event, such as an enemy raid from the same screen, enter "**E**" to toggle port existence from true to false. Until changed again, the port will be closed to sea transportation.

The capability to manage port capacities gives the user control over the port throughput. When modeling wargame scenarios in underdeveloped countries with small ports, this feature can be a major factor in the overall rate of surge and sustainment for a logistics network.

### *f. Transporter Existence*

Transporter existence is also controllable in S3. The creation of new transporters and the destruction of existing transporters are both possible. Here it is appropriate to mention one current shortcoming of S3, in that it does not model the in-between stages of a transporter's life, specifically transporter breakdown and repair. It is possible to mimic these events using creation and destruction, but it is difficult to accomplish. Next, is an explanation of how to use S3 to change the available transportation during a simulation run.

To manage the network transporter pool select "**T**" from the Main Execution Display. Next, the screen will display a list of all transporters in the scenario. At the end of this list will be a list of choices to alter the numbers of transporters available. Three actions will alter the number of available transporters. Prepositioned ships can be activated, new transporters can be added, and existing transporters can be destroyed. Each method is explored separately below.

A prepositioned ship can be activated at any time, but only once during simulation. Once activated the transporter is effectively added to the transportation pool. To do this enter "**P**" for prepositioned ship from the transporter display. Next, enter "**A**" to activate and a list of the scenario's prepositioned ships will appear. To activate one of these ships enter the name, for example "**Bonneyman**" is a prepositioned ship in the S3DEMO scenario. The ship will activate when the next time step begins and it will proceed to the unit for which the prepostioned cargo had been prerouted. After unloading at this unit the ship will enter the regular transportation pool.

To create and destroy transporters enter "**M**" for modify transporters. Two choices will given, add or destroy. Enter "**A**" to add a new transporter. Next, S3 will ask for the transporter type name and how many should be added. For example to enter ten Hauge class ships type "**Hauge**" and "**10**". Last, the simulation will ask where the transporter should be added. Enter the name of the base or unit, and the new transporters will become available at the specified port once the simulation begins again. To destroy a transporter, enter "**D**" rather than "**A**". Next, give the transporter type and the vehicle identification number for the transporter to be destroyed. This number can be found in the transporter list displayed when entering this section of S3. To destroy Truck Convoy 14 enter "**TRUCKCONVOY**" and "**14**". The simulation will ask for a simple yes or no confirmation of this action. Enter "**Y**" to confirm and the named transporter and any cargo onboard will be destroyed. The destination unit for any destroyed cargo will show the loss with lower on order figures for the lost commodities. New orders will be placed during the next unit consumption period.

### *g. Summary*

The six user control points of S3 each individually impact the logistics network effectiveness for a given scenario. Together their actions can drastically alter the overall surge and sustainment picture. Using the control points is easy, but tedious. Unfortunately, the micro level control exercised with most of these control points is necessary for computer wargaming, but is extra work for strategic questions that may be addressed with S3 in seminar wargaming. Eventually, the menu system for S3 will be replaced with a windows environment that will make many of these control points easier to implement.

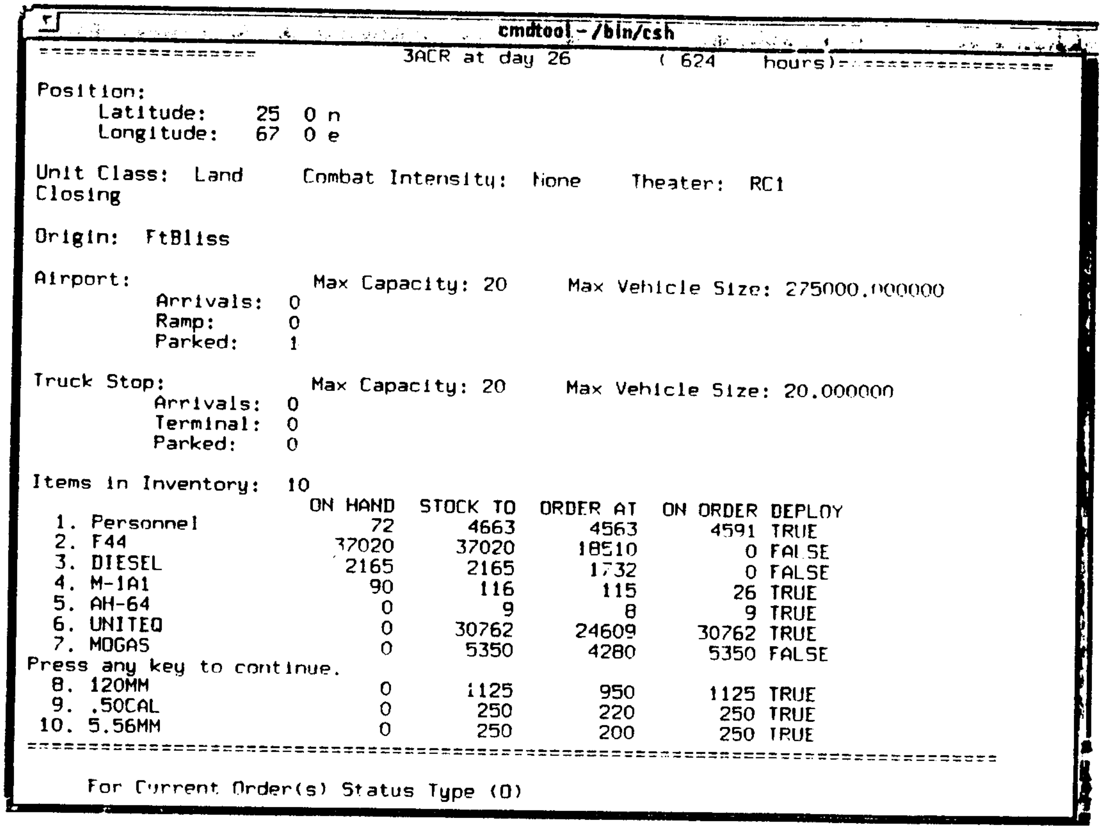
### 4. Logistics Network Information

Before the user can implement any of the control point tools mentioned in the previous section he or she must first make a decision about ***how*** the network should be altered. This decision must be built upon an informed picture of the current status of the logistics network. S3 provides several sources of information that can be used to build this picture. Four specific areas facilitate this data gathering; unit and base status displays, port throughput data, transporter usage figures, and closure data. Together they give sufficient information to provide logistics network insight. The user can use this picture to decide where the logistics network's weakpoints are and where improvements might be made to improve network performance. Descriptions of how to locate and use this information follow below.

#### *a. Unit and Base Status Information*

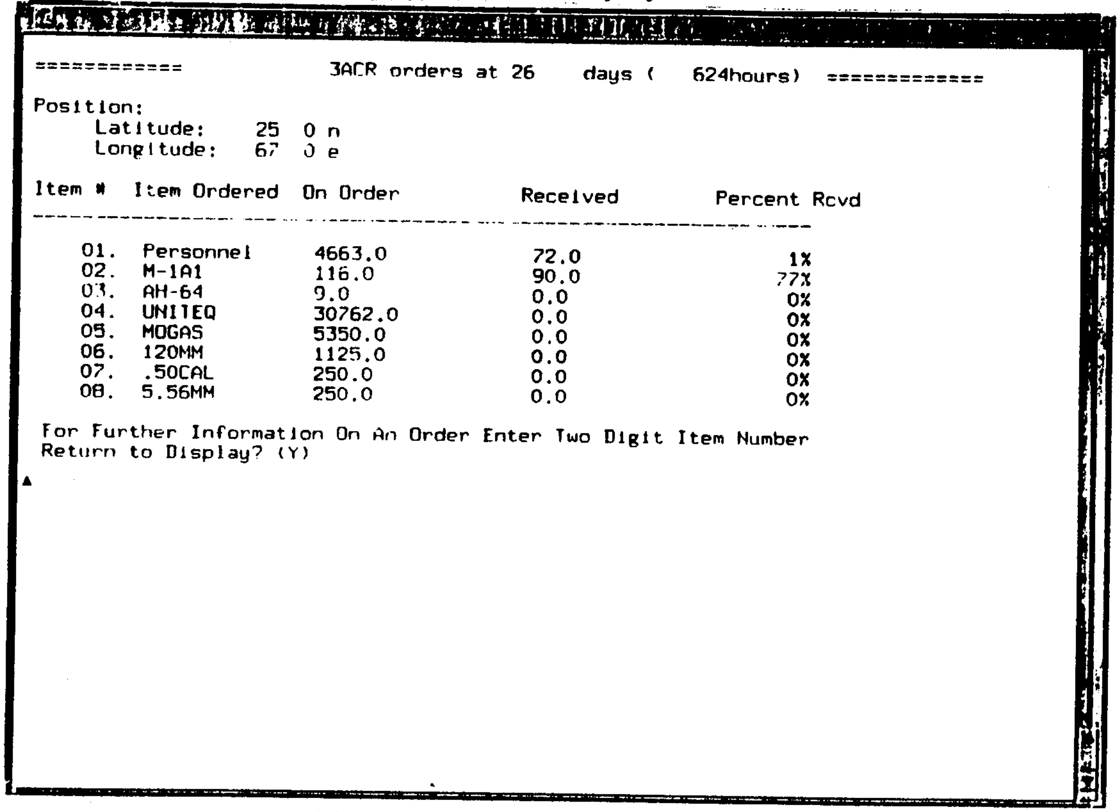
At the micro level, the most important information available to the S3 user is the current status of a base or unit. The Unit and Base Displays provide this data. Everything from the unit combat intensity level to the current on-hand inventory is viewable from these data screens at the end of any time-step. In addition to the Unit and Base displays two secondary screens provide Order Status reports. The current status of any existing order can be found using these screens.

As an example to explore the unit and base display, suppose the 3rd Armored Calvary Regiment(3ACR) is being considered for action in an ENWGS wargame at day twenty-six of the S3DEMO scenario. Assume the user wishes to know the logistics status of the unit to confirm whether it is ready for action. The data sought in this inquiry can be found in the Unit Display. To reach this screen begin at the Main Execution Display. From that menu "**B**" or "**U**" can be entered to view bases or units respectfully (See



```
==================              3ACR at day 26          ( 624      hours)=================
Position:
     Latitude:    25  0 n
     Longitude:   67  0 e

Unit Class:  Land      Combat Intensity:  None      Theater:  RC1
Closing

Origin:  FtBliss

Airport:                    Max Capacity: 20      Max Vehicle Size: 275000.000000
          Arrivals:  0
          Ramp:      0
          Parked:    1

Truck Stop:                 Max Capacity: 20      Max Vehicle Size: 20.000000
          Arrivals:  0
          Terminal:  0
          Parked:    0

Items in Inventory:  10
                          ON HAND  STOCK TO  ORDER AT  ON ORDER DEPLOY
  1. Personnel                 72      4663      4563      4591 TRUE
  2. F44                    37020     37020     18510         0 FALSE
  3. DIESEL                  2165      2165      1732         0 FALSE
  4. M-1A1                     90       116       115        26 TRUE
  5. AH-64                      0         9         8         9 TRUE
  6. UNITEQ                     0     30762     24609     30762 TRUE
  7. MOGAS                      0      5350      4280      5350 FALSE
Press any key to continue.
  8. 120MM                      0      1125       950      1125 TRUE
  9. .50CAL                     0       250       220       250 TRUE
 10. 5.56MM                     0       250       200       250 TRUE
=================================================================================

     For Current Order(s) Status Type (O)
```

Figure 4-8 Unit Display for 3ACR



```
============            3ACR orders at 26        days (    624hours)    ============

Position:
     Latitude:    25  0 n
     Longitude:   67  0 e

Item #   Item Ordered   On Order           Received          Percent Rcvd
---------------------------------------------------------------------------

     01.   Personnel      4663.0             72.0                     1%
     02.   M-1A1          116.0              90.0                    77%
     03.   AH-64          9.0                0.0                      0%
     04.   UNITEQ         30762.0            0.0                      0%
     05.   MOGAS          5350.0             0.0                      0%
     06.   120MM          1125.0             0.0                      0%
     07.   .50CAL         250.0              0.0                      0%
     08.   5.56MM         250.0              0.0                      0%

 For Further Information On An Order Enter Two Digit Item Number
 Return to Display? (Y)
```

Figure 4-9 Orders Display for 3ACR

Figure 4-6). To view the 3ACR enter "**U**". Next, enter "**S**' to view a single unit. A listing of all units in the scenario will be displayed. Make the appropriate selection from the given list; enter "**3ACR**" for this example. The next screen shown is the Unit Display, Figure 4-8 shows the 3rd Armored Calvary logistics picture.At the top of the screen is the current simulation time in days and hours, twenty-six days or 624 hours in Figure 4-8. Next, the current location of the unit or base is displayed by latitude and longitude; "25 00 N" by "67 00 E" for the 3ACR. A second position indicator is the theater location. For the 3ACR this field is "RC1" for regional conflict one. This is one of three choices; RC1, RC2, and Neither. The assignment of these theater numbers is leftto the user discretion, for the S3DEMO scenario RC1 is the Middle East theater and RC2 is Southeast Asia.

The next group of information is unique to the unit display. The next four fields indicating unit status are not part of the base display. They are combat intensity, unit closing, unit origin, and the unit class data fields. For the 3ACR, the combat intensity level is now "None" ,the unit is still "Closing", the unit origin is "Ft. Bliss", and the unit class is "Land". The Base display is identical to the Unit's except that these fields are left out.

The port status for the displayed unit or base is shown next. For example, look at the data for the 3ACR truck stop in Figure 4-8. The maximum size truck convoy receivable at each of the twenty unloading spots is shown to be ten trucks. The truck convoy count in each of the three port queues is shown. Currently, there are no truck convoys waiting to be unloaded, unloading, or parked.

The last information available on the Unit or Base Display is the current commodities inventory. For the 3ACR six items make up the unit inventory. Each commodity is listed with the current on-hand, stocking objective, ordering point, the number on order and whether the item is a deployment item. For example, observe the

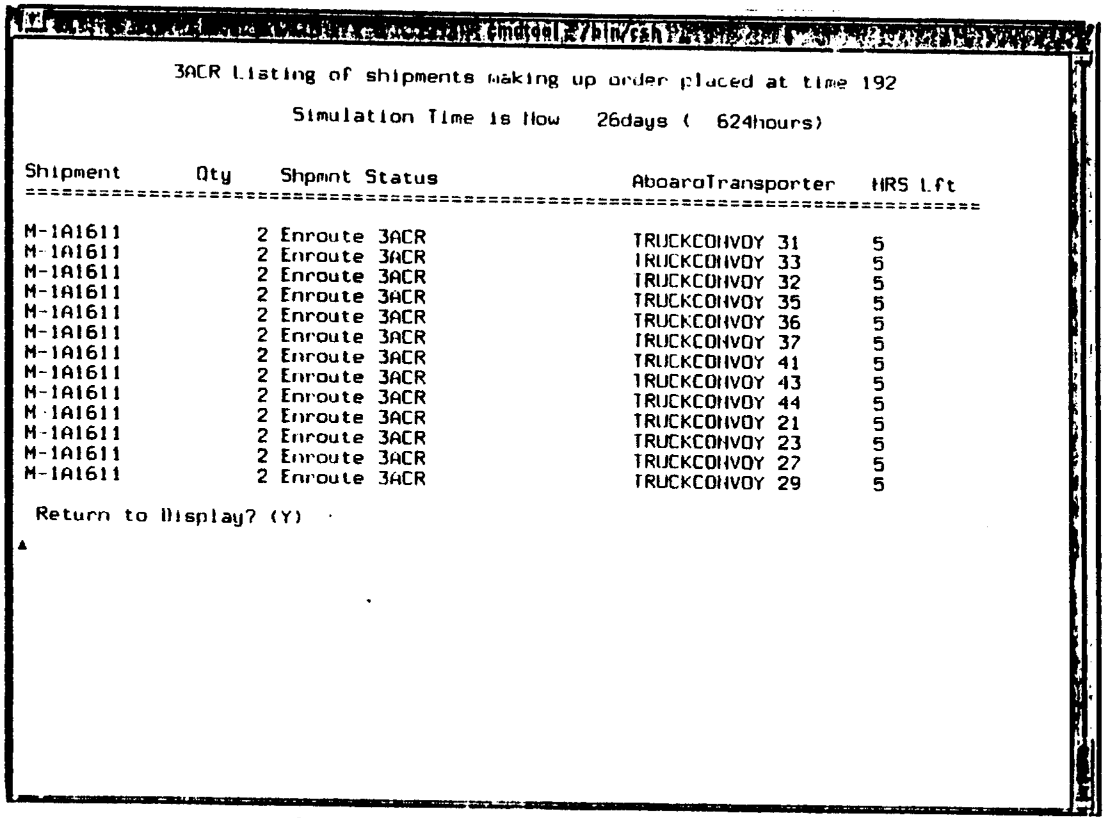
M-1A1 tank data in Figure 4-8. The current on-hand inventory of tanks is 90 while 26 are shown to be still on-order to meet the stocking objective of 116. Furthermore, it is stated to be "True" that this item is a deployment item. If this field had been "False" this would indicate that the item is not deployable, and should not initially be sought from the unit origin, but instead should come from the closest supplier to the theater. Typically, this is the case for fuel and subsistence.

This information is given for each item in the inventory list. From this data the user has a good idea of the current logistics status of the unit or base in question. In regard to the combat readiness of the 3ACR, the user can easily see that the unit is not prepared. Tanks and fuel are the only inventory items currently in theater. Ammunition, helicopters, unit equipment, and personnel have not even begun to arrive. The fact that tanks arrived at this unit ahead of personnel should be troublesome to the reader. This is a result of the personnel arrival in theater being delayed by the program to coincide with the arrival of unit equipment. This timing is currently based on a ship speed for the unit cargo of twenty knots. Unfortunately, the tanks for this unit arrived via a high speed ship capable of twenty-six knots. This explains, but does not justify this discrepancy.

The next step for the user is to determine when this unit *will* be ready. To do this the location of the current commodities on order must be found. At the bottom of the inventory list is one of two statements; "For Current Order Status Type O" or "There Are Currently No Orders Outstanding". For the 3ACR, at C+26, there are outstanding orders; to view these type "**O**". The next screen shown (Figure 4-9) is an aggregate listing of all current outstanding commodity orders for the 3ACR. The percentage received of each order is given, for example seventy-seven percent of the M-1A1 tanks are in at the 3ACR. To view the status of a specific order start from this screen and enter the number corresponding to the list position of each commodity.

For example, type "**03**" to view the 3ACR order status of M-1A1 tanks. It is important to note that due to the large number of shipments that can make up one order, two digits were required to display the order numbers on this menu screen. It is imperative that for single digit orders a zero is typed along with the digit as in "**03**" and not "**3**".

Next, the Order Status Screen (Figure 4-10) displays with a listing of all shipments that make up the order in question. For the M-1A1 tank order the order size of 116 tanks was filled by several smaller shipments. Each shipment is labeled by the commodity name plus the creation time for the current shipment. For example,the shipment name M-1A1611 stands for M-1A1 tanks in a shipment created at time 611 hours. Each shipment has one of severa! status; unloading, loading, enroute, prepositioned, or on backorder. A sixth status is reserved for use with one commodity, personnel; time delayed order. This is a feature coordinating the arrival of troops with



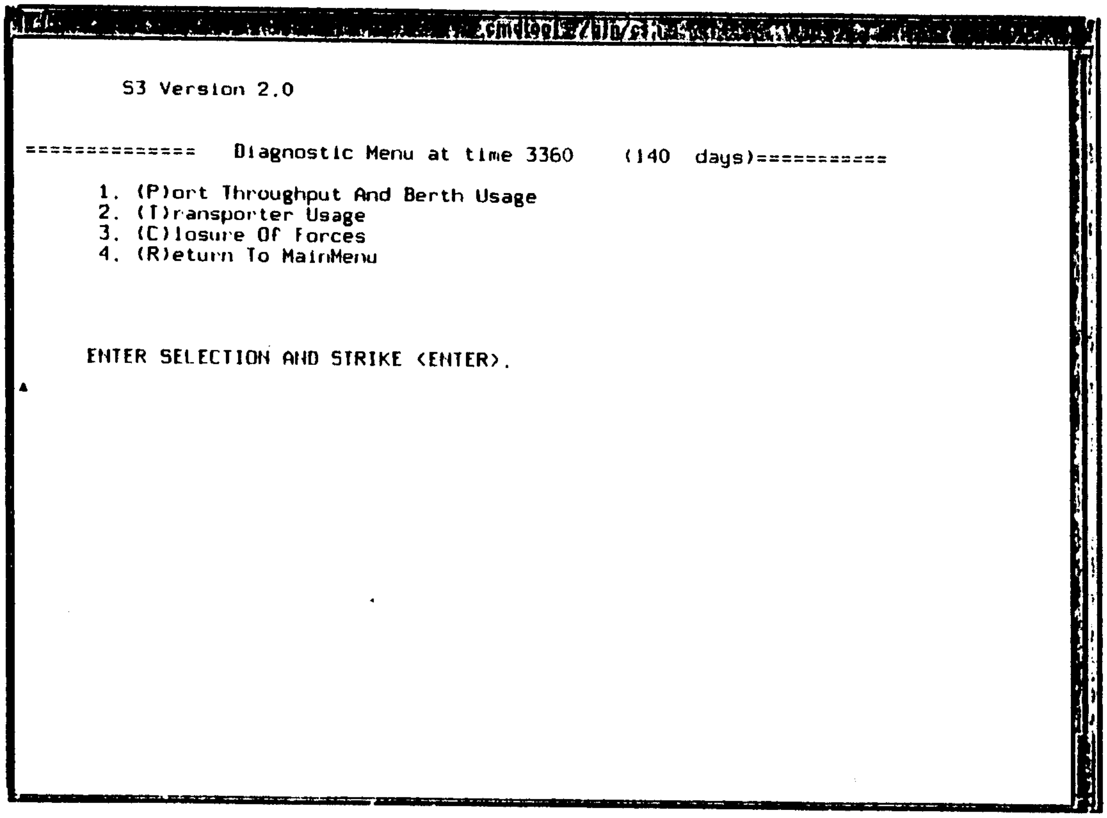
```
              3ACR Listing of shipments making up order placed at time 192

                          Simulation Time is Now   26days (  624hours)

Shipment       Qty    Shpmnt Status                        AboaroTransporter   HRS Lft
==================================================================================

M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 31      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 33      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 32      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 35      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 36      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 37      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 41      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 43      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 44      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 21      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 23      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 27      5
M-1A1611           2 Enroute 3ACR                         TRUCKCONVOY 29      5

 Return to Display? (Y)
```

**Figure 4-10 Order Shipments for 3ACR**

equipment, preventing the model from sending troops into theater weeks before the rest of a unit.

The shipments making up the remaining twenty-six tanks on order are shown in Figure 4-10. The first of these holds two tanks. These tanks are enroute to the BlueFoxtrot base aboard truck convoy number thirty-one. This transporter will arrive at BlueFoxtrot in five hours as shown at the far right of the screen. It is important to note that this is *not* necessarily the same as five hours from the 3ACR. The vehicle unloading time is not included. The present version of S3 only gives the time remaining for in the current transporter leg. For example if the tanks were aboard a ship enroute from Corpus Christi, TX to BlueFoxtrot, CountryBlue, the time shown in this display for any shipments on board would indicate the remaining time in the ships leg to BlueFoxtrot. It would not include the unloading times at BlueFoxtrot and the 3ACR or the travel time to the 3ACR. The actions taken to view the M1A-1 tank orders may be repeated to view the status of other orders given in the unit orders listing in Figure 4-9.

Unit or Base Display data is critical for the user's understanding of what the logistics network is doing. Simply viewing the status of a unit or base provides an opportunity for logistics insights. For example, a desperately needed order may turn out to be waiting for transport at its point of embarkation because there are no ships available to carry it overseas. Perhaps a unit is consistently running out of a particular commodity, the order status may show a large number of small orders and a non-increasing inventory of the same commodity telling the user the theater stocking objectives are too low. There are several large scale logistics problems that can be diagnosed by checking the health of one unit. In summary, the order visibility and unit and base status information available in the Unit and Base Display is sufficient to provide the user with a picture of how well the logistics network is supporting the units in theater.



```
      S3 Version 2.0

=============   Diagnostic Menu at time 3360    (140  days)==========

      1. (P)ort Throughput And Berth Usage
      2. (T)ransporter Usage
      3. (C)losure Of Forces
      4. (R)eturn To MainMenu



     ENTER SELECTION AND STRIKE <ENTER>.
```

**Figure 4-11 Diagnostics Menu**

*b. Port Throughput*

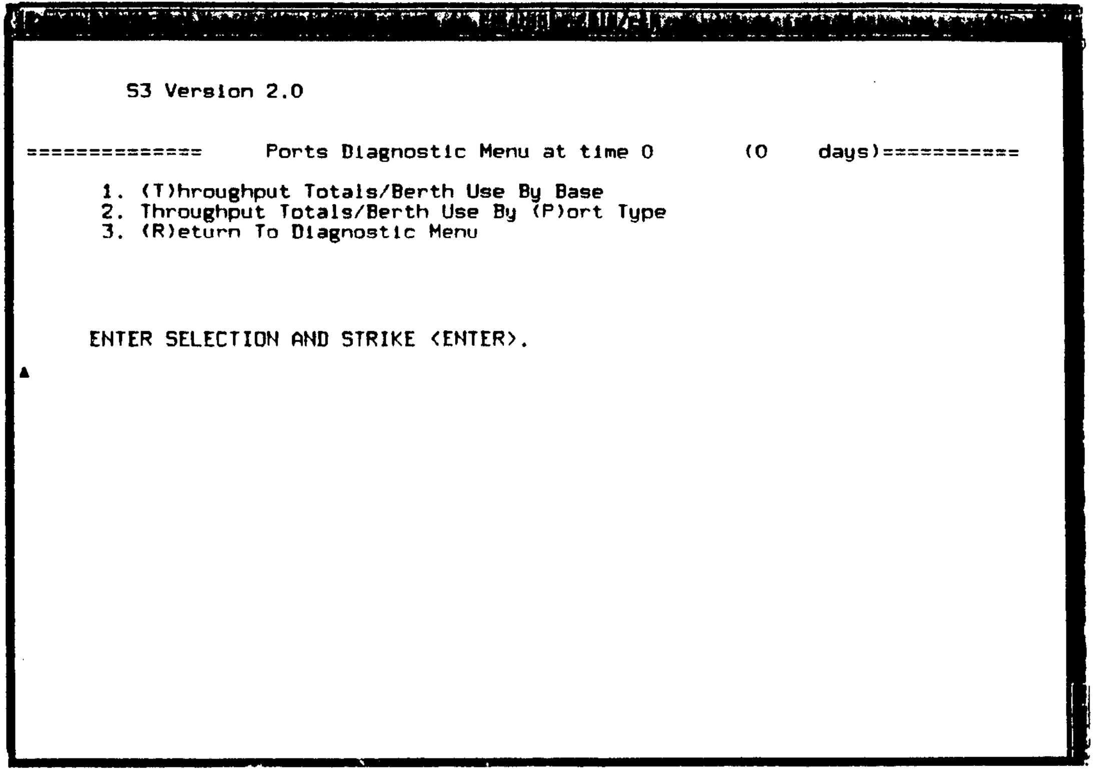
This section and the next two after it deal with subsections of the Diagnostics Module of S3. This module is new to S3 for Version 2.0, and it deals only with logistics network performance data. To get to the Diagnostics Menu select "**D**" from the Main Execution Display. Figure 4-11 shows the three diagnostic areas, port throughput, transporter usage, and closure information.

Suppose that a seminar wargame is being run with the assistance of S3. The user wishes to find some insight into the effectiveness of different logistics scenarios provided by different strategies or force makeups. For this Diagnostic Module discussion assume that the user is currently testing the S3DEMO scenario and the simulation time is C+140, the end of the S3DEMO scenario timeline. Using the Diagnostics Module the user can check different aspects of the scenario with each of the three diagnostic areas.

For this first section, the port throughput data area of the Diagnostics Module is examined. To enter the port throughput area, select "**P** " from the Diagnostics Menu. The Port Throughput Menu (Figure 4-12) will display with a list of choices. Throughput data for each base can be viewed by a single port type or for all ports at a given base. Select "**T**" to view overall base throughput. Next, S3 will ask which bases will be viewed. The selections include viewing a single base, all bases, all POE bases or all POD bases. Enter "**S**" to view a specific base; next enter the base name. For an example, enter "**BlueFoxtrot**" to observe a complete port throughput data display. Figure 4-13 shows BlueFoxtrot at time C+140. The data shown on this screen gives insight into the overall performance of each BlueFoxtrot port.

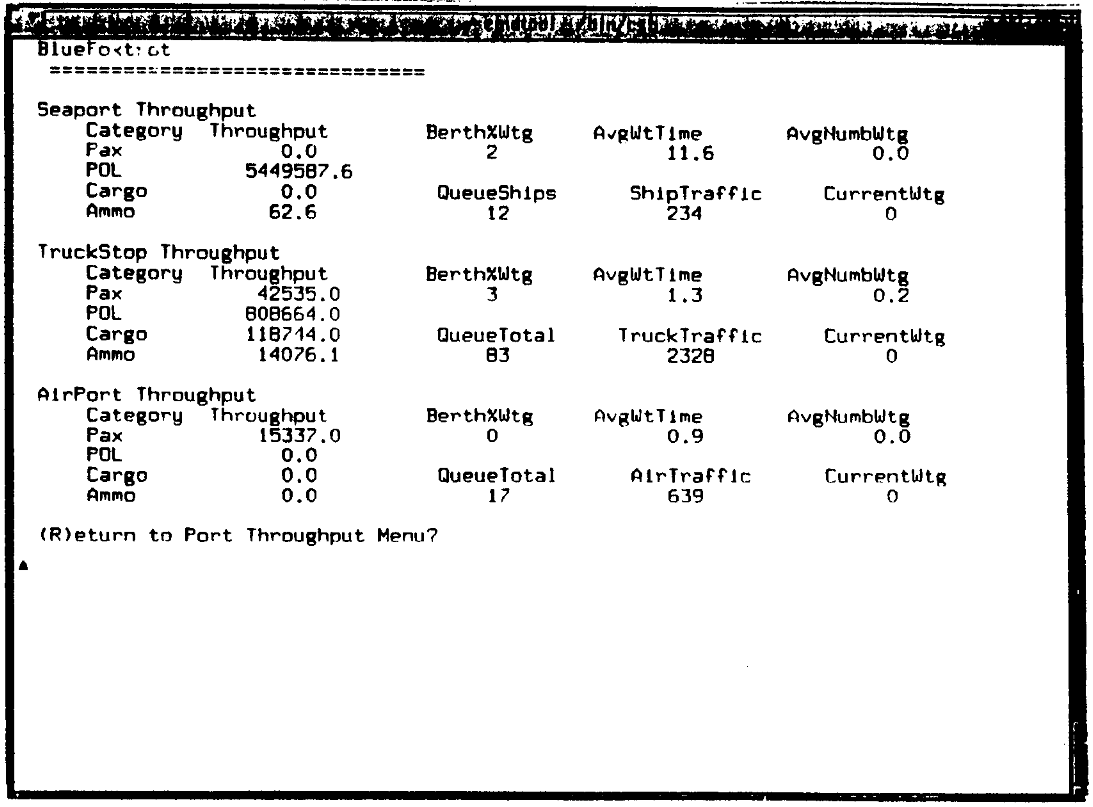
The port display is identical for all port types. For an example look at the truck stop data for the BlueFoxtrot base in Figure 4-13. The current display does not list the current simulation time, but this figure was made at time C+140. The first column of numbers in the display shows data about the total outgoing cargo from the BlueFoxtrot truck stop. These figures are the port throughput, only outgoing cargo is counted. For instance 42,535 personnel traveled through the BlueFoxtrot port enroute to theater units. In addition to the throughput data there are several statistics on the port performance. Working from the top row to the bottom and left to right the next sentences will describe each field shown.

The first field is the percentage of time transporters are waiting for berths out of the entire simulation time; for the BlueFoxtrot truck stop this has been three percent over the last 140 days. Second, is the average waiting time for the trucks; one and one third hours at BlueFoxtrot. Next, the average number of transporters waiting is shown; about two tenths of a transporter at any one time is waiting for an unloading space at BlueFoxtrot. The first item in the second row is the current number of trucks in the port; eighty-three at BlueFoxtrot. The next figure shows the total number of truck



```
       S3 Version 2.0

=============       Ports Diagnostic Menu at time 0       (0     days)==========

      1. (T)hroughput Totals/Berth Use By Base
      2. Throughput Totals/Berth Use By (P)ort Type
      3. (R)eturn To Diagnostic Menu



     ENTER SELECTION AND STRIKE <ENTER>.
▲
```

Figure 4-12 Port Throughput Menu



```
BlueFo<t:ct
 ==============================

Seaport Throughput
    Category  Throughput          BerthXWtg     AvgWtTime        AvgNumbWtg
    Pax             0.0              2              11.6               0.0
    POL       5449587.6
    Cargo           0.0           QueueShips     ShipTraffic      CurrentWtg
    Ammo           62.6              12             234                0

TruckStop Throughput
    Category  Throughput          BerthXWtg     AvgWtTime        AvgNumbWtg
    Pax         42535.0              3               1.3               0.2
    POL        808664.0
    Cargo      118714.0           QueueTotal    TruckTraffic      CurrentWtg
    Ammo        14076.1              83            2328                0

AirPort Throughput
    Category  Throughput          BerthXWtg     AvgWtTime        AvgNumbWtg
    Pax         15337.0              0               0.9               0.0
    POL             0.0
    Cargo           0.0           QueueTotal     AirTraffic       CurrentWtg
    Ammo            0.0              17             639                0

(R)eturn to Port Throughput Menu?
▲
```

Figure 4-13 Blue Foxtrot Port Display

convoys which passed through BlueFoxtrot up through day C+140, a sum of 2328. The last piece of information is the current number of transporters waiting for berths; zero for BlueFoxtrot.

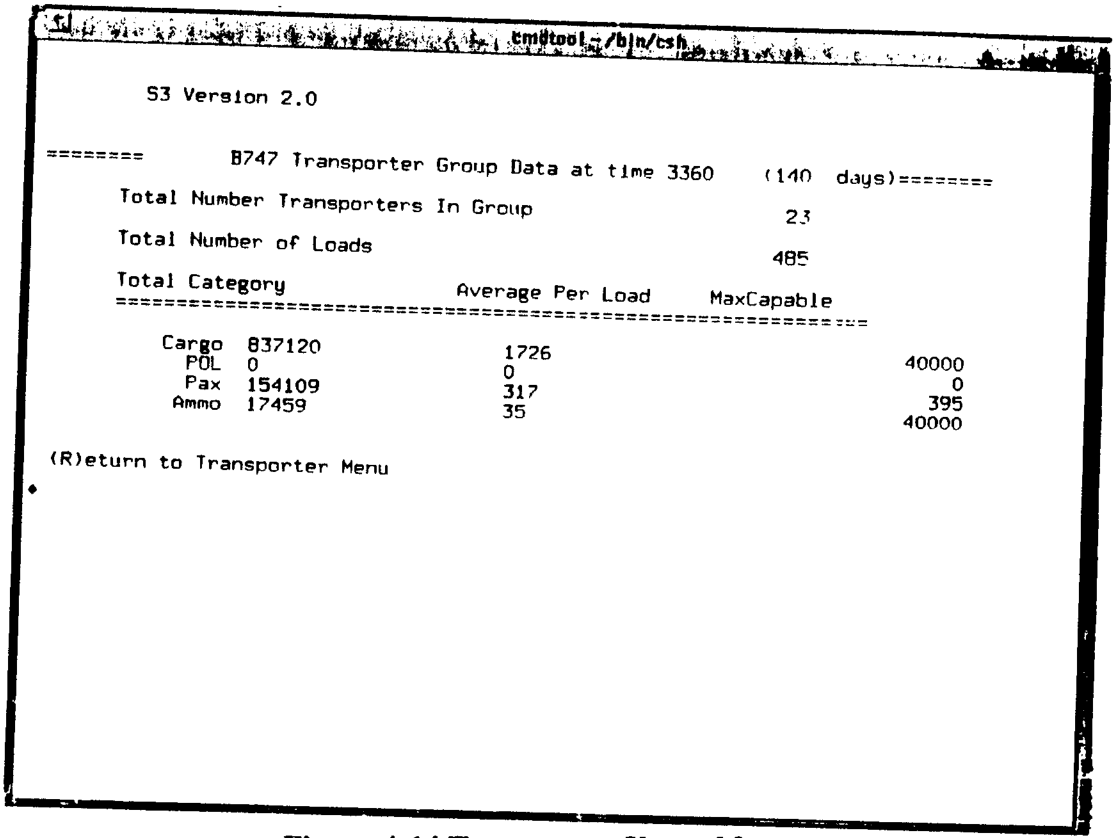
The data currently provided in the port throughput display is not complete and some of it is misleading. The average time that transporters are waiting for unloading spots is equal to the total time transporters are waiting over the total simulation time. A large portion of that simulation time has little traffic, the three percent figure given in Figure 4-12 is really indicative of a higher percentage of waiting transporters during the busy periods for the BlueFoxtrot truck stop. Port throughput should also show the figures coming into a base, as well as those going out, in order to give the total picture of thecargo movement at a particular port. In addition the average number waiting is currently only shown with one significant digit and often rounds down to zero over time.

In summary, the port throughput data displayed is sufficient to provide the user a sense of how well the current port set-up for a given base is handling the traffic produced by the logistics network. The user can derive shortcomings in the current port or network setup. Two possible fixes for these problems are to provide an additional base to lighten the load on a burdened port or to generate additional unloading spaces to minimize transporter backups. For example, the BlueFoxtrot base examined in this section appears to have served its customers fairly well. The only sign of a problem is the truck stop. A user might want to consider adding additional unloading spots to reduce the waiting times, or opening another POE to serve some of the theater units. Thus the port diagnostics tool is appropriate to identify network chokepoints for surge and sustainment.

### c. *Transporter Usage*

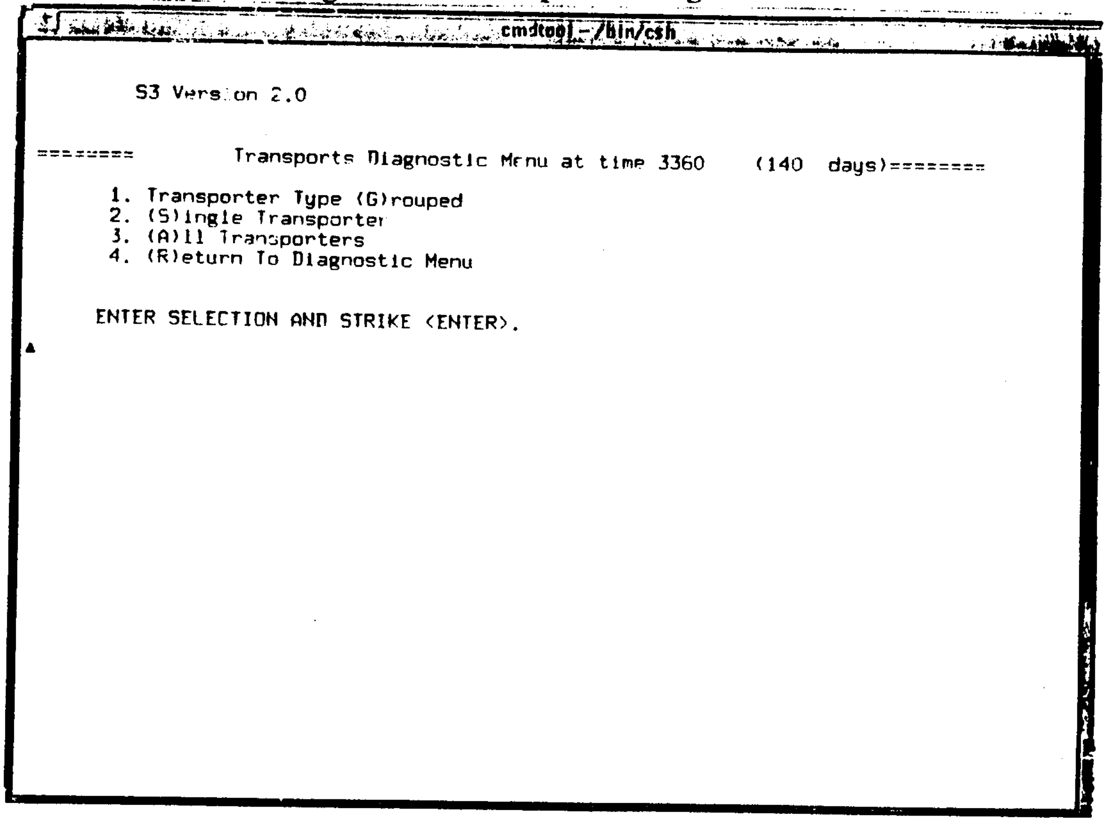
The second display of diagnostic data is for transporter usage. From the Diagnostics Menu select "**T**" for the transportater usage. Next, the Transporter Menu will display (Figure 4-14) with a listing of choices for the viewing of data for either a transporter group, a single transporter or all transporters in the scenario. For an example of the data provided in this area, enter "**G**" for group. Next, enter the group name; for an example enter "**B747**". Next, the diagnostics module will display the transporter usage data for all B747s in the scenario. The data given at C+140 in the S3DEMO scenario is shown in Figure 4-15.

The data provided in Figure 4-15 is divided into two parts. There is aggregate data about group transporter usage and there is information concerning the average load size for these transporters. The aggregate group usage data is at the top of the screen. Both the total number of loads and the total number of transporters in the given group is given. These numbers are good for drawing comparisons between different simulation runs of a single scenario using different transporter packages. The average number of cargo runs is listed next. For B747's in this scenario, 485 loads were carried by 23 aircraft. This translates to 21 trips with cargo per aircraft. Currently group transporter information does not provide data on the number of empty trips a transporter group undertakes. That information is, however, available on a single transporter basis. Below the general information mentioned above are three columns of cargo load data. The first column is the total quantity of each commodity category carried by the B747s in the S3DEMO scenario. The second column is the average load carried, while the third is the maximum capacity of each category for the B747. The figures from the S3DEMO scenario run indicate that the B747 was primarily used for passenger transport and at about eighty percent capacity.



```
   S3 Version 2.0

========         B747 Transporter Group Data at time 3360     (140  days)========
      Total Number Transporters In Group                           23
      Total Number of Loads                                        485
      Total Category                 Average Per Load      MaxCapable
      ================================================================

            Cargo  837120               1726
              POL  0                    0                                40000
              Pax  154109               317                                  0
             Ammo  17459                35                                 395
                                                                         40000

(R)eturn to Transporter Menu
```

Figure 4-14 Transporter Usage Menu



```
   S3 Version 2.0

========          Transports Diagnostic Menu at time 3360     (140  days)========
       1. Transporter Type (G)rouped
       2. (S)ingle Transporter
       3. (A)ll Transporters
       4. (R)eturn To Diagnostic Menu

     ENTER SELECTION AND STRIKE <ENTER>.
```

Figure 4-15 B747 Group Usage Display

The information in the transporter usage screen will show whether a transporter type is being used to its full potential. This data can be used to help size the transporter fleet for a given logistics scenario. A few runs with varying numbers of a single transporter type provides insight into what the appropriate number is for the model.

### *d. Closure Information*

The third area of data in the Diagnostics Module is closure data. Currently closure data is not graphically displayed for the user. Instead it *is* collected and saved to a file. To currently view and display closure information, this file must be brought into a spreadsheet and input into a chart. A graphics feature is planned for Version 3.0. Thus, selecting "**C**" for closure data from the Diagnostics Menu currently does nothing. Regardless, the data is collected and exists in data files at the end of each run of S3.

Two types of closure data are sought by the Diagnostics Module, unit and theater closure. Unit closure data, as mentioned in Chapter III, is kept by saving the simulation time of each significant deployment event for each unit in a scenario. The times are recorded when the first equipment arrives, and at the twenty-five, fifty, seventy-five, ninety, and one hundred percent unit inventory levels. For the S3DEMO scenario, Figure 4-16 shows the times recorded for the deployment of all units which did not begin the simulation active.

Theater closure data is kept as well as unit data. Theater closure is monitored for up to two different theaters in S3. A separate file is kept for each theater and a third for commodities moved between theaters, such as the 25Th Light Infantry swinging from the Middle East theater to the Southeast Asia theater at time C+90 in the S3DEMO wargame timeline. Theater closure is monitored for four categories of commodities; passengers, POL, unit equipment, and ammunition. The closure sums are

| Unit | First Equip | 25% | 50% | 75% | 90% | 100% |
|---|---|---|---|---|---|---|
| 3ACR | +24 | 34 | 45 | 48 | 50 | 51 |
| 82ABNDIV | +22 | 23 | 29 | 40 | 43 | 45 |
| 10MtnInf | +26 | 30 | 32 | 38 | 41 | 43 |
| 25THLtInf | +29 | 32 | 36 | 41 | 44 | 46 |
| 7MEB | +94 | 96 | 98 | 101 | 103 | 104 |
| 101ABNDIV | +110 | 111 | 122 | 128 | 128 | 128 |
| 25th Redeploy | +124 | 124 | 135 | 139 | | |

**Figure 4-16 Unit Closure Data Through C+140**

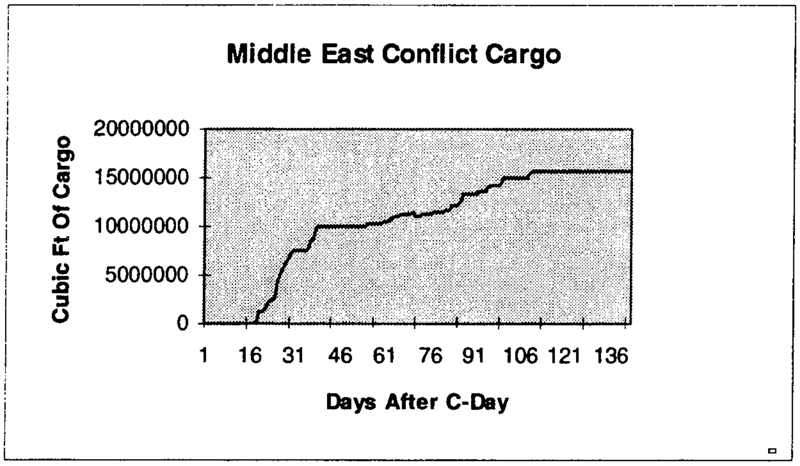


**Figure 4-17 Cargo Closure Data**

recorded every twenty-four hours for each theater. The data collected for the Middle East Theater of the S3DEMO scenario is shown below in Figures 4-17 through 4-20respectfully. The closure data given is identical to the data provided in several campaign logistics analyses of the Persian Gulf War. The graphs are self explanatory, but not necessarily straightforward. Figure 4-17 is a graph of the unit cargo closure over time. Cargo here is defined to be unit equipment and major equipment items such as tanks and trucks. Furthermore, host nation support was stated earlier to include POL in the Middle East theater. That statement was true, but this support was modeled as a fixed amount for the S3DEMO and subsequent fuel usage was supplied out of theater. The

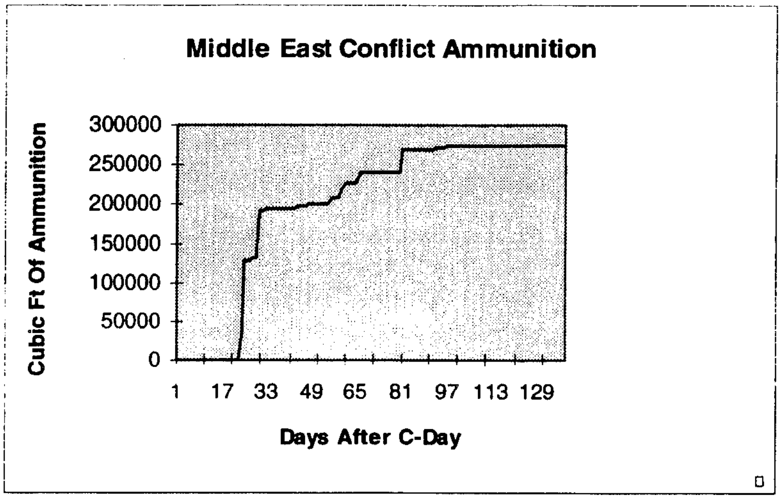


Figure 4-18 Ammunition Closure Data

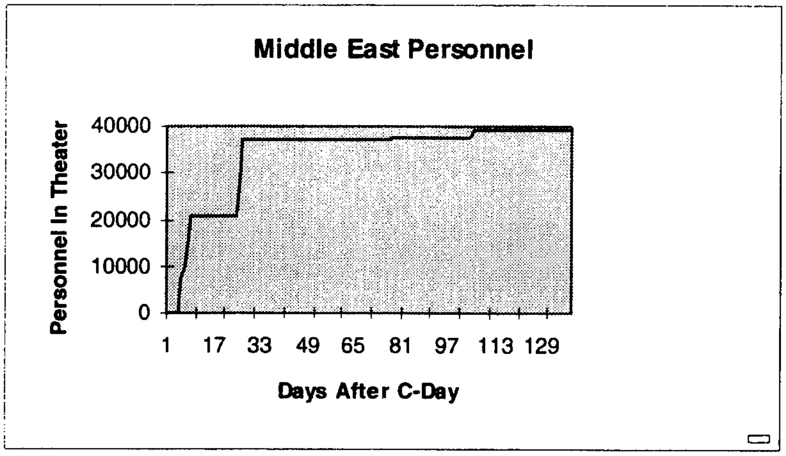


Figure 4-19 Personnel Closure Data

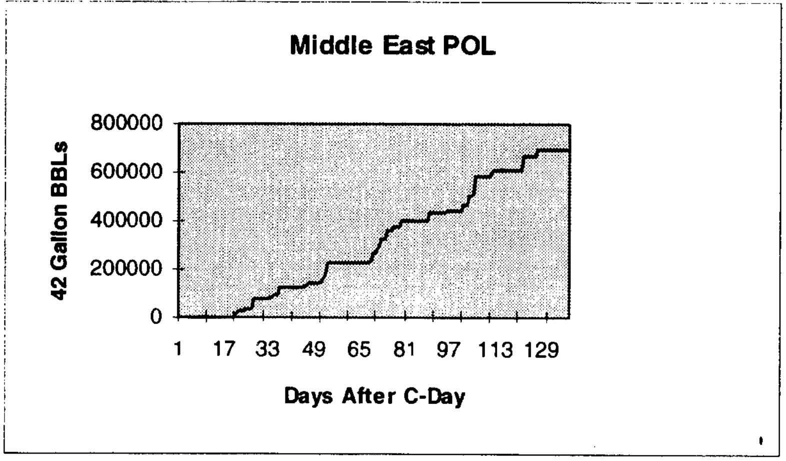


Figure 4-20 POL Closure Data

data shown in these graphs is primarily of interest to seminar wargamers who may be comparing the closure rates of several different logistics scenarios, but it is also appropriate for the computer wargamer who may be looking for trends in surge and sustainment into theater.

One thing that the closure graphs do not do is separate closure into surge and sustainment categories. Currently the data shown is an aggregate of the two. In addition, the data does not tell the user if the sustainment of commodities occured in a timely fashion. Although it is clear from the graphs that sustainment shipping is arriving in theater, the units may still have been undersupplied due to a variety of reasons. A better visual representation of sustainment would be a graph of daily commodity inventory over time. This graph would act as a track record for the inventory information that is currently available for a given instant in the Unit and Base display.

### 5. Summary

In summary, S3(Version 2.0) is a very capable logistics wargaming tool. The means by which S3 is employed and ways to seek direction in that employment have all been covered in this chapter. The six user control points and the S3 information displays are the pieces of S3 that must be understood in order to effectively use the model. Though there remain some shortcomings in S3, overall, the program is very capable and the user armed with the contents of this chapter should be able to build a logistics network in S3 and wargame the surge and sustainment for any specific scenario. This chapter concludes the introduction and description of S3 Version 2.0. The final chapter follows with conclusions about this version and recommendations for the next version of the S3 program.

# V. CONCLUSIONS AND RECOMMENDATIONS

## A. CONCLUSIONS

In conclusion, S3 is thought to be a valid logistics tool for both seminar and computer wargaming. This thesis has brought to fruition S3 (Version 2.0) as a logical step forward in the evolution of this tool. The new version improves upon the model fidelity and the user's access to network performance data. Although there are still gains to be made in both of these areas, Version 2.0 is a significant improvement over its predecessor.

S3, like any model, is an attempt to close in on the truth, but not the means to actually find it. Instead, S3 is a tool to help reduce the scope of possibilities, the idea being the truth is harder to fish out of an ocean than out of a lake. S3 provides a logistics measure for wargaming that is much closer to reality than the tools currently being used.

The true test for S3 will come with actual use of the model for wargaming. Looking to the past, however, it is evident that S3 has a niche to fill. In January of 1994 a seminar style logistics wargame, Logistics 2001, was held at the Naval War College in Newport. The wargamers there often talked in terms of what they could not do in terms of strategic mobility. The only answer given to some of these questions were simple true or false about whether the task can or can not be accomplished. With S3 the capability exists for simple on-site sensitivity analysis comparing different makeup's of strategic assets with the specific wargame scenario.

In the computer wargaming arena, S3 provides logistic data for ENWGS. Although the optimum ENWGS system would have the logistic constraints provided by S3 built in, the current version of S3 is sufficient to fill the void in the meantime. Logistics in wargaming is here to stay and S3(Version 2.0) is headed in the right direction.

## B. DIRECTION FOR S3 VERSION 3.0

Version 2.0 is not, nor should it be, the last version of S3. There is room for growth in model fidelity, the accessibility of sustainment information and in the user interface. Specific recommendations for improvements in these areas follows in the paragraphs below.

The first topic is model fidelity. Increasing model fidelity is a matter of continually updating the computer code to produce a better representation of reality. There are several ways this can be done. Two key areas stand out that were encountered during the work for this thesis, transporter management and port modeling. The current transporter manager needs to have additional logic incorporated that looks forward in time for transporter assignment. The current version fills transportation requests from a list of transporters that are *immediately* available. In other words, if a viable transporter exists 500 miles from the requester at time C+50, the transporter that becomes available 200 miles away at time C+51 is ignored and left out of consideration. Some means of considering these transporters should be devised and implemented in S3.

Port modeling is also in need of an additional dose of reality. For ships, draft needs to become an additional constraint. Port unloading times also need to become random. The current software uses mean unloading times without any chance of variance. In addition, theater ground transportation needs to be modeled as both a transporter *and* a commodity to be transported into theater. It is perhaps the biggest flaw in the current modeling of the logistics network that this transportation is assumed in theater at the beginning of each simulation run.

After increasing model fidelity, the addition of a sustainment information display would most improve the effectiveness of S3. The current program provides the user snap shots of unit supply status at one moment in time, but it does not provide a historical picture of unit sustainment. A simple addition to the Diagnostics Module would fill this need. A new method should be created which measures and saves unit and theater commodity inventories on a daily basis and saves this data to a file. The user should then be able to view this collected data in a graphical form. From this data a judgement as to the effectiveness of the logistics network in providing theater sustainment could be made.

In addition to model fidelity and sustainment information, the user interface for S3 is also in need of significant change. The current menu system is archaic and drab. Neither of these contributes to the model effectiveness. A modern windows interface is required. The user should be able to use mouse click and drag actions to direct the model. Both the SetUp and the Execution phases of S3 should use these new windows interfaces. Examples of where this capability would improve S3 are numerous. Displaying database files and allowing access to all fields in the file from one window and selecting bases to view their logistics status rather than typing a case sensitive name are just two to name a few. Regardless of how the windows are implemented, they would make S3 easier to use. Probably the biggest drawback to the current version of S3 is the difficulty in using the menu interface.

The effort required to add this new code to the S3 model is not trivial. In fact, as the model becomes more complex the work for each new addition becomes harder and harder. However, the work done for this thesis has left the author with the distinct impression that the payoff from the completion of these tasks will be worth it. The S3 model has come a long way so far and with the work of an additional thesis or two it will be something that the Naval War College can use with confidence.

# LIST OF REFERENCES

1. Clausewitz, Carl, *On War*, Michael Howard and Peter Paret, eds., p. 20, Princeton University Press, 1976.

2. Long, J.A., *Modeling Theater Level Logistics For Wargaming,* Master's Thesis, Naval Postgraduate School, Monterey, California, December 1993.

3. CACI Products Company, MODSIM II The Language of Object-Orientated Programming: Reference Manual, 1993.

## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
Cameron Station
Alexandria, Virginia 22304-6145

2. Library, Code 052 2
Naval Postgraduate School
Monterey, California 93943-5002

3. Deputy Chief of Naval Operations, N401C 2
ATTN: CDR Kevin Cheezum, USN
Department of the Navy
Washington, D. C. 20350-2000

4. Defense Logistics Studies Information Exchange 2
U.S. Army Logistics Management Center
Fort Lee, Virginia 23801

5. President, 2
Naval War College
ATTN: WGD 334L
686 Cushing Road
Newport, Rhode Island 02841-1207

6. PROF David A. Schrady, Code OR/So 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93943-5000

7. PROF Thomas E. Halwachs, Code OR/Ha 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93943-5000

8. LT Thomas G. Halvorson 2
c/o Mr. Robert G. Halvorson
5283 Red Fox Lane SE
Alexandria, Minnesota 56308

9. LT John A. Long 1
c/o Mrs. Hester J. Long
242 W. Lemon Street
Lancaster, Pennsylvania 17603

10. CDR Newton Rodriques Lima 1
Centro de Analises de Sistemas Navais
Avenida Pasteur, 480
Urea - Rio de Janeiro 22290-000
Brazil