Computer Corporation of America

575 Technology Square
Cambridge
Massachusetts 02139
617-491-3670

6

Spatial Data Management System.

9

Final Report. 1 Nov 77-
11/1/77-12/30/80
30 Dec 80,

10

Christopher F./Herot
Richard/Carling
Mark/Friedell
David/Kramlich

11

30 Dec~~ember~~ 1980

12 30

Submitted to: Defense Advanced Research Projects Agency
Defense Sciences Office
Systems Sciences Division
1400 Wilson Boulevard
Arlington, Virginia 22209

Attention: Dr. Clinton Kelly

Submitted by: Computer Corporation of America
575 Technology Square
Cambridge, Massachusetts 02139

15

This research was supported by the Defense Advanced Research Projects Agency (DARPA) of the Department of Defense under Contract No. MDA903-78-C-0122, ARPA Order-3487 3487. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of DARPA or the United States government.

387285

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
Computer Corporation of America
675 Massachusetts Avenue
Cambridge, MA 02139

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP
None

3. REPORT TITLE

"Spatial Data Management System"

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Final report

5. AUTHOR(S) *(Last name, first name, initial)*
Christopher F. Herot, Richard Carling, Mark Friedell, David Kramlich

6. REPORT DATE
30 December 1980

7a. TOTAL NO. OF PAGES
25

7b. NO. OF REFS
10

8a. CONTRACT OR GRANT NO.
MDA-903-78-C-0122

b. PROJECT AND TASK NO.

c. DOD ELEMENT

d. DOD SUBELEMENT

9a. ORIGINATOR'S REPORT NUMBER(S)

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
None

10. AVAILABILITY/LIMITATION NOTICES
Distribution of this document is unlimited.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

11. SUPPLEMENTARY NOTES
None

12. SPONSORING MILITARY ACTIVITY
Defense Advanced Research Projects Agency/Defense Sciences Office
System Sciences Division

13. ABSTRACT

This is the final report of a project by Computer Corporation of America (CCA) to design and implement a prototype Spatial Data Management System (SDMS). SDMS is a unique, man-machine interface that provides a uniform means of accessing different kinds of computerized information.

Other documents, listed in the bibliography, have described in detail the progress of the SDMS project. This report adds information about the following:

1) The transfer of SDMS from its development environment at CCA to a demonstration environment at the Demonstration and Development Facility of DARPA's Cybernetics Technology Office.
2) Our initial efforts towards designing and implementing a system to provide improved methods of training tactical decision makers.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computer graphics						
Database management						
Spatial data management						
Tactical decision making						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through

______________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through

______________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through

______________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, *or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

Contents

1. INTRODUCTION

This is the final report of a project by Computer Corporation of America (CCA) to design and implement a prototype Spatial Data Management System (SDMS) -- a unique, man-machine interface that provides a uniform means of accessing different kinds of computerized information. SDMS is a computer-based tool that uses graphical symbols to convey information. SDMS merges the visual power of symbols with the information-handling facilities of a conventional database management system (DBMS). It provides a highly effective means for people who are not data processing professionals to organize and access a database.

Spatial data management is a technique for organizing and retrieving information by representing and positioning it graphically. Data are viewed through color displays. The displays show flat "data surfaces" on which pictorial representations of the data are arranged. The SDMS "Graphical Data Surface" (GDS) is the collection of all the data surfaces, or all the pictures that the user can access. SDMS automatically creates these pictures from data stored by the DBMS.

The user can traverse the data surfaces or "zoom" into an image to obtain greater detail. This approach permits many types of questions to be answered without requiring the use of a keyboard. A conventional query language is also provided.

Spatial data management is motivated by the needs of a growing community of people who need to access information through a DBMS but are not trained in the use of such systems. A database viewed through SDMS is more accessible and its structure is more apparent than when viewed through a conventional DBMS. Users of conventional DBMSs can access data only by asking questions in a formal query language. In contrast, users of SDMS benefit from the ability to access computer-resident information while retaining a familiar, visual orientation.

By presenting information in a natural, spatial framework, SDMS encourages browsing and requires less knowledge of the contents and organization of the database. Thus, a user can find the information he needs without having to specify it precisely or know exactly where in the database it is stored. Users can easily organize, locate, and handle a great deal of information of different types.

Other documents have described the progress of the SDMS project (see the bibliography, in Section 4). The rest of this document adds the following information:

- Section 2 describes the transfer of SDMS from its development environment at CCA to a demonstration environment at the Demonstration and Development Facility of DARPA's Cybernetics Technology Office (CTO).

- Section 3 describes our initial efforts towards designing and implementing a system to provide improved methods of training tactical decision makers and keeping them well trained.

2. SDMS/CTO PRODUCT INTEGRATION

In this phase of our effort, we transferred SDMS from its development environment to a demonstration environment. In addition, we optimized the system for effective demonstrations of SDMS, as well as other CTO products. Specifically, we:

- Transferred the physical SDMS hardware and software from Cambridge to Washington.
- Enhanced the hardware and software to provide more effective demonstration of SDMS and other products.
- Located and corrected errors in the SDMS software.
- Trained and supported the CTO/DDF staff on relevant aspects of SDMS.
- Integrated other CTO products with SDMS.
- Demonstrated SDMS to potential users, as directed by ARPA/CTO.

Details of these tasks are provided in the following subsections.

This effort has demonstrated clearly that UNIX-based products can be integrated easily to run under SDMS. The resulting demonstration makes clear the value of such an integrated system for rapid decision making, where access to large amounts of information is necessary.

2.1 Physical Transfer of SDMS

Physical transfer of the equipment was accomplished by a commercial moving company. De-installation and reinstallation were accomplished primarily by the computer vendor, with some assistance by CCA personnel. No major problems were encountered.

2.2 Hardware and Software Enhancement

The original SDMS hardware was enhanced by the procurement of an Audio/Video electronic switcher, which allowed programs running on the SDMS computer to dynamically select the sources and destinations of audio and video signals during a demonstration. This hardware was supported by software developed by CCA, which allowed use of the switcher at three levels:

- Direct access to low level switches was provided by routines implementing switcher primitives.
- A higher level of functionality was provided by routines that provided symbolic access, through configurable tables to switch lines, and groups of lines, logically.
- The highest level of support allowed access to the second level functionality through typed commands. This level also allowed simple modification of the configuration tables through standard text editors.

This support enhanced the demonstration capability of the CTO/DDF, as well as simplified the integration of CTO products. Additional software enhancement, resulting from the writing of an Ann Arbor 4080 simulator, allowing the PRESS program to run under SDMS with minimal difficulty.

2.3 Location and Correction of Errors

A standard error-logging procedure was established and made accessible through SDMS. All known errors were cataloged, and a number of major problems corrected.

2.4 CTO/DDF Staff Training and Support

CTO/DDF staff and Washington-based CCA staff were trained in a three-day seminar held after SDMS was installed in Washington. Numerous informal sessions were held subsequently and CTO/DDF staff were supported in response to specific needs.

2.5 Product Integration

The products listed in the original proposal were integrated with SDMS. This integration consisted of creating an SDMS data surface, providing relevant images at various points on the data surface, and arranging for the activation of the specific products associated with the different images, when the image was selected by zooming in on it. Additionally, products were deactivated automatically, as directed by SDMS mechanisms. In the rest of this section, we describe the results of integrating the products listed in the original proposal.

1. EWAMS was successfully integrated with SDMS. After invocation via an SDMS port, the EWAMS imagery is displayed on the projection TV system as well as on the SDMS monitors. Input for EWAMS is accepted from the SDMS keyboard.

   Integration was accomplished by using the video/audio switcher to make the Tektronix 4027 image generated by EWAMS appear on the appropriate SDMS screens. Standard UNIX facilities were used to switch the input stream to the SDMS keyboard, so no modification to EWAMS code was required.

2. PRESS was successfully integrated with SDMS. This integration was accomplished by providing an Ann Arbor 4080 simulator running on the SDMS graphics hardware and by simulating the 4080 keyboard through the SDMS keyboard.

   It should be noted that the data-gathering portion of PRESS does not operate reliably, due to unresolved problems in PRESS itself. Therefore, the status of PRESS should be checked prior to any demonstration, and the data-gathering portion restarted if necessary.

3. Executive aids for Crisis Management was successfully integrated with SDMS. Integration was accomplished by providing four selection ports on SDMS, corresponding to the four components of EXECAIDS. Invocation of any of these ports caused execution of the appropriate program and switching of the resulting Tektronix 4027

output to the SDMS displays.

4. The Ultra Rapid Reader was successfully integrated with SDMS. As integrated, it drives a Tektronix 4027 whose output is switched to the SDMS monitor.

5. At the request of ARPA/CTO, the Group Decision Aid was not integrated. This decision was made since the product requires several terminals for its use, and no effective way could be seen to demonstrate it in the current SDMS environment.

6. Soviet Executive Decision Aids was successfully integrated with SDMS. It operates in the same manner as the Executive Aids for Crisis Management described earlier.

7. Terrorism Research and Analysis Program (TRAP) was not integrated, at the request of ARPA/CTO.

8. Videodisk Based Interactive Maps was not integrated, since the requisite hardware was not provided. A subset of this capability could be provided through the CTO Managerial and Technical Videodisk described below.

9. Marine Corps Combat Readiness Evaluations System (MCCRESS) was not integrated, at the request of ARPA/CTO. The decision resulted from the non-interactive operation of MCCRESS, which made it of limited interest in the SDMS environment.

10. The integration of CTO Managerial and Technical Videodisk was performed by making a general interface from SDMS to videodisk. The demonstration invocation provided in SDMS allowed single frame stepping and continuous mode sound and picture presentation of selected portions of the vendor-supplied demonstration videodisk. The videodisk image was presented on the projection TV screen.

11. Scoring Rule was successfully integrated with SDMS. This product drove the Tektronix 4027, whose image was switched to the SDMS displays. During integration of this product, a number of program and database errors were encountered in the original product. It is suggested that any demonstration be carefully rehearsed to avoid these problems

12. OPINT, RAM, DECISION, and HIER (INFER) were successfully integrated with SDMS. The output they generated to the Tektronix 4027 screen was switched to the SDMS

displays at invocation. A number of problems were encountered in testing the programs as delivered, and it is suggested that any demonstration involving them be carefully rehearsed.

13. Duncan/Job/Probability Estimators was not integrated, since the code was not provided to CCA.

14. Steamer was not integrated, since the working code was not provided to CCA.

3. COMPUTER-AIDED TRAINING OF TACTICAL DECISION MAKERS

Since the discovery that guns are more effective than swords, there has been a continuing evolution of improved technology in armed conflict. In many cases, this evolution has made it physically easier to use modern military systems but more complicated to make decisions. This increased complexity of decision-making arises from several things:

- The increased delivery speed of modern weapons reduces the available decision time.
- The increased range of modern weapons and sensors increases the number of threats that must be considered simultaneously.
- The increased destructive power of modern weapons extracts severe penalties for a single mistake, so correctness of response is vital.

These characteristics of modern weapons most severely affect tactical decision makers. Such people frequently must make correct decisions -- involving consideration of large numbers of threats and response options -- within a few seconds.

In addition to the direct effects of technology, there is a more subtle effect. In tactical decision making, the technology stands between the decision maker and the world he affects. Factors like good eyesight and personal strength, though vital to the swordfight, are of little use to the officer dealing with sonar screens and radar signatures. The contemporary officer must have a good understanding of the capabilities and limitations of the technology he commands.

In this report, we present initial efforts towards designing and implementing improved methods of training tactical decision makers and keeping them well trained.

3.1 Problem

In this section, we focus on the problem of training a Tactical Action Officer (TAO). In the United Stated Navy, a TAO makes the tactical decisions for a small group of ships, submarines and aircraft assigned to a specific mission. Generally, the orders given by the TAO result in the use of sensors or weapons without being further reviewed by senior officers. There are four aspects of training a TAO:

- Development of abstract knowledge of weapons, sensors, and doctrine -- both the TAO's and those of potential opponents.
- Practice, where the abstract knowledge is applied to real or simulated problems.
- Review and revision, where new information is presented and old information is reviewed.
- Practice of new material, so that it is incorporated into the already acquired skills.

The first and third items primarily involve presenting information to the TAO. They seem to be handled adequately by a traditional classroom/textbook environment. The second and fourth items, however, involved practice in applying new knowledge and developing new skills.

When conducted in a real, operational environment, practice has two primary problems:

- It is very expensive, involving thousands of men and millions of dollars worth of equipment.
- It does not serve the TAO's pedagogical needs, since exercises generally are driven by more pragmatic concerns (e.g., testing new weapons systems).

Because of the high cost, TAOs get very little real-world practice. Because of the non-pedagogical approach, the practice the TAO does receive has limited value.

3.2 Solution

In this section, we summarize an approach for providing relevant, inexpensive practice for the TAO training environment. In the first part of our effort, we investigated an alternative to actual practice: computer-based simulation. The modern tactical environment is in many ways ideally suited to this approach. As noted earlier, much of the TAO's information is obtained indirectly via displays and text. Because of this, it is not difficult to simulate the important aspects of the sensory environment.

In our effort, we have focused on a simulation of one portion of the TAO's environment: antisubmarine warfare. The simulation runs on a $2000 personal computer. By connecting two of these computers (which can be connected via ordinary telephone lines over long distances), TAOs can practice at very low cost.

In addition to being less expensive, the personal computer and associated hardware are extremely portable. A complete system, requiring only a standard wall plug and telephone jack, can be accommodated in an oversized briefcase.

The second part of our effort was to develop ways of generating problems for practice. Each of these problems would exercise specific pieces of knowledge that a student had just learned, and would require the use of only the knowledge the student had previously learned.

The results of this effort were the selection of a minimal but comprehensive set of problems, the exploration of several aspects of an integrated, knowledge-based system, and the creation of descriptions of specific algorithms for scenario generation.

3.3 Example Session

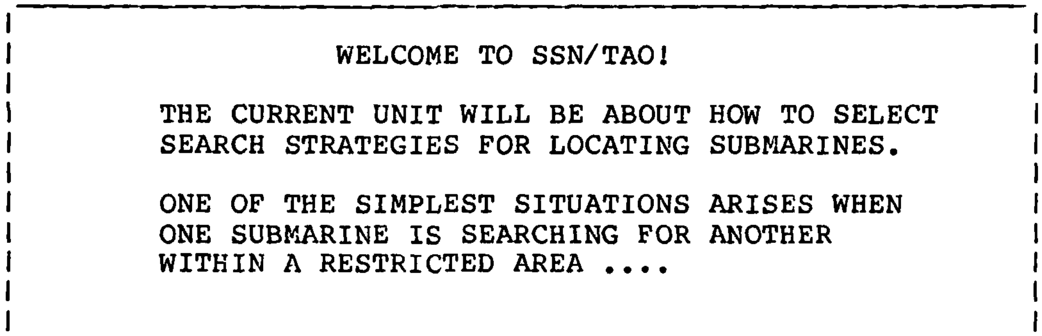
In this section, we present an example of the integration of simulation and scenario generation into an effective training system. SSN (Simulated Antisubmarine) is the simulation component of the system, and is based upon a commercially-available game. TAO (Trainable/

Tactical Automated Opponent) is the actual training portion of the system. It includes an automated opponent, scenario generation, and a student model.

The next few pages show a hypothetical session with a fully implemented SSN/TAO system. The student would see what is shown in the figures. In this session, the student is learning about search strategies for locating submarines. After text has been presented, he is encouraged to use what he has learned in practice problems.

The equipment required for this session includes a personal computer -- used for text presentation and for the simulation -- and a moderately large, central computer on which the problem generation software is run.

In Figure 3.1, the student is initiating an SSN/TAO session. Since the system remembers where the student is in the lesson plan, it starts without further interaction. The student presses the carriage return key on his terminal.



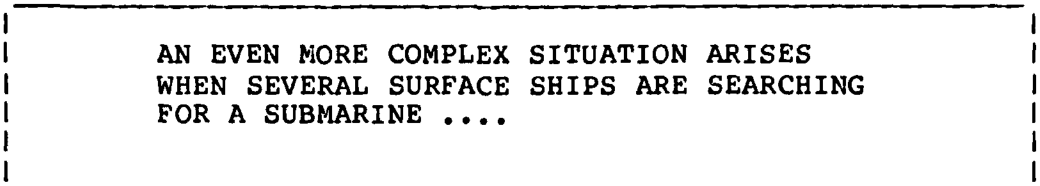
```
                 WELCOME TO SSN/TAO!

     THE CURRENT UNIT WILL BE ABOUT HOW TO SELECT
     SEARCH STRATEGIES FOR LOCATING SUBMARINES.

     ONE OF THE SIMPLEST SITUATIONS ARISES WHEN
     ONE SUBMARINE IS SEARCHING FOR ANOTHER
     WITHIN A RESTRICTED AREA ....
```

Figure 3.1 Screen 1

More text is presented (Figure 3.2). This material could be presented equally well in printed form. However, the low cost and ease of update of having the text stored on the personal computer would make use of the computer desirable in practice.

The student can read more pages by pressing return, or he can go back one page at a time by pressing "B." The student continues reading pages through many more screens until: the student presses return.

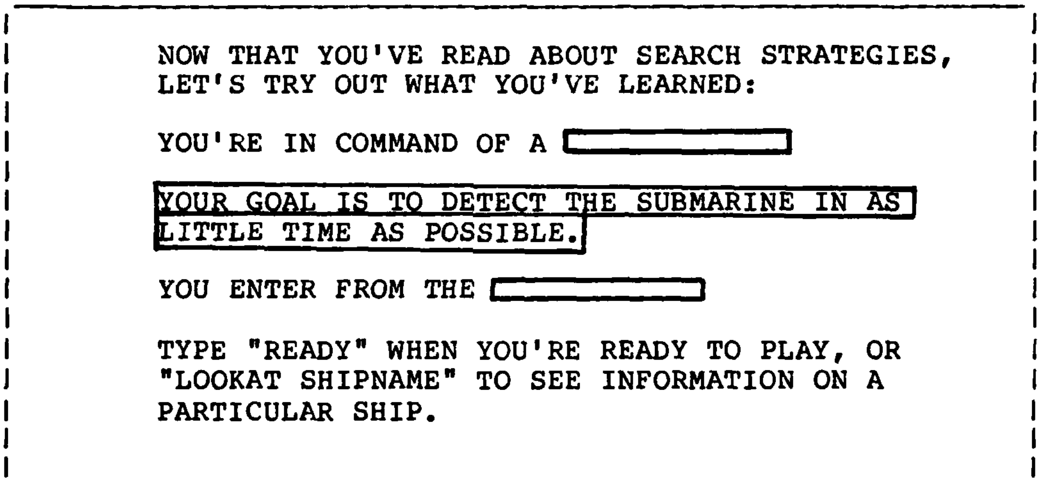
In Figure 3.3, the student is presented with a generated scenario that will exercise some facet of the information he has just been taught. The boxed areas show sections generated by the computer -- indicating the great



AN EVEN MORE COMPLEX SITUATION ARISES
WHEN SEVERAL SURFACE SHIPS ARE SEARCHING
FOR A SUBMARINE ....

Figure 3.2 Screen 2

flexibility inherent in the scenario-generation capability.



NOW THAT YOU'VE READ ABOUT SEARCH STRATEGIES,
LET'S TRY OUT WHAT YOU'VE LEARNED:

YOU'RE IN COMMAND OF A □

YOUR GOAL IS TO DETECT THE SUBMARINE IN AS
LITTLE TIME AS POSSIBLE.

YOU ENTER FROM THE □

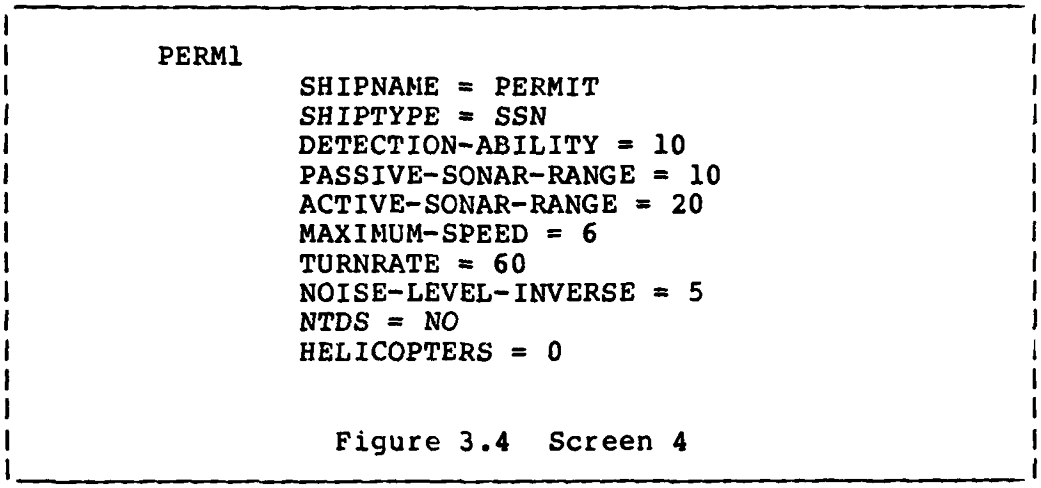
TYPE "READY" WHEN YOU'RE READY TO PLAY, OR
"LOOKAT SHIPNAME" TO SEE INFORMATION ON A
PARTICULAR SHIP.

Figure 3.3 Screen 3

The student, having forgotten the relevant attributes of his platform, the "Permit," asks for a display by typing "look at Perml." The requested information about the student's platform is displayed (Figure 3.4). This display is generated entirely within the personal computer. The student presses return.

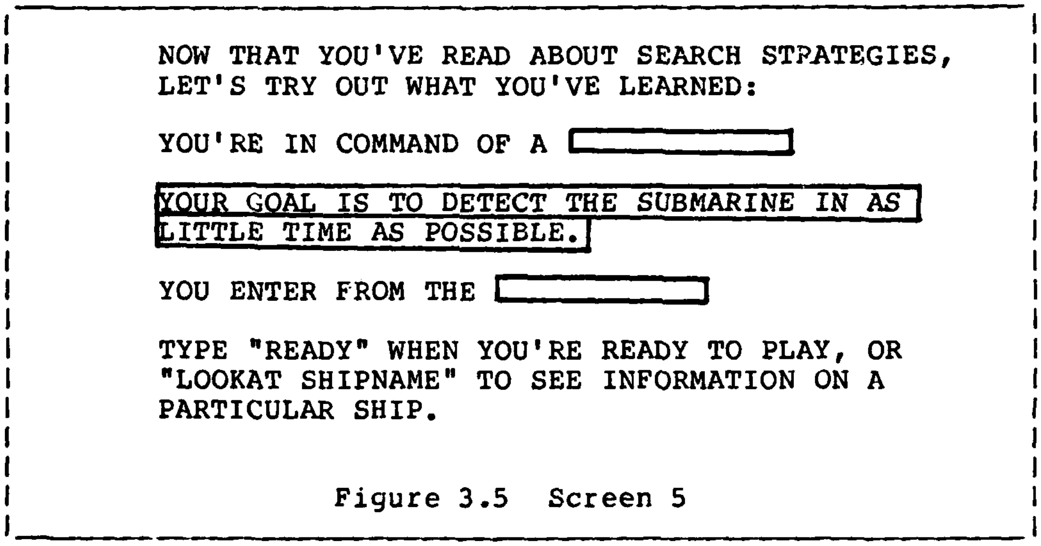
The student is returned to the scenario parameters (Figure 3.5). The student types "READY" and presses return.

The student's submarine is shown at the beginning of the practice (Figure 3.6). The opposing submarine does not appear, since the student does not know its location. The student allows this submarine to continue at its current speed and course by typing nothing.



```
PERM1
        SHIPNAME = PERMIT
        SHIPTYPE = SSN
        DETECTION-ABILITY = 10
        PASSIVE-SONAR-RANGE = 10
        ACTIVE-SONAR-RANGE = 20
        MAXIMUM-SPEED = 6
        TURNRATE = 60
        NOISE-LEVEL-INVERSE = 5
        NTDS = NO
        HELICOPTERS = 0
```

Figure 3.4 Screen 4



```
NOW THAT YOU'VE READ ABOUT SEARCH STPATEGIES,
LET'S TRY OUT WHAT YOU'VE LEARNED:

YOU'RE IN COMMAND OF A □

YOUR GOAL IS TO DETECT THE SUBMARINE IN AS
LITTLE TIME AS POSSIBLE.

YOU ENTER FROM THE □

TYPE "READY" WHEN YOU'RE READY TO PLAY, OR
"LOOKAT SHIPNAME" TO SEE INFORMATION ON A
PARTICULAR SHIP.
```

Figure 3.5 Screen 5

The student, now being potentially in range of the opposing submarine, turns on his action sonar, thus increasing his detection range. This also instantly causes him to be detected by the opponent. The opponent submarine either must must be near the "bottom" of the area or has been traveling at less than maximum speed.

In Figure 3.7, the student has typed a "C 22S" command to change course to heading 22S degrees. The heading change requires some time to take effect.

In Figure 3.8, the student catches the opponent trying to sneak across the southern edge of the area.

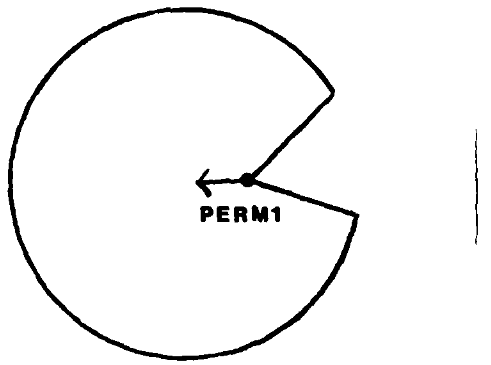


Figure 3.6 Screen 6

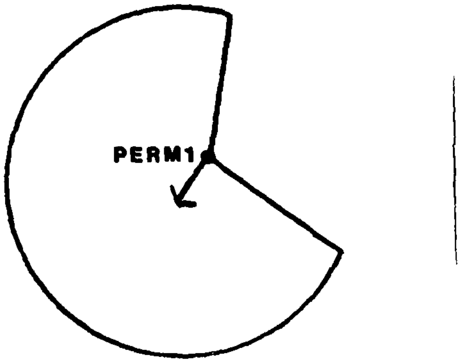


Figure 3.7 Screen 7

In Figure 3.9, the student has typed "C 120" to change course to follow the opponent.

The student "wins" (Figure 3.10), since he tracked the opponent for the required amount of time. Though the student was successful, the system noticed that he forgot to turn off his active sonar after a detection. Because this which was part of the doctrine presented, the system redisplays the appropriate part of the doctrine.

This completes the detailed example. If we were to follow the practice for more cycles, we would notice

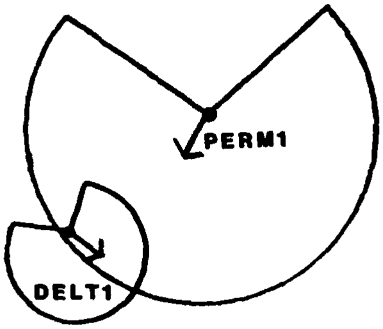


Figure 3.8 Screen 8

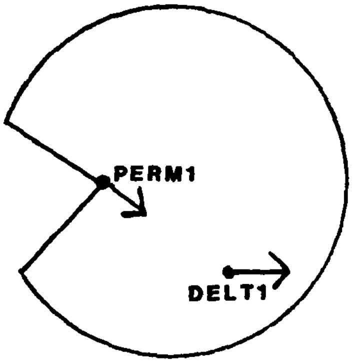


Figure 3.9 Screen 9

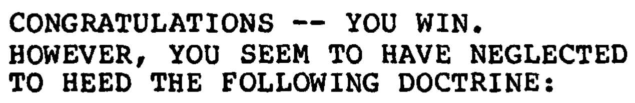

CONGRATULATIONS -- YOU WIN.
HOWEVER, YOU SEEM TO HAVE NEGLECTED
TO HEED THE FOLLOWING DOCTRINE:

Figure 3.10 Screen 10

several things:

- Each problem is unique - there are virtually no duplications.

- Even when several successive problems test the same knowledge fragment, they generally are quite different.

- Problems seem to increase in difficulty during a given session. This arises because the problem selection algorithm starts with the most general cases and then moves toward more specific situations.

3.4 System Model

In this section, we present a brief, two-level model of the integrated system. This model is presented in the form of a structured diagram with some explanatory tests. The diagrams consist of two basic elements: boxes, which are processes, and lines, which represent flow of information. The lines are further subdivided into:

1. Inputs - representing data to be transformed by the process. They always enter the left side of a box.

2. Controls - representing the information used to make decisions. They always enter at the top.

3. Outputs - representing the results of a process. They exit at the right of a box.

4. Mechanisms - the means of the implementation of a process. They enter from the bottom of a box.

Figure 3.11 is a high level model of the system, from the program's viewpoint.

Present Lesson causes a single, selected lesson to be presented to the student. The selection is controlled by a combination of the lesson plan and the results of the preceding practice. Information about the contents of the lesson are passed to subsequent processes. This process is implemented entirely on the personal computer, although some of its controls come from the TAO computer.

Generate Scenarios creates a specific scenario that will test a specific fragment of knowledge from the current lesson. It does this by applying information

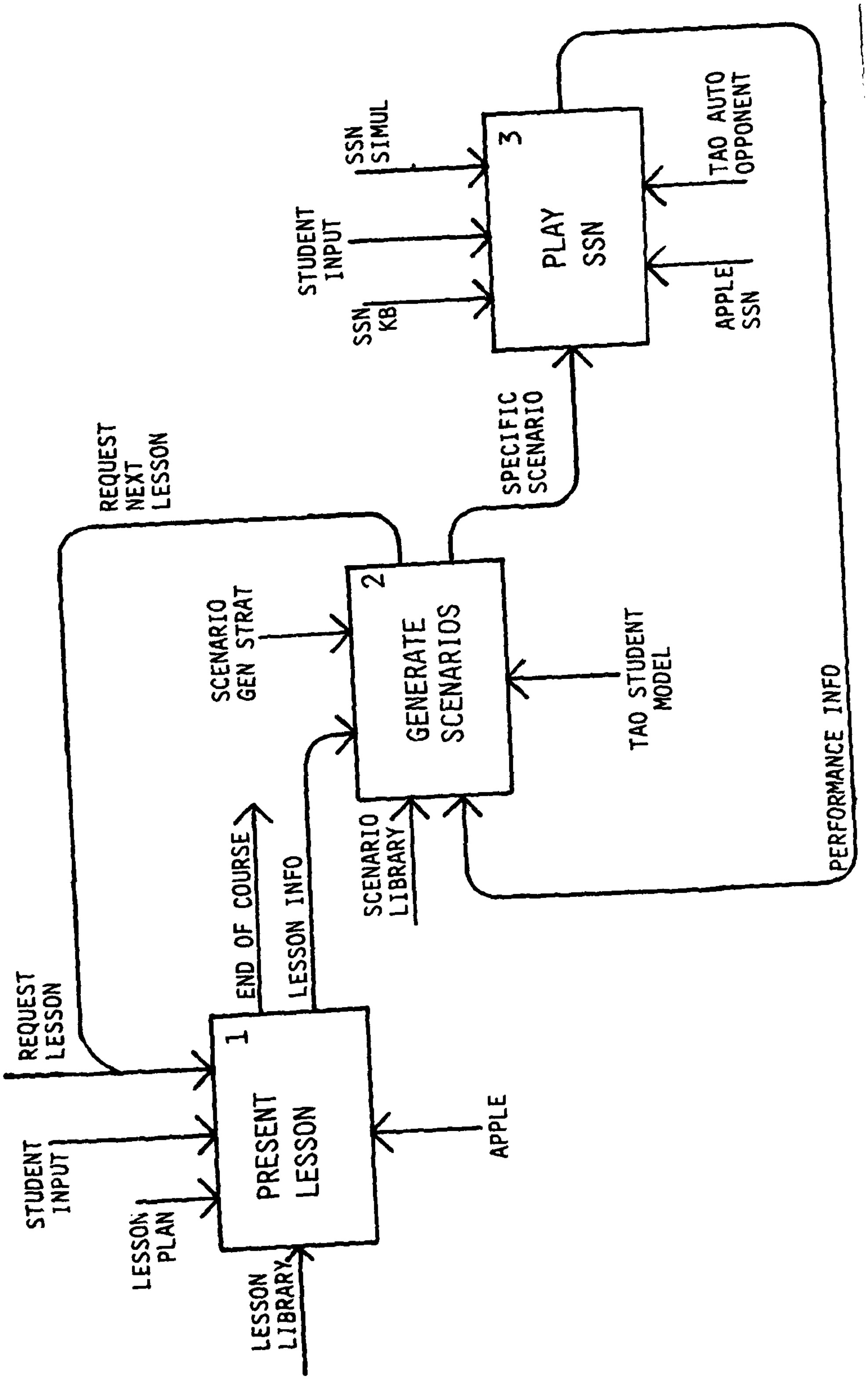


Figure 3.11 Program View -- Top Level

about the current lesson and the student's performance on the previous scenario, to select a skeleton scenario from the scenario library. The scenario generation strategy is then used to create a specific scenario. If the performance information applied to the scenario generator's internal model of the student indicates that the student has had sufficient practice with the current lesson, then the present lesson process requests that the next lesson begins. This process is implemented on the TAO computer, and makes heavy use of the student model.

Play SSN is the process where practice actually occurs. A specific scenario is used with an SSN simulation. The student and Automated Opponent then provide control information that directs the flow of play. This process is implemented in two places. The SSN simulation runs on the personal computer, while the automated opponent portion is part of the TAO component and executes on the TAO computer.

3.4.1 Present Lesson

Get Next in Plan selects the next lesson in the lesson plan, based on an external request for a view-lesson. If there are no more lessons, it indicates that the end of the course has been reached. A lesson-ID is normally output.

Present Text uses the lesson-ID to select the appropriate text from the lesson library, which is then presented to the student. When the student indicates completion of the lesson, the lesson-ID is passed to the next process.

Get Lesson Information for Scenario Generator uses the lesson-ID to select and output the TAO encoded form of the lesson.

3.4.2 Generate Scenarios

Select Skeleton Scenario selects a specific scenario from the scenario library, based upon the piece of knowledge chosen to be exercised. The choice of the piece of knowledge depends upon performance information, the

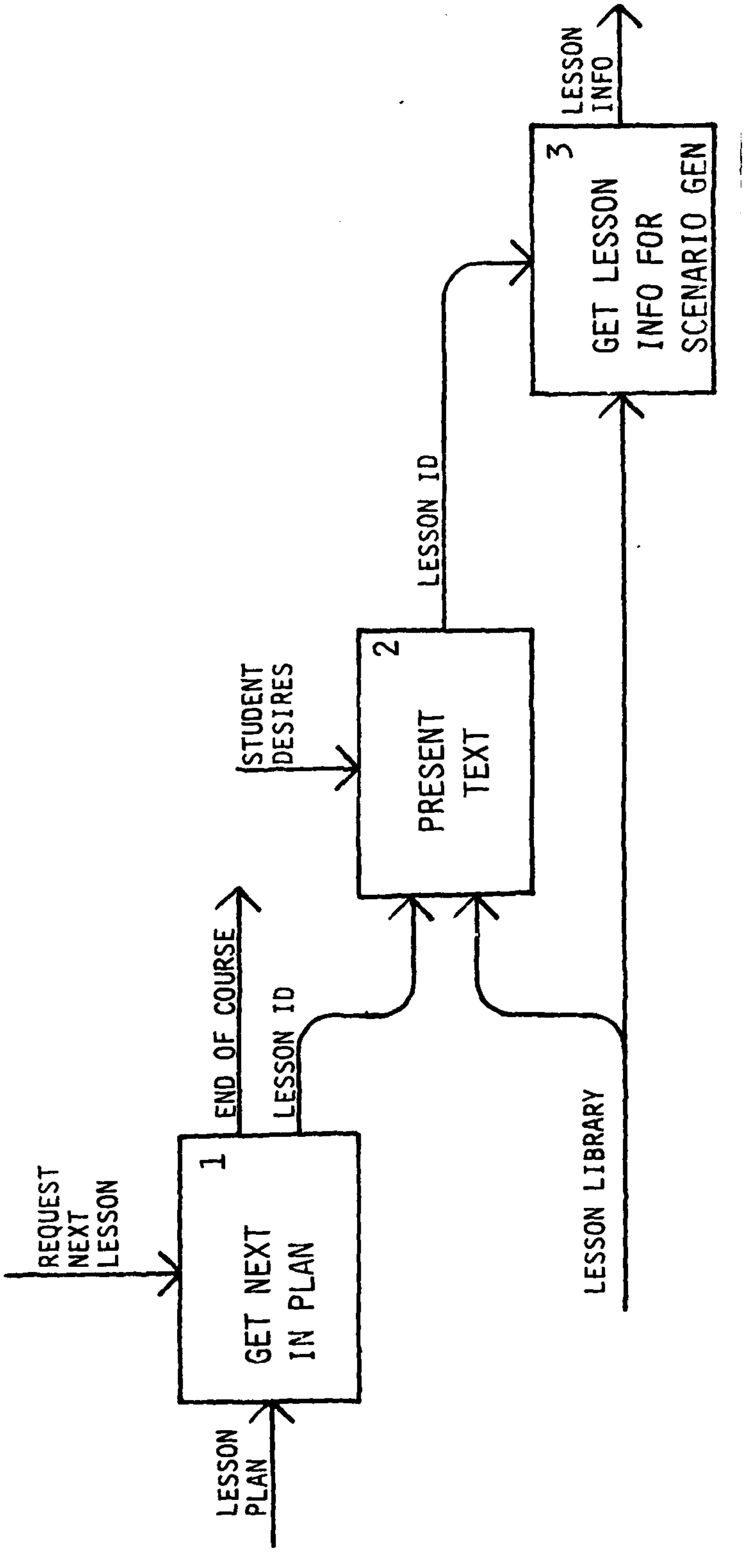


Figure 3.12 Program View -- Decomposition of Present Lesson

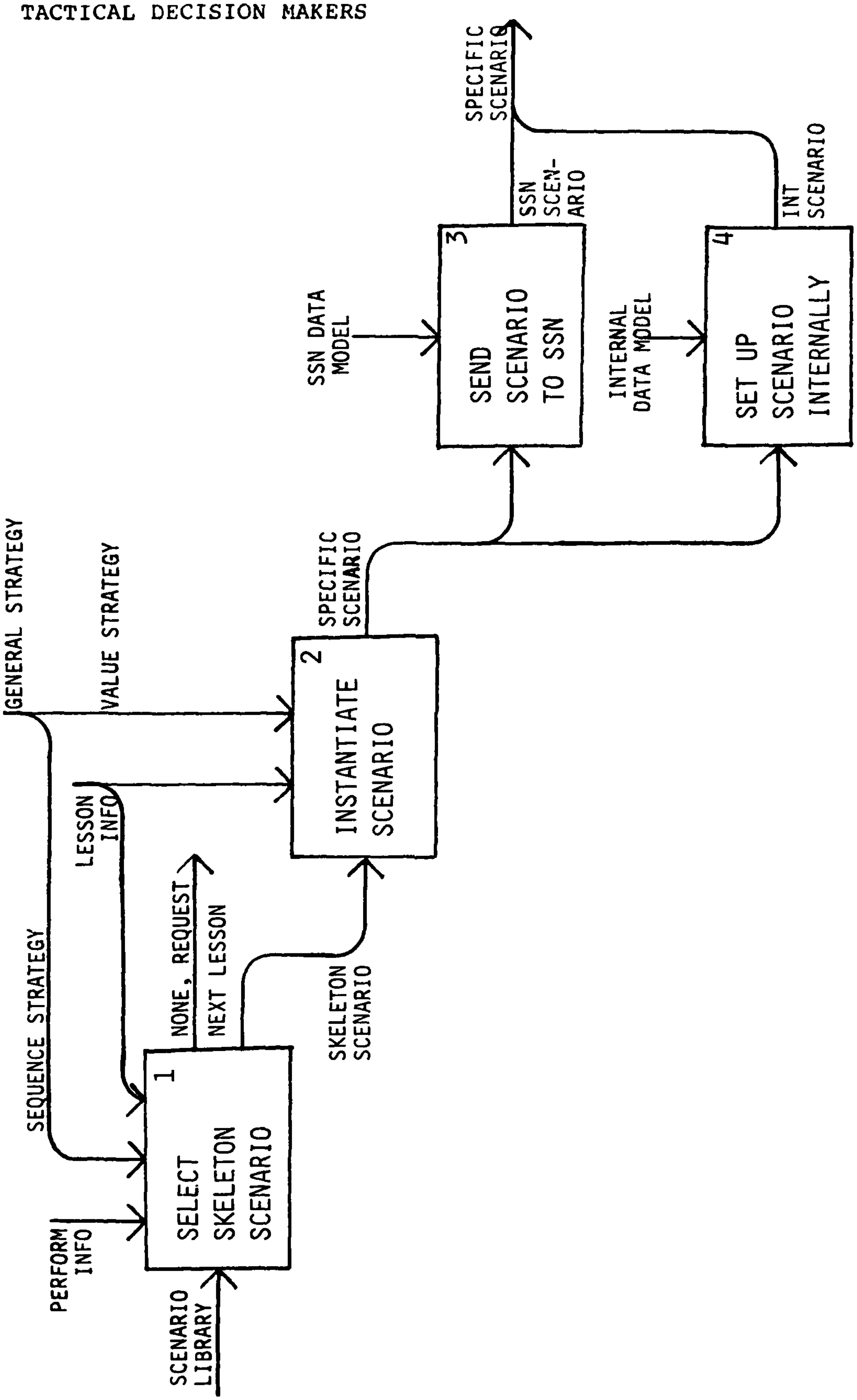


Figure 3.13 Program View -- Decomposition of Generate Scenarios

sequencing strategy, and information supplied as part of the lesson. If no piece of knowledge needs further exercise, a request is generated to proceed to the next lesson.

Instantiate Scenario takes a general, skeleton scenario and makes it into a specific scenario. It does this by selecting values for variables, as well as selecting among alternative expressions. Lesson information and a value strategy control the choices made.

Send Scenario to SSN packages the specific scenario information in a form acceptable to the SSN simulation and ships it out over the communications line.

Set up Scenario Internally puts the specific scenario in the internal form required by the TAO.

3.4.3 Play SSN

Set up Initial Automated Opponent Conditions establishes the beginning context and termination rules from the specific scenario generated for the Automated Opponent.

Update Automated Opponent World keeps changing the Automated Opponent's model of the SSN situation in response to SSN changes sent to it.

Generate Automated Opponent Orders generates SSN-format orders in response to the current TAO world model, using the Automated Opponent knowledge base.

SSN is the simulation running on the personal computer. It accepts "orders" from the student and opponent and then computes the effects of the orders in the simulated environment.

3.5 Conclusions

Our experience with implementing a personal, computer-based, tactical environment simulation led us to

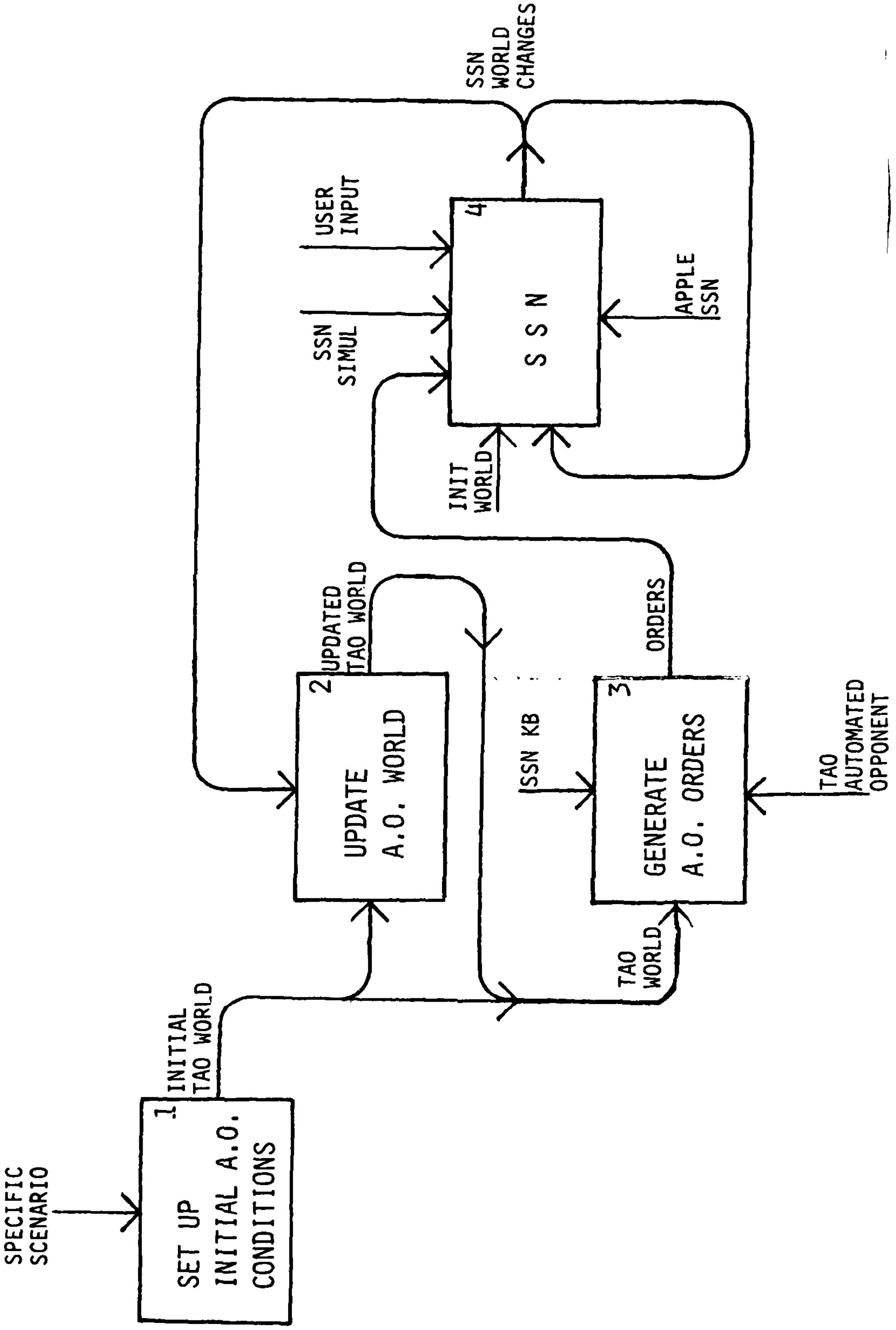


Figure 3.14 Program View -- Decomposition of Play SSN

draw several conclusions:

- The current generation of personal computers (Apple II, TRS80I, II, III, etc) are barely adequate to implement interesting tactical simulations. The primary problem is insufficient memory, both internally and in the limited capacity of floppy disks.

- The next generation of personal computers clearly will be adequate for tactical simulation training.

- However, development of simulation software should be done on larger systems. Both this generation and the next generation of personal computers are inadequate for serious development.

- Personal, computer-based simulation can be very cost-effective in the training environment. It should be pursued aggressively by the armed services.

In the area of automated problem generation, we offer several conclusions:

- Automated problem generation is probably feasible if a very well structured knowledge representation is chosen.

- Acquisition of highly structured knowledge bases from non-programmers is a reasonable goal. This would be very valuable in automating many training functions.

- Further effort in knowledge-based systems for training should emphasize an integrated systems approach, rather than attack isolated problems.

- The knowledge representation used in our work (HIPRS) deserves further exploration.

- Automated problem generation requires a large computer system. It is clearly not feasible on today's personal computers.

4. BIBLIOGRAPHY

Friedell, Mark; and Schmolze, Jim. ICDL Reference Manual -- Version 1.0 [DRAFT]. Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 02139. 15 April 1980.

Herot, Christopher F. "Spatial Management of Data." ACM Transactions on Database Systems," 5, 4 (December 1980) 493-513.

________. "A Spatial Graphical Man-Machine Interface." Information Processing 80. Edited by S. H. Lavington. North-Holland Publishing Company (1980).

Herot, Christopher; Carling, Richard; Friedell, Mark; Kramlich, David; and Rosenberg, Ronni L. "Overview of the Spatial Data Management System." Technical Report CCA-81-08, Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 02139. To appear.

Herot, Christopher; Carling, Richard; Friedell, Mark; and Kramlich, David. "A Prototype Spatial Data Management System." ACM Computer Graphics, 14, 3 (July 1980). Proceedings of SIGGRAPH '80, July 1980, Seattle, Washington.

Herot, Christopher F.; Carling, Richard; Friedell, Mark; and Kramlich, David. "Spatial Data Management System." Proceedings of the 1980 Office Automation Conference, March 3-5, 1980, Atlanta, Georgia.

Herot, Christopher F.; Carling, Richard; Friedell, Mark; Kramlich, David; and Thompson, John. "Spatial Data Management System: Semi-Annual Technical Report." Technical Report CCA-79-25, Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 02139. 30 June 1979.

Herot, Christopher F.; Schmolze, Jim; Carling, Richard; Farrell, Jim; and Friedell, Mark. "Spatial Data Management System -- Detailed Design Document." Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 02139. 6 October 1978.

"Spatial Data Management System." 1/2" videotape. Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 02139. 30 January 1980.

"SQUEL Reference Manual." Draft report. Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 01239, 18 January 1980.