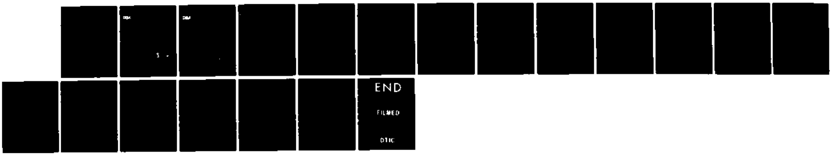
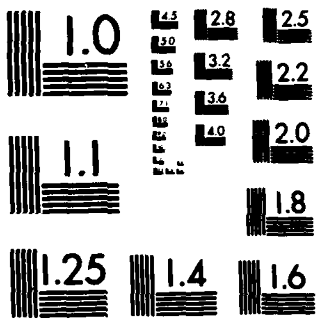
AD-A150 421 ACQUISITION AND LOGISTICS INFORMATION AND ANALYSIS SYSTEM (ALIAS) EXECUTIVE SUMMARY(U) DECISION-SCIENCE APPLICATIONS INC ARLINGTON VA J C KRUPP 31 OCT 84 1/1

UNCLASSIFIED DSA-605 N00014-82-C-0813 F/G 15/5 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)

DSA ***DECISION-SCIENCE APPLICATIONS***

AD-A150 421

DSA Report #605

31 October 1984

EXECUTIVE SUMMARY--ACQUISITION AND LOGISTICS INFORMATION AND ANALYSIS SYSTEM (ALIAS)

FINAL REPORT

Prepared under Contract N00014-82-C-0813 for:

Scientific Officer
Navy Office for Acquisition Research
NAVMAT 08
Washington, D.C. 20360

Attention: Dr. Thomas C. Varley

Suite 1000, 1901 North Moore Street, Arlington, Virginia 22209 (703) 243-2500

DSA DECISION-SCIENCE APPLICATIONS

DSA Report #605

31 October 1984

EXECUTIVE SUMMARY--ACQUISITION AND LOGISTICS INFORMATION AND ANALYSIS SYSTEM (ALIAS)

FINAL REPORT

J. KRUPP

Distribution unlimited per Dr. Thomas C. Varley, Navy Office for Acquisition Research, ATTN: NAVMAT-08, Washington, DC 20360

Accession For		
NTIS GRA&I	X	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution/		
Availability Codes		
Dist	Avail and/or Special	
A-1		

Prepared under Contract N00014-82-C-0813 for:

Scientific Officer
Navy Office for Acquisition Research
NAVMAT 08
Washington, D.C. 20360

Attention: Dr. Thomas C. Varley

Suite 1000, 1901 North Moore Street, Arlington, Virginia 22209 (703) 243-2500

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS
N/A

2a. SECURITY CLASSIFICATION AUTHORITY
N/A

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE
N/A

3. DISTRIBUTION/AVAILABILITY OF REPORT
Direct
(Naval Sea Systems Command (SEAOOD))
Washington, D.C. 20362

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
DSA Report No. 605

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Decision-Science Applications, Inc.

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
Navy Office for Acquisition Research

6c. ADDRESS *(City, State and ZIP Code)*
1901 N. Moore Street, Suite 1000
Arlington, Virginia 22209

7b. ADDRESS *(City, State and ZIP Code)*
NAVMAT 08
Washington, D.C. 20362

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
Office of Naval Research

8b. OFFICE SYMBOL *(If applicable)*
ONR

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
N00014-82-C-0813

8c. ADDRESS *(City, State and ZIP Code)*
Department of the Navy
800 North Quincy Street
Arlington, Virginia 22217

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
N68342	98077Q		

11. TITLE *(Include Security Classification)*
Executive Summary--Acquisition & Logistics Information & Analysis System (ALIAS)

12. PERSONAL AUTHOR(S)
J. Krupp

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Final	FROM 10/82 TO 8/84	31 October 1984	15

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB. GR.

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
ALIAS, DATA BASE, DBMS, MODELS, STRUCTURED PROGRAMMING

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The Acquisition and Logistics Information and Analysis System (ALIAS) is used by NAVSEA 90 to improve ship acquisition planning so that programs are on time, within budget, and fit into a fleetwide programming plan. ALIAS consists of 1) a large, integrated relational data base, 2) menu command system which allows the user to perform any task through use of menu selection or direct command, 3) extensive help subsystem which guides the novice or experienced user, 4) libraries of utility procedures, and 5) a variety of high level analytical and functional modules which interface with both the command system and the data base. The system currently resides on the Navy's HP-3000 computer housed at PMS-392. It makes extensive use of the RELATE data base management system. A companion applications generator package, BUILDER, is also used extensively. Most utility routines and high level modules are written in HP FORTRAN, while the user interface is written in BUILDER.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☒ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
Joseph C. Krupp

22b. TELEPHONE NUMBER *(Include Area Code)*
703-243-2500

22c. OFFICE SYMBOL
/DSA

DD FORM 1473, 83 APR — EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED

TABLE OF CONTENTS

ABSTRACT

Decision-Science Applications, Inc. (DSA) has created, installed, and tested a system for the Navy termed the Acquisition and Logistics Information and Analysis System (ALIAS). This analyst information system is used by NAVSEA 90 to improve ship acquisition planning so that programs are on time, are within budget, and fit cohesively into a fleetwide programming plan. ALIAS provides the Navy with the tools needed to help identify poorly planned programs quickly, and to allow for quick analysis of alternative program schedules. It consists of 1) a large, integrated relational data base, 2) a menu command system which allows the user to perform any task through the use of menu selection or direct command, 3) an extensive help subsystem which guides the novice through the system or provides the experienced user with a quick reference guide, 4) several libraries of utility procedures, and 5) a variety of high level analytical and functional modules which interface with both the command system and the data base. Modules communicate with each other only through the data base allowing for quick and easy insertion of newly created modules. An extensive data base updating system was installed to ensure data integrity. A scenario system is employed to allow any variety of "What if?" questions without the need for multiple copies of the data base ensuring quick turn around with data integrity and security.

The system currently resides on the Navy's HP-3000 computer housed at PMS-392. It makes extensive use of the RELATE data base management system from CRI, Inc. The companion applications generator package, BUILDER, is also used extensively. Most utility routines and high level modules are written in HP FORTRAN, while the user interface is written in BUILDER.

1.0 BACKGROUND

The Navy devotes substantial resources to the planning and monitoring of shipbuilding programs. Individual ship class programs are planned and monitored by Ship Acquisition Project Managers (SHAPMs). Work at particular shipyards is supervised by on-the-spot Supervisor of Shipbuilding, Conversion, and Repair (SUPSHIPS) offices. Detailed surveys of yard capacities and capabilities are conducted as part of the contract award process.

The principal organization conducting long-range, overall shipbuilding program planning is NAVSEA 90. It is concerned not only with individual program plans, but how all programs fit together into a fleetwide program. Analyses performed by SEA 90 are designed both to determine the feasibility of fleetwide plans and to highlight potential future problems caused by the interaction of ongoing programs before they occur.

Planning activities within SEA 90 include the generation of a large number of alternative five- to thirty-year total program plans each year, each examining the implications of a particular set of assumptions, possible constraints, or policies. These are performed as part of the Program Objectives Memorandum (POM) call and for the Five-Year Development Plan (FYDP). Mobilization and force buildup studies are conducted, and methods to promote maintenance of an acceptable level of industry capacity and to encourage better industry performance are considered. A large number of varied studies are performed on issues of policy interest, such as on submarine construction alternatives.

The responsibilities and staff capabilities of SEA 90 have made it the primary developer and reviewer of ship acquisition policy alternatives in the past few years. As a result, many DoD agencies and legislative staffs use the results of SEA 90 analyses as well as the CNO. In the past, SEA 90 depended on planning and estimation techniques developed either by other offices for their own purposes, or on ad hoc analyses in response to short term requirements. This situation inhibited the productivity and effectiveness of the planning and monitoring process. Without an appropriate, explicit framework, requests for analyses were often met by relatively laborious construction of models for one-time use, using procedures and data borrowed from other offices with underlying assumptions not always fully documented by the providers.

In August of 1982, therefore, a project was started to provide SEA 90 with the tools and data base required to perform its mission in a timely manner. This project, the creation of the Acquisition and Logistics Information and Analysis System (ALIAS), took two years and resulted in a general purpose platform which is state of the art and can easily grow with time as needs and policy change.

2.0 SYSTEM DESIGN

2.1 The Problem

The objective of SEA 90 analyses is to improve ship acquisition planning so that programs are on time, are within budget, and fit cohesively into a fleetwide programming plan. The goal of ALIAS is to provide the Navy with tools to help identify poorly planned programs quickly, and to allow for quick analysis of alternative program schedules.

2.2 The Approach

The structure of the shipbuilding data base and the specific SEA 90 requirements determined the ALIAS implementation approach.

The data needed by SEA 90 are not specific to program planning, but are of general interest. Also, the amount of data required is quite large. Accordingly, it was decided to design a data base which could be independently queried and whose structure was decided by the data and not by any computer program or module which would use the data.

A major SEA 90 requirement was for a flexible, responsive system capable of generating results to support many "What if?" types of questions. It was decided that the best approach for fulfulling this requirement was to use models which are known as outcome calculators. These models are used to generate answers to very specific types of questions. For example, one outcome calculator may answer the question "Is the following program feasible given the current yard resources?". This approach was selected because the outcome calculators could be quickly implemented and are essentially independent, thus allowing some system features to be used long before the system was totally complete.

The outcome calculator approach is difficult to use in sensitivity analyses since the user is required to formulate the exact programs being compared rather than just specifying a change in one or more 'tuning' parameters. The addition of a large-scale simulation or optimization model which could perform policy or sensitivity analyses automatically was provided for in the ALIAS design. Future large-scale models will be able to interface with the outcome calculators already created.

3.0 IMPLEMENTED ARCHITECTURE

Figure 3-1 diagrams the final ALIAS design. The design is best described as a wheel and spoke architecture with the user at the center. Within the diagram, circles represent interfaces while straight lines indicate that no interface is allowed. Note that the data base is what ties everything together. All modules communicate with one another only through the data base. This feature allows complete modularity of the system. Any module may be easily plugged into the system with a small amount of effort. In practice, it takes about one hour to add a newly completed module to the ALIAS system.

3.1 The Data Base

Because everything is tied to the data base, the design of the data base itself was crucial. A relational Data Base Management System (DBMS) was employed to manage the data base. For the current ALIAS system, the RELATE DBMS was used since it operates on the host HP-3000 computer system, was flexible enough to handle a data base of the size required by ALIAS, and incorporated some of the security features required by the system. The companion application generator language, BUILDER, was a further reason for using the RELATE package.

Great care was taken to ensure that the data base was properly placed into a normalized relational structure. The user himself is not allowed to access the data base directly except through the user interface (menu and command system in the right-hand circle) or through the DBMS interactive query system

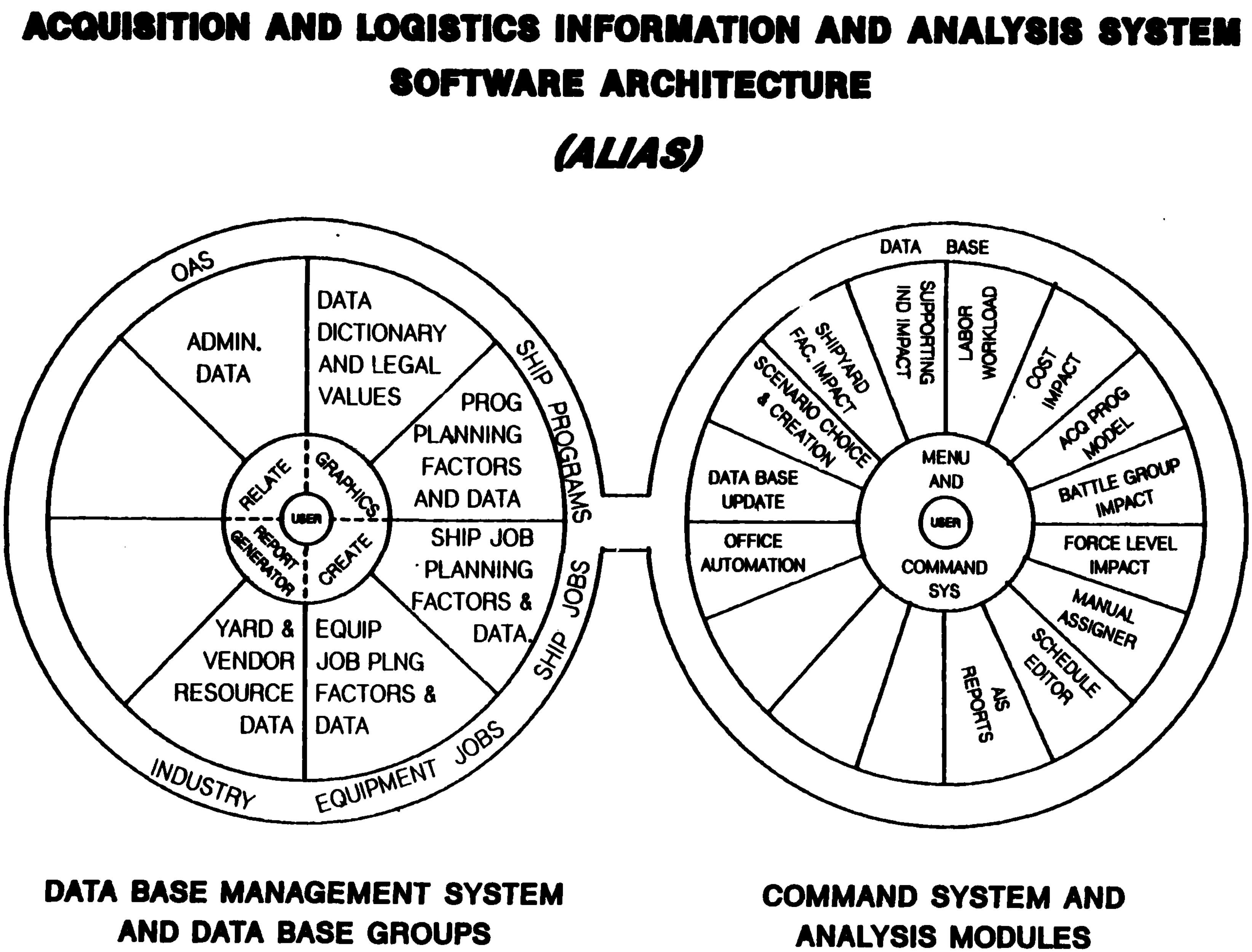


Figure 3-1: System Architecture

(RELATE in the left-hand circle). Security controls implemented through RELATE ensure that the user cannot indiscriminately change, add, or delete data (unless the user has a high privilege available only to the Data Base Administrator (DBA)). All changes to the data must be made through the data base updater (DBU), seen as one of the modules in the right hand circle. The creation of the DBU was actually that part of the system which consumed the most project resources. It makes certain that any changes to the data base conform to integrity standards.

For convenience and manageability, the data base is divided into several sections of related files, as shown in Figure 3-2. Large boxes represent sections, each of which is implemented in a separate HP-3000 file group named after the section. The relations (files) which comprise each group are listed within the boxes. Labels on the lines drawn between the boxes and the central circle (the user) indicate the principal retrieval keys for the sections, although any field in a file may be used as a retrieval key.

As the diagram indicates, the data base is quite extensive, consisting of over 100 files, and is expected to grow steadily. Descriptions of the data base at the section, file, and field levels are contained in a custom data dictionary which forms a small data base of its own (see below).

The ALIAS data base has a normalized design. No data element appears in more than one place unless it must function as a key. This structure, and the fact that all modules use the data base for input and output, ensures that the ALIAS system outputs are mutually consistent. An analyst can be confident that a cost estimate and a force level estimate made for the same projected shipbuilding program will in fact be using the

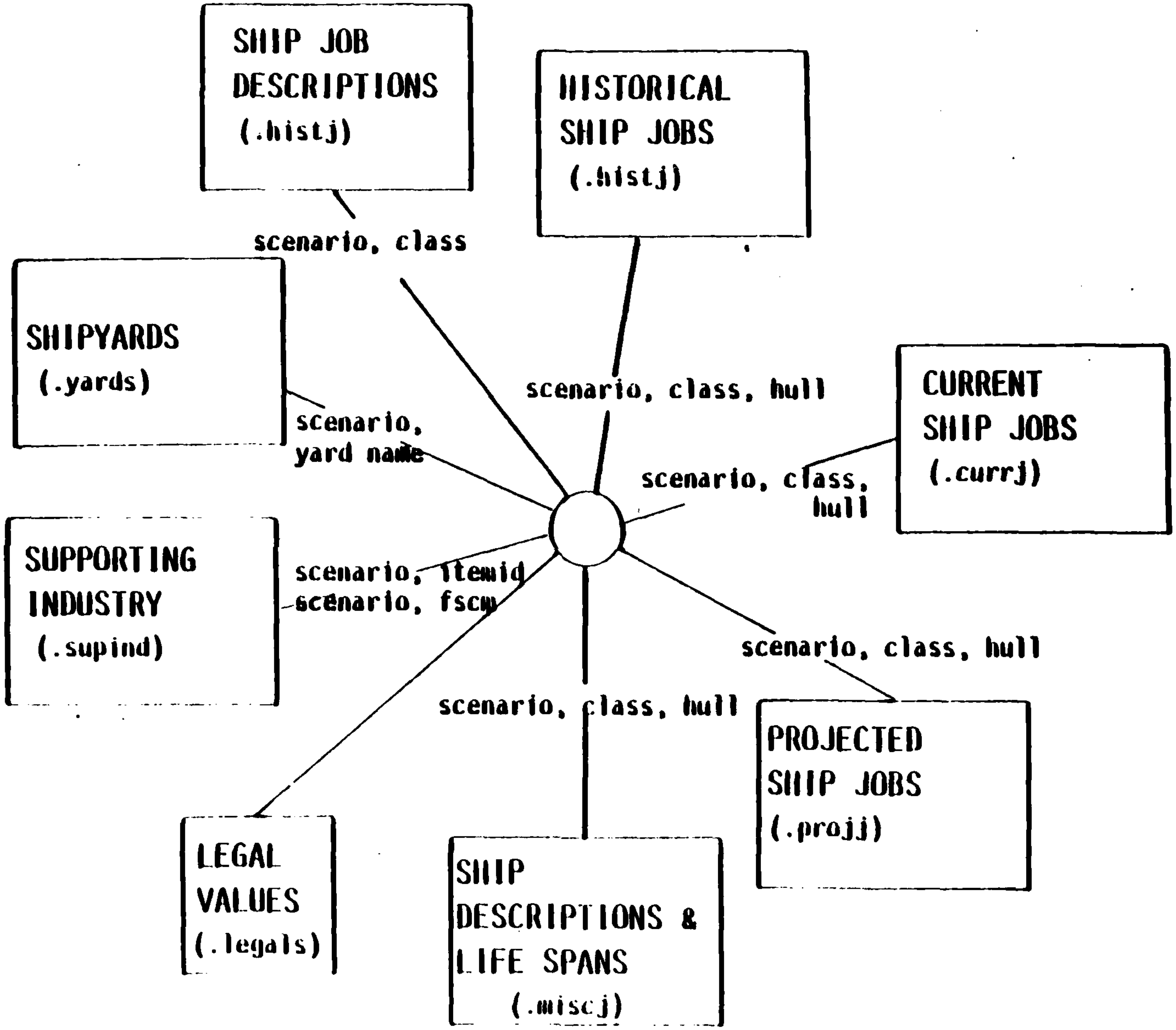


Figure 3-2. Data Base Structure

same program structure as their basis.

3.2 The Data Dictionary

The data dictionary (and legal values module), seen in the left hand circle of Figure 3-1, is used to ensure data integrity. When the user adds some data via the DBU, the DBU checks that the data is consistent with the current data base. For example, when a shipyard is given, that yard must be present in the data base. There are actually three types of data which are checked in this process. The first type of data is a required field, which simply means that some data must have been entered for that field. The second type of data is a legal field. This type of data is one whose value must be one of a given set of values. For example, physical units must be spelled the same throughout ("in" instead of "inch", etc.), states must be of the standard Post Office abbreviation, etc. The third type of data is called a key field. Whenever a key field is entered, the data dictionary is checked to ensure that all of the "joins", i.e., the combination of fields among several tables in the database, make sense with the value given by the user.

3.3 The Scenario System

One of the most important features of ALIAS is its scenario system. It is the scenario system that allows ALIAS to be both flexible and yet easily maintainable. The scenario system allows the user to ask numerous "What if?" types of questions while maintaining data base integrity (and also not causing a proliferation of multiple copies of the data base).

The system is implemented as follows. All data tables or relations contain a field labeled "scenario". When a user wants to see what affect changing some data might have, he may create a new scenario at will. This scenario is created in a combination of three different modes. For data which the user does not want to change, for example yard descriptions, all that is performed is the addition of a single record to one table informing the system that when the user wants this data for his new scenario, the system should use the data from an existing scenario (usually scenario MAIN). No data is copied, and the user is not allowed to change any of this data since it really belongs to another scenario. One nice feature about this method is that the user will always have the latest data. For example, if a new yard has been added to the main data base, this would be available to his scenario automatically since he is really just using the main data base when he calls up his scenario.

If the user does want to change some of the data, for example planning factors, he has the option of starting fresh or of copying the data from an existing scenario to his own. In this way the user creates a scenario consisting of 1) pointers to the primary data base (or any other scenario within the data base), 2) data tables consisting of only that data which he himself has input, and 3) data tables which are copies of data used by another scenario with the changes that he needs for his study.

Once again, the DBU ensures that all runs smoothly. For example, when a scenario is deleted the DBU checks to make sure that no other scenario depends on the one being deleted, and if so that the relations are preserved with the scenario name changed from the one being deleted to the one dependent on the data. The user of the dependent scenario is unaware of this change, of course, but is another reason why it is essential that no one outside of the DBA be allowed to change data except through the DBU.

3.4 Menu Command System

The ALIAS system is extremely user friendly. The user may receive help at any level with the pressing of one or two keys, and yet is not burdened with extensive on-screen displays after he becomes proficient with the system. The menu command system is a combination of a strict menu system, whereby the user makes selections from a predefined list of items, and a command system whereby the user exsplicitly types what it is he wants done. The user can, if he wants, stay completely within the menu structure, going from menu to menu in their logical order, or can, at his option, override the menu system and go directly to the module of interest.

There are three types of menus used within the menu command system of ALIAS: choice menus, list menus, and parameter menus. Choice menus are those which allow the user to choose a particular action, or to exercise a particular module. List menus are those which allow the user to choose which sets of data are to be acted upon, e.g., which job type (refuel, repair, new construction, etc.). Parameter menus are those which allow the user to set parameters which control the ALIAS system, e.g., environment parameters which inform ALIAS of the type of terminal the user is using.

Subsidiary to the menu structure of ALIAS is a series of help screens. At any time the user may type the ESC key followed by a question mark and receive a help screen explaining the current menu. Sometimes the help screen is another menu offering help about a variety of subjects. The user is free to examine the help screens at will and can return to where he started with the pressing of a single key. In addition to help screens about the current menu, additional help screens are available for any data item which the user must enter. These screens explain what the data item is and whether the item is a key field, legal field, a required field, or some combination of the three.

The menu and help structure at the higher levels of the system is completely data driven. Adding a new choice to a choice menu is as simple as adding a line to a data file. When a new module is added, help screens, list menus, and parameter menus may all be added simply by including them within this one data file, and exercising a preprocessor which reads the file and creates automatically the necessary data files, relations, and FORTRAN source code.

At a lower level, the creator of a module often wishes to include subsidiary menus and help screens. This is usually accomplished within the module itself through the use of the BUILDER application language of the RELATE DBMS. Templates are available to the programmer to facilitate the addition of these menus and help screens. Further information concerning this may be found in the ALIAS Maintenance and Expansion Guide.

4.0 CURRENT STATUS

The ALIAS core system, including data structures, security, DBU, data dictionary, and command system are complete, making it relatively easy for new modules to be added. In addition, a large number of outcome calculator type modules have been added to the system. The primary way that ALIAS is being used by SEA 90 at present is the following. A program is input using the assigner module. The system then generates ship schedules using key events (such as keel laying, launch, etc.).

Force structure projections are then made by the system based on these schedules, and various reports, such as a force level report or a battle group report, are created based on these projections.

The addition of more modules would be useful. A particularly useful module would be one allowing NAVSHIPSO to automatically perform industrial load studies (e.g., the flow of AEGIS control systems). While the ALIAS system supports the data needed to perform these studies, there is no module which automatically performs them and creates a report.

The system currently resides on an HP-3000 computer, housed in PMS-392. This small computer is taxed to its limit by the ALIAS system. Moving the system to a more capable computer will improve the turn-around time of the system and allow a variety of new modules to be added.

5.0 CONCLUSION

The ALIAS system is a modular, easily expanded system. It currently contains a variety of modules for force impact analysis, shipyard assignments/modifications, gross level cost analysis, general calculations, reports, graphics, queries, etc. It is functional, friendly, easy to use, and gives the analyst a high degree of control. It provides for both flexibility and consistency of results. The system is currently being used by SEA 90 for a variety of analyses, and there are plans for future expansion of the system and for movement of the system to a larger computer.

Use of this system will allow future models and reports to be added with a minimum amount of time and money, and will ensure that all results will be consistent with each other. Furthermore, maintenance of the data base will ensure that the Navy will have a central place for ship construction and repair data which is easily accessible and will preclude the problems inherent with multiple sources of data.

END

FILMED

3-85

DTIC