DEPARTMENT OF COMPUTER SCIENCE
COLLEGE OF SCIENCES
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23508

USER INTERFACE FOR INTEGRATED COMPUTER AIDED DESIGN SYSTEMS

By

James L. Schwing, Principal Investigator

Final Report
For the period January 1, 1983 to May 31, 1985

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665

Under
Research Grant NCC1-74
Alan W. Wilhite, Technical Monitors
SSD-Vehicle Analysis Branch

November 1986

DEPARTMENT OF COMPUTER SCIENCE
COLLEGE OF SCIENCES
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23508

USER INTERFACE FOR INTEGRATED COMPUTER AIDED DESIGN SYSTEMS

By

James L. Schwing, Principal Investigator

Final Report
For the period January 1, 1983 to May 31, 1985

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665

Under
Research Grant NCC1-74
Alan W. Wilhite, Technical Monitors
SSD-Vehicle Analysis Branch

Submitted by the
Old Dominion University Research Foundation
P.O. Box 6369
Norfolk, Virginia 23508

November 1986

TABLE OF CONTENTS

1. Introduction

The purpose of this research was the development of a user interface and other appropriate tools to be used in Computer Aided Design (CAD) systems which integrate a wide variety of independently developed design and analysis tools.

Specifically, the interface was intended for the integration of programs to be used in the conceptual design of aerospace systems. By way of motivation consider the following factors typical to the make up of such a system. First conceptual design involves consideration of all aspects of the ultimate vehicle or structure. This includes such things as geometry definition, structural and thermal dynamics, trajectory analysis, weights and measures, economic and mission analysis and more. Development of computer software for each of these areas would be a major undertaking in itself. Thus in developing such a CAD system, it is typical to find that a group will search for and select the best available software to address each of the areas needed for the overall analysis. In general this means that the software packages are brought in from a wide variety of sources and were never intended to be integrated together.

This diversity leads to several problems. Each of the packages may have a different formats for the input and output of data. Units of measure will vary. Although there is an interdependence of program data there is no standard method for transferring that data. Packages will have a wide range in the way they address errors, from non-existent to program controlled. Similarly, the help facilities will have a wide range, from non-existent to an on-line user's manual. In addition, the definition and/or modification of input data is frequently complicated. For example, a package may have thousands of input values but only a few need to be modified.

Each of the above cited problems and others, put a high burden on the users of such a conceptual design system. Users must learn multiple command formats, editors, input/output techniques, help and error facilities. In order to get the various packages to interact with one another, the user must be more than an engineer with a basic programming background. The user must have the knowledge of an experienced system programmer.

What follows describes the development of a user interface and associated tools which address the above listed problems. The system is known here as the Aerospace Interactive Design Environment (AIDE). Also included is a copy of the user's manual for the system.

2. Methodology

As expressed in Newman and Sproul [1] the design of the user interface breaks down into four basic components..

1. The User Model
2. The Command Language
3. The Feedback
4. The information Display

The user model should be based upon the user's conceptual picture of the tasks which must be accomplished to achieve a desired result. The development of the user interface must attempt to include this picture as its foundation. The essence of the user model can be summarized by two basic sets. The first set is a collection of objects which the user wishes to create and/or manipulate. To complement that set is a collections which the user envisions for carrying out the intended manipulations.

From the start it was obvious that the development of a system for controlling independently designed programs would be complicated. Such developments are eased by the application of software engineering principles. In particular, the following phases lead to a well structured, easily modified and maintained software system.

1. Requirements Definition and Specification
2. Design of the System
3. Implementation of the System
4. Testing
5. Maintenance

As it will be seen later, the development of this system involved more tools than just the user interface designed under this grant. In order to coordinate the development of these tools, a program integration team was formed and became responsible for monitoring the application of the software engineering principles to each portion of the design. This team consisted of the following members: Alan Wilhite (NASA), Larry Rowell (NASA), Jim Schwing (ODU), Greg Wright (ODU), Don McMillan (ODU), Kenny Jones (CSC) and Don Randall (CSC).

In any case the first step in deriving the user interface involved deciding upon the requirements in light of a user model. As a starting point, this research focused on an existing preliminary design system AVID and its users. An overview of the AVID system is given by Wilhite [2]. As an aerospace vehicle design system AVID has the ability to interact with many independent design and analysis programs, beginning with those for the definition and design of geometry. "A preprocessing program has been written to retrieve design data from a database. Using this design data, preset constant data from the preprocessor program and data input by the user, the

preprocessor program constructs an input file in the exact format that the analysis program expected before it was integrated into the AVID system. If the correct data is not in the database, the user is notified. After the input file is created, the analysis program is executed and the desired design data is stored in the database to be retrieved by other analysis programs. ...

"The archive system stores all the design data and bulk data files that pertain to one configuration. ... This capability is used to freeze the design so that the engineer can break for an interruption and return at a later time."

Additionally an executive command language was envisioned having the following description. "The main purpose of the executive is to allow the user to interactively execute technology programs without having to know computer languages or computer commands. The executive displays a menu of analysis or utility programs. After making the appropriate selection, the user may be asked to respond to questions about data input to insure that the correct information has been retrieved. The executive then steps through a series of computer commands that comprise one step in the design process. After finishing that design step, the menu is again displayed to the user for selection of the next step. If a definite pattern emerges from those single design steps, the executive can easily be programmed to perform a sequence of design steps automatically. This function is useful for design iteration and also for parametric studies."

3. Requirements Definition and Specification

Using the description cited above as a starting point, the requirements for the user interface were generated via a task analysis of the users of AVID. A summary of those requirements follows.

1. Core User Interaction

The word "core" here represents a specialized subset of the total interface defined specifically for introductory users. This "core" should satisfy the following properties.

a. Simplicity - The size and complexity of the "core" should be no more that a relatively inexperienced user can master in a few days.

b. Completeness - There should be no glaring omissions from the "core", especially with respect to operations and analysis which are both obvious and natural to the engineer user.

c. Robustness - Any desired model should be definable and any analysis applied to within the "core" commands.

2. Advanced User Interaction

The following properties are included to enhance the effectiveness and limit the frustration of advanced users.

a. The system should provide the ability to forgo any menus and queries that become obvious and time consuming.

b. The system should provide the ability to customize the computing environment to the user's needs and tastes. For example. it should provide the ability to define specialized commands and terminal modes.

3. The interface should automatically provide for the communication of input data to all programs.

4. The user should have the ability to select any application program for execution.

5. The user should have the ability to select only certain inputs for review and modification. That is, the user need not be required to redefine input data values that have not changed since the last analysis.

6. The interface should provide for execution and control of the associated analysis programs.

7. As with input the user should be able to define a set of critical results for review. All results should be automatically communicated to the database.

8. The interface should provide the ability for the user to define frequently used execution sequences as a single design step.

9. A help system should be implemented which includes not only information about the interface and its commands but also information concerning program purposes and variable meanings.

10. During execution the system should provide a local database, so that original "master" or "golden" databases are not corrupted during execution.

11. The system should check for missing or potentially wrong or illegal data.

12. The system should supply default baseline data if required.

13. The interface should maintain a record of all user actions and system responses.

14. The interface should provide consistent error handling for both user errors and analysis program errors.

15. The interface should provide feedback on the status of the user's environment.

4. Design of the System

From this set of requirements and in conjunction with the program integration team, a CAD system for integrating independent programs together was developed. The following is a list of components identified as necessary for the implementation of such a CAD system.

1. Database Manager
2. Data Dictionary
3. Program Dictionary
4. Data Templates
5. Reviewer
6. Formatter
7. User Interface

In order to get a feel for the design of the system, consider the following overview of the system. As mentioned earlier, the basis for the design of the systems lies in the user model. In the AIDE system the principle class of objects that the user will manipulate is the configuration database. The primary classes of actions that the user will apply to these objects are those of viewing and modifying the data and executing analysis programs using the data.

In the conceptual design process one of the first steps is the careful identification of the parameters necessary to define an appropriate design. It is this data that is to be collected into the configuration databases. In the early stages of design baseline data is set up as a starting point for further design. These special baseline databases are referred to in the AIDE system as

either "golden" databases or "master" configurations. In AIDE all other analysis refers back to a baseline case, in this manner automatic defaults can be supplied when the user decides not to define certain input variables. In addition, once the parameters used to define the configuration have been identified, it becomes appropriate to insert them into the data dictionary. The data dictionary is then used by the AIDE system during a help enquiry about a program variable.

To support the first of the classes of actions, viewing and modifying the data, a software tool called a reviewer is provided. Basically the reviewer works like the popular spreadsheet tools. For single parameters, the reviewer presents the current value of the variable along with a short description and the units of measure. Users select only those values that they wish to change. Presentation of tabular data is even more like a spreadsheet. Users simply indicate the row and column values they wish to change. Generally the data presented corresponds to all the data associated with the input or output of one of the analysis programs. However, users also have the opportunity to define exactly those variable in the database that they wish to have presented for review. Thus if only a few of the inputs are involved in a parametric study, the user can review and modify only those and the rest will default to the current values. The data template is the tool which is used to define a user's view of the database.

Support of the second major class of operations, namely execution of the analysis programs, is provided with the program dictionary, the formatter and once again data templates. Recall that one of the requirements mentioned in the previous section was that all programs should be run automatically. Two elements are necessary to execute an analysis program. First the location of

the program and all associated run files must be known. This information is provided in the program dictionary along with a listing of program help files. Secondly the values of input variables must be retrieved and presented to the program. Finally the output values must be captured and restored in the configuration database. Much like requested data being presented to the user with the appropriate data template, the proper input and output variables are defined for each analysis program in its set of data templates. The formatter provides an automated process for the creation of pre- and post-processors which interact with the data templates to fetch and store the data for each of the analysis programs.

The user interface provides a command environment through which the user controls the design process. Associated with any configuration that the user is developing is a workspace. This workspace tracks the current state of the design through a set of state variables, such as the name of the current configuration database and analysis program, and a log of commands issued and the system response. Commands may be issued to the interface in any of several modes. For the "core" user, a set of menus are presented while the more advanced user can interact with a command driven system. Finally several tools are provided in the interface for the user to customize the AIDE environment. Primary among these is the procedure builder. Procedures are basically collections of interface commands. Typically they represent a collection of programs used for given design step. This can be used to simplify design processes such as parametric studies where the procedure continually loops over the steps of the design process.

The final item mentioned as a tool but not included in the above discussion is the database manager. Obviously, this is used to access the data in

the configuration database. In addition, the data dictionary, program dictionary, data templates and AIDE system information are also contain in system databases. Thus the database manager affects every phase of the CAD system. In essence it is the glue that holds the system together. Yet it is the one tool that the user does not interact with directly. That follows since the user's access to the configuration data is through the powerful tool of the reviewer (to be sure to reviewer uses the database manager for its access to the system).

Interrelationships of the various portions of the system are given in figures 1 and 2 below. Figure 1 represents the end user's view that was derived from a task analysis of users of the AVID system. Figure 2 show the interaction of the various tools described above.

5. Implementation of the System

The AIDE system was implemented initially on a PRIME computer using the FORTRAN language. The AIDE system is currently being extended for release by NASA under the name of EASIE. A short discussion of this extension will be covered later.

In choosing a database manager for the system, it was decided to use the most efficient relational database system then available to the program integration team. Much consideration has been given to the type of database to use in any given situation and current recommendations go with the relational database for scientific and engineering data (see for example [3] and [4]). At

the time of design, the most efficient such system then available was the ARIS system. ARIS is described in [5].

The remainder of this section will be devoted to a list of the requirements for each of the software tools mentioned above along with an example of that tool as implemented. Detailed discussion of these tools can be found in the documentation of the EASIE extension of AIDE (see [6], [7], [8] and [9]).

Elements of the data dictionary.

1. A textual description.
2. Physical units.
3. Type (real, integer, ...).
4. Dimension (for arrays).
5. Location in the configuration database.

   This allows the reviewer and formatter to look up data values.

Figure 3 represents a page from the data dictionary of an AIDE system used for space station configuration (SSCDM).

Elements of the program dictionary.

1. Program name.
2. Library containing run file, source file and help file.
3. Run file name.
4. Mode of execution.
5. Input and output templates.
6. Cognizant person.

Figure 4 represents a page from the program dictionary for the SSCDM example.

Elements of the data template.

1. Template name.
2. Relevant relation in the configuration database.
3. List of items to choose from the relation listed in item 2.
4. Indicator of "where" attributes.

   For example this can be used to select all material with density less than a specified maximum.
5. Sort indicator.

   Much like item 4. It allows the chosen items to be sorted prior to presentation.

Figure 5 is a sample of the input data template for the atmospheric analysis program from the SSCDM example.

Elements of the reviewer.

1. Uses as input a given data template and one or two configuration databases.
2. Uses the template as a guide for the extraction or replacement of data values to the configuration database.

Figure 6 is a typical screen presented to the user by the reviewer, here part of the input data to the atmosphere analysis program is presented for review. Figure 7 pictorially represents the action of the reviewer.

Elements of the formatter.

1. Uses as input a given data template and configuration database.
2. Uses the template as a guide for the extraction or replacement of data values to the configuration database.

Figure 8 pictorially represents the action of the formatter.

Elements of the user interface.

1. The interface is menu or command driven.
2. The interface has user selectable display formats.
3. The menu has a user selectable experience level (core or advanced).
4. Command syntax:

   <command><delim><object type><delim><object name>

   For example:

   RVU IDB ATMOS

   Review the input database of the atmosphere analysis program.
5. The command set is semantically consistent. That is, regardless of the state of the interface a given command always has precisely the same meaning.
6. The total interface is relatively small and easy to learn.

   22 basic commands

   11 basic object types
7. All of the variable information concerning the interface is stored in a system database. This allows configuration of the system to any given CAD without the need to modify and recompile code.

Details of the AIDE user interface are contained in the AIDE USER GUIDE in the appendix.

6. Testing

The AIDE system has been put through two major tests. The first as has already been mentioned was the implementation of the space station simulation programs SSCDM. Additionally, during the conversion of AIDE to the VAX system and its new extensions now known as EASIE, the system has been tested as the controller to the design system IDEAS**2. In both cases, there have been three outstanding features of the tests. first the ease with which new programs can be integrated into the system without major modifications. Next, the utility of the data reviewer for definition and modification of user selected data. Finally, the utility of designing just the right procedures for each design situation, has made the design process all the more effective.

REFERENCES

1. Newman and Sproul, Principles of Interactive Computer Graphics, Second Edition, McGraw hill, New York.
2. Wilhite, A.W., S.C. Johnson and V. Crisp, "Integrating Computer Programs for Engineering Analysis and Design.", AAIA Conference Proceedings, 21st Aerospace Science Meeting, Reno, Nev., Jan. 1983.
3. Jacky, J.P. and I.J. Kalet, "A General Purpose Data Entry Program.", Communications of the ACM, v. 26, no. 6, pp. 409-417, June 1983.
4. Dube, R.P. and M.R. Smith, "Managing Geometric Information with a Database Management System.", Computer Graphics and Applications, v. 3, no. 7, pp. 57-62, October 1983.
5. Wilhite, A.W. and V. Crisp, A Relational Information System: ARIS User Manual, NASA/Langley Research Center, 1983.
6. EASIE Overview, NASA/Langley Research Center, to appear.
7. EASIE Installation Guide, NASA/Langley Research Center, to appear.
8. EASIE Programmer's Guide, NASA/Langley Research Center, to appear.
9. EASIE Maintenance Guide, NASA/Langley Research Center, to appear.

FIGURES

END USER'S VIEW - DERIVED FROM A USER TASK ANALYSIS

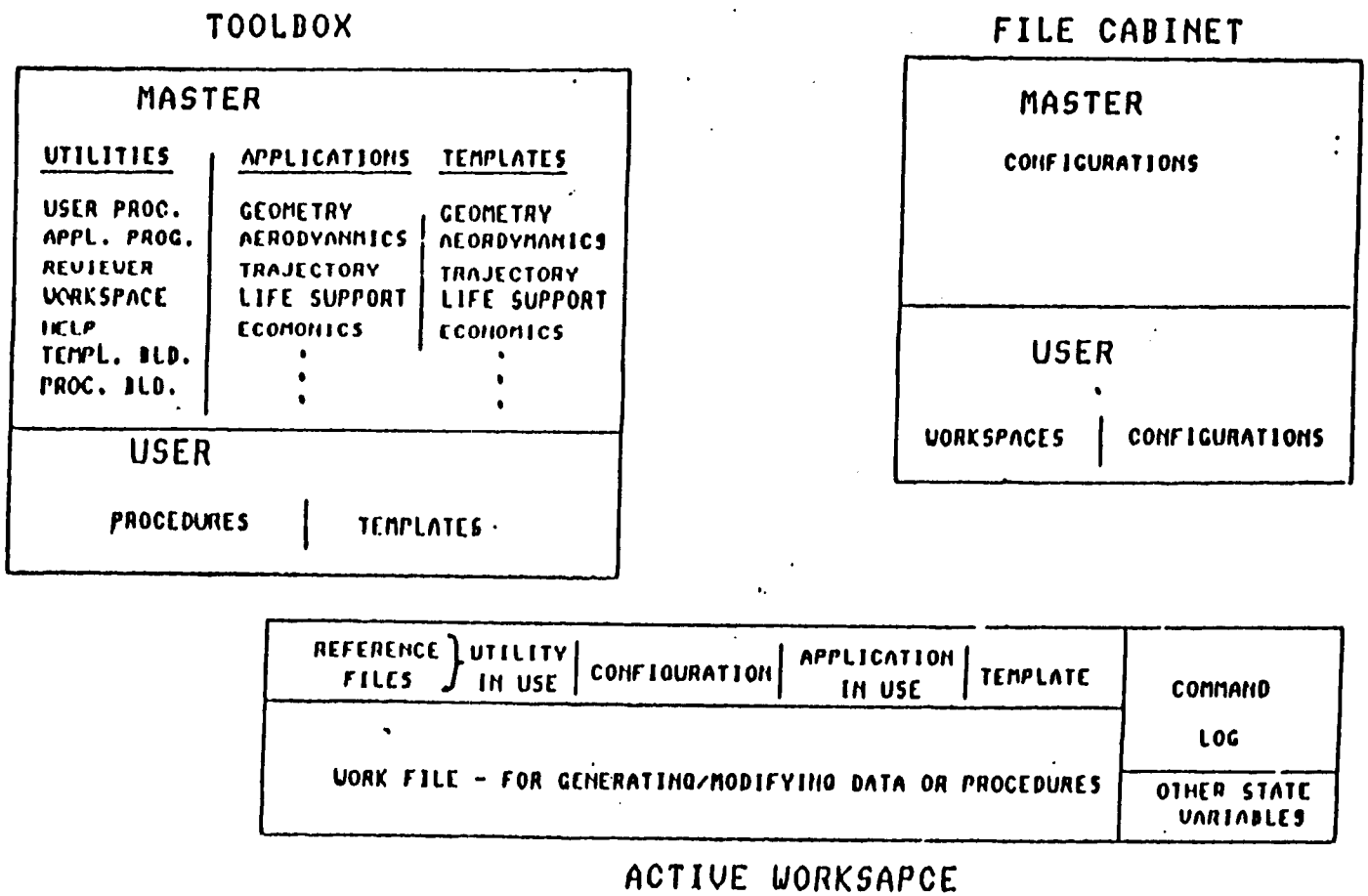

Figure 1

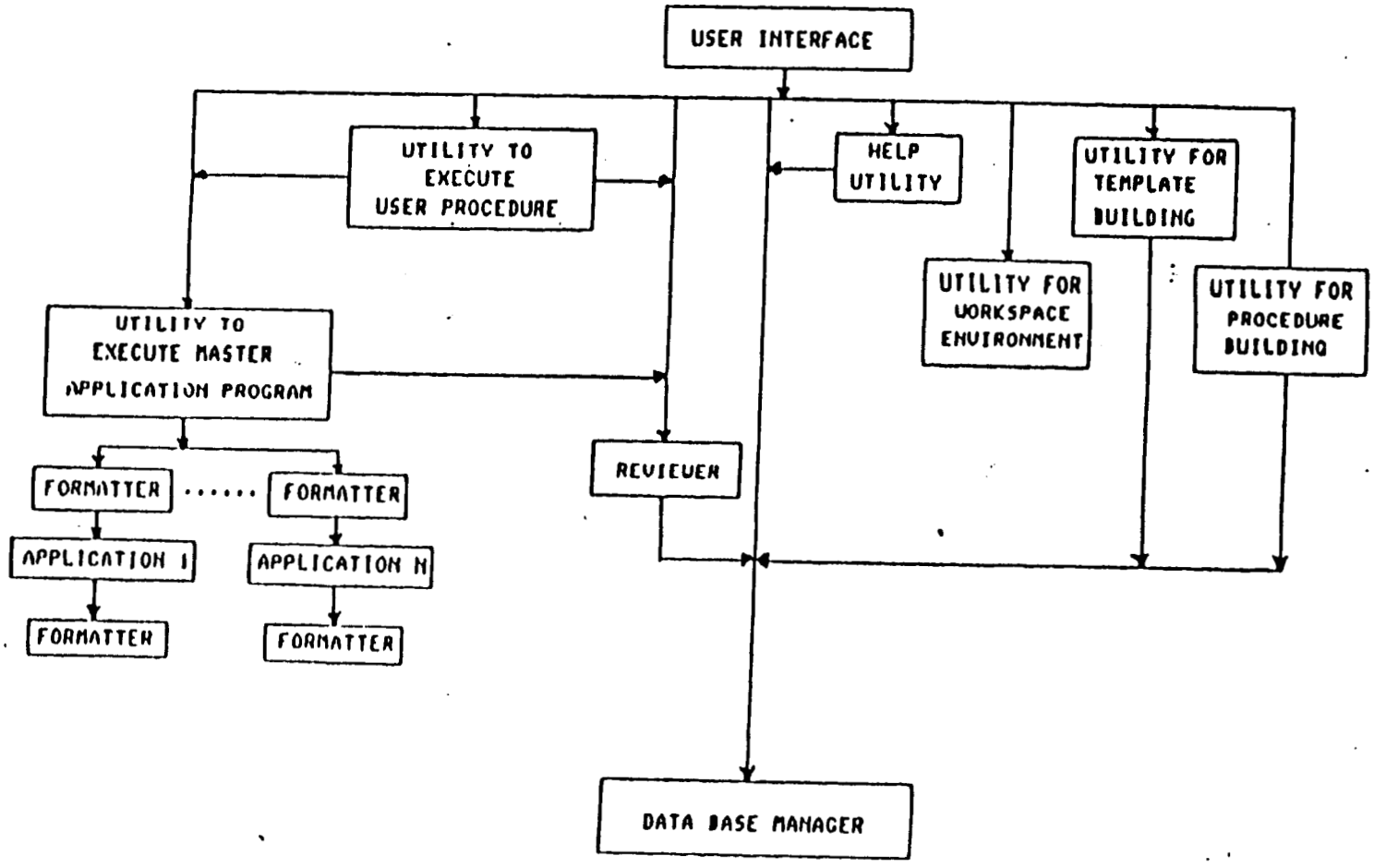
USER INTERFACE
UTILITY TO EXECUTE USER PROCEDURE
HELP UTILITY
UTILITY FOR TEMPLATE BUILDING
UTILITY FOR WORKSPACE ENVIRONMENT
UTILITY FOR PROCEDURE BUILDING
UTILITY TO EXECUTE MASTER APPLICATION PROGRAM
REVIEWER
FORMATTER
FORMATTER
APPLICATION 1
APPLICATION N
FORMATTER
FORMATTER
DATA BASE MANAGER

Figure 2

****
* RELATION PARAM *
****

RELATION TYPE	PARANAME NCHAR	ORDER DIM	DESCRIP FORMAT	UNIT
BLK01 REAL	A 0	27 12	AIRLOCK LENGTH	FT
BLK01 REAL	A0 0	4 1		
BLK01 REAL	A1 0	32 1	SCALING FACTOR FOR PARASITIC LOAD	-
BLK01 REAL	A4 0	23 1	EXPERIMENT MOUNTING RACK WIDTH	FT
BLK01 REAL	AA 0	30 1	FRACT. POW. FOR EMERG. ECLSS OPERATION	-
BLK01 REAL	AC 0	28 1		
BLK01 REAL	ACC 0	10 1		
BLK01 REAL	ACMAX 0	29 1	MAX. LIOH CANISTER CROSS SECTION AREA	IN2
BLK01 REAL	ACN 0	35 1		
BLK01 REAL	ACP 0	14 1		
BLK01 REAL	ACPC 0	10 1		
BLK01 REAL	ACPCP 0	21 1		
BLK01 REAL	ACS1 0	9 1		

HIT RETURN TO CONTINUE

Figure 3

```
**********************
*                    *
* RELATION PROGRAM   *
*                    *
**********************
```

SEQ#	PRCGNAME		LIB		RPROC
	PROCTYPE	TMPL_IN	TMPL_OUT	COG_PERS	
1	FOOD		VAB2>CSC		FOOD
	RMOD	FOOD.IN	FOOD.OUT	J L SCHWING	
1	CONFIG		VAB2>CSC		CONFIG
	RMOD	CONF.IN	CONF.OUT	J L SCHWING	
1	AGAPPL		VAB>JLS		AGPRE.SEG
	VMOD	AGTPL.IN	AGTPL.OU	A G WRIGHT	
2	AGAPPL		VAB>JLS		AGAPPL.SEG
	VMOD			A G WRIGHT	
3	AGAPPL		VAB>JLS		AGPOST
	VMOD			A G WRIGHT	
1	ATMOS		VAB2>CSC		ATMOS
	RMOD	ATMO.IN	ATMO.OUT	J L SCHWING	
1	CO2REM		VAB2>CSC		CO2REM
	RMOD	CO2R.IN	CO2R.OUT	J L SCHWING	
1	CONTAM		VAB2>CSC		CONTAM
	RMOD	CONT.IN	CONT.OUT	J L SCHWING	
1	ELECT		VAB2>CSC		ELECT
	RMOD	ELEC.IN	ELEC.OUT	J L SCHWING	
1	REACT		VAB2>CSC		REACT

Figure 4

```
                    **********************
                    *                    *
                    * RELATION TEMPLATE  *
                    *                    *
                    **********************

NUMBER   (  1)              1    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK01
TYPE     (  1)     PARA     #ITEMS  (  1)           1    ITEMNAME(  1)     A
NATTW    (  1)              0    NSORT   (  1)           0    I/O     (  1)     IN

NUMBER   (  1)              2    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK03
TYPE     (  1)     PARA     #ITEMS  (  1)           9    ITEMNAME(  1)     C
ITEMNAME(  2)     CAPKI         ITEMNAME(  3)     CAPKMI      ITEMNAME(  4)     CAPKWI
ITEMNAME(  5)     CAPM          ITEMNAME(  6)     CAPN        ITEMNAME(  7)     CAPNX
ITEMNAME(  8)     CAPP0         ITEMNAME(  9)     CAPPN       NATTW   (  1)               0
NSORT    (  1)              0    I/O     (  1)     IN

NUMBER   (  1)              3    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK04
TYPE     (  1)     PARA     #ITEMS  (  1)           3    ITEMNAME(  1)     D
ITEMNAME(  2)     DELTAT        ITEMNAME(  3)     DOTL        NATTW   (  1)               0
NSORT    (  1)              0    I/O     (  1)     IN

NUMBER   (  1)              4    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK05
TYPE     (  1)     PARA     #ITEMS  (  1)           5    ITEMNAME(  1)     E
ITEMNAME(  2)     EI            ITEMNAME(  3)     E2          ITEMNAME(  4)     ETA0
ITEMNAME(  5)     ETAN          NATTW   (  1)           0    NSORT   (  1)               0
I/O      (  1)     IN

NUMBER   (  1)              5    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK06
TYPE     (  1)     PARA     #ITEMS  (  1)           8    ITEMNAME(  1)     FL4
ITEMNAME(  2)     FL7           ITEMNAME(  3)     FLNTH       ITEMNAME(  4)     FNU10
ITEMNAME(  5)     FNU4          ITEMNAME(  6)     FNUS        ITEMNAME(  7)     FNU6
ITEMNAME(  8)     FNUM          NATTW   (  1)           0    NSORT   (  1)               0
I/O      (  1)     IN

NUMBER   (  1)              6    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK07
TYPE     (  1)     PARA     #ITEMS  (  1)           2    ITEMNAME(  1)     H0
ITEMNAME(  2)     HN            NATTW   (  1)           0    NSORT   (  1)               0
I/O      (  1)     IN

NUMBER   (  1)              7    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK08
TYPE     (  1)     PARA     #ITEMS  (  1)           1    ITEMNAME(  1)     IOPT
NATTW    (  1)              0    NSORT   (  1)           0    I/O     (  1)     IN

NUMBER   (  1)              8    TP.NAME (  1)     ATMO.IN     RELATION(  1)     BLK09

HIT RETURN TO CONTINUE
```

Figure 5

LB	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	6.00000	6.00000	1, 1, 1	A	AIRLOCK LENGTH	FT
2	6.00000	6.00000	1, 1, 2			
3	6.00000	6.00000	1, 1, 3			
4	6.00000	6.00000	1, 1, 4			
5	6.00000	6.00000	1, 1, 5			
6	6.00000	6.00000	1, 1, 6			
7	6.00000	6.00000	1, 1, 7			
8	6.00000	6.00000	1, 1, 8			
9	6.00000	6.00000	1, 1, 9			
10	6.00000	6.00000	1, 1, 10			
11	6.00000	6.00000	1, 1, 11			
12	6.00000	6.00000	1, 1, 12			
13	8.00000	8.00000		C	STATION CREW SIZE	-
14	0.350000E-04	0.350000E-04		CAPKI	CRYOGENIC TANK INSULATION CONDUCTIVITY	BTU/HRFTDR
15	0.347000E-05	0.347000E-05		CAPKMI	STAINLESS STEEL MATERIAL CONSTANT	-
16	0.210000E-01	0.210000E-01		CAPKUI	CRYO. ATMOS. STORAGE SYSTEM CONTR.POW.	KW
17	0.970000	0.970000		CAPM	METABOLIC WATER PRODUCTION	LB/MAN/DAY
18	26.0000	26.0000	1, 1, 1	CAPN	AIRLOCK OPERATIONS RESUPPLY INTERVAL	-
19	26.0000	26.0000	1, 1, 2			
20	26.0000	26.0000	1, 1, 3			
21	26.0000	26.0000	1, 1, 4			
22	26.0000	26.0000	1, 1, 5			
23	26.0000	26.0000	1, 1, 6			
24	26.0000	26.0000	1, 1, 7			
25	26.0000	26.0000	1, 1, 8			
26	26.0000	26.0000	1, 1, 9			
27	26.0000	26.0000	1, 1, 10			
28	26.0000	26.0000	1, 1, 11			
29	26.0000	26.0000	1, 1, 12			
30	0.500000	0.500000		CAPNX	REPRESSURIZATIONS PER RESUPPLY INTERVAL	-
31	736.000	736.000		CAPP0	CRYOGENIC O2 TANK PRESSURE	LB/IN2
32	492.000	492.000		CAPPN	CRYOGENIC LN2 TANK PRESSURE	LB/IN2
33	4.00000	4.00000	1, 1, 1	D	AIRLOCK DIAMETER	FT
34	4.00000	4.00000	1, 1, 2			
35	4.00000	4.00000	1, 1, 3			

Nn - Next page, R - Reprint page, n - modify LB n, Ln - n LB per page, E - End and savemods, Q - Quit, H - Help, PA - Print all

EDIT
>

Figure 6

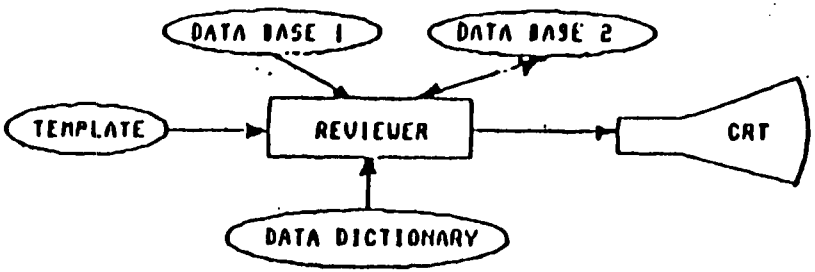

Figure 7

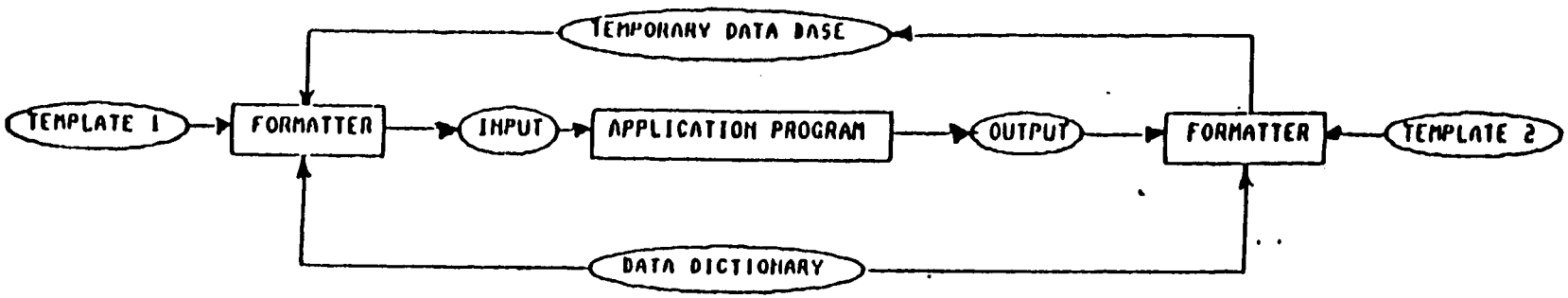

Figure 8

APPENDIX

AIDE USER'S MANUAL

Table Of Contents

Introduction

This document describes the "executive" capabilities provided by the AIDE system and defines the command language and menus available under Version 1.0. The sections have been designed so that a minimum of reading is needed to be able to begin using the system. Sections 2 and 3 provide an overview of terms and a sample session sufficient to get started. Sections 4 and 5 provide a fuller description of the system and a more complex sample session. Section 6 and 7 describe the use of the AIDE procedures and the database structures (files, workspaces, data dictionary, etc.). Section 8 is a quick reference for the full AIDE command language. The AIDE user interface is primarily intended to provide the designer the capabilities to examine input and output data and execute application programs; in addition, the interface provides a full set of system support tools. AIDE provides users with two basic modes of execution. The first is a menu driven execution mode which provides a novice user with sufficient guidance to quickly review data and select and execute application programs. Experienced designers can use this menu-driven mode to tailor the execution environment to meet the specific needs. The second mode of operation provides a full executive interface which allows the designer in-depth control of the design process. For example, when using this second mode, techniques are provided which allow the user to establish a design sequence and then automatically re-execute the sequence allowing the designer to iteratively refine input and review the output without repetitive computer manipulation. Users can switch between these modes at will and thus gain speed as their familiarity with the system improves.

Section 2
BASIC AIDE CONCEPTS AND DEFINITIONS

The most predominant design methodolgy uses the iterative technique. Here one progresses to a final solution through successive applications of analysis techniques to increasingly refined data. It is the purpose of AIDE to facilitate this process. With that in mind the most important concepts for the AIDE user are the configuration data, the application programs and the design setting itself.

2.1 Configuration Data

Configuration data (so named because the data will be descriptive of a single point design or configuration) is stored in a system-managed database. In concept, members of a design team will define, develop, and evolve the basic variables and constants required to describe a basic model configuration. Once this information is determined, it is entered into the database along with the necessary descriptive information. For example, a typical data entry might include the variable name, a short description of what the variable represents, the type of units to be used when referencing the variable, a default value if applicable and some computer-specific information such as data type (real or integer, scalar or array, parameter or attribute). The data is stored via a relational database manager. Several studies have shown [1,2] that this is more effective for the storage and retrieval of engineering data than typical hierachical or network designs. However, as it will be seen, it will not be necessary for designers unfamiliar with a relational database manager to learn how to use one.

Once the basic data definitions (serving as the starting point for design) have been made, a copy of this "golden" database is placed in a configuration-controlled project directory. Access to this database is provided on a "read only" basis. That is, designers may call up the configuration data for review, or they may make a copy of the database for their personal files. However, designers are prevented from making changes in the "golden" database. Updates to the "golden" database can be entered only by the design team leader or configuration manager. As a side note there may be several "golden" databases representing different benchmark configurations or versions of design.

There are two methods by which a designer may access his working copy of the configuration database. The AIDE interface provides a program called a "reviewer" which will access data for the designer. Essentially the designer indicates a desire to view configuration data related to, say, the inputs required to execute a particular application program. The reviewer, using the informatin contained in the database, can then make the appropriate database selections to retrieve the necessary input

data and present the data at the designer's terminal. As the data is presented to the designer, several options arise. The designer may choose to move forward (or back) to another data screen, to modify one or more of the entries or to print the information on a nearby printer. This capability also makes the reviewer an effective tool for analyzing changes in output due to execution of the application program with a variety of inputs. The reviewer requires no specific knowledge of the database management software. In the second method, a designer familiar with the database software may call up information via the relational database manager. AIDE has been implemented using the ARIS [3] relational database manager.

2.2 The Application Program

It is assumed that a suite of application programs has been provided for use in analyzing the design problem described by the variables in the configuration database and the input and output variables have already been placed in the database by persons familiar with the programs. Designers need only give the command to execute a given application program; AIDE then takes over. Data is extracted from the database and put in a form appropriate for the application program. This is followed by the execution of the applications program itself. Finally, resulting answers are inserted back into the configuration database. The input or output data "templates" that are used here may also be used by the reviewer program, mentioned in the previous subsection, to extract data for presentation at the designers terminal.

2.3 Working Environment

It is recognized that in spite of the best possible preparation, a designer rarely completes one version of a design at one setting. Additionally, designers may well wish to keep several active versions over some phase of the design. This requires several different copies of the configuration database. For this reason the idea of a "workspace" is used with the AIDE system.

Fundamentally, a workspace is used to keep track of the current standing of a particular design or version of a design. This allows the user to have different workspaces for each variant of a design.

While a designer is executing in the AIDE setting, the workspace automatically tracks not only the current version of the configuration database but also the current application program, data template, and procedure file. In addition, AIDE will automatically record all instructions given during the development of the version associated with this workspace. This "command log" plays several important roles. First it serves as an audit, so there is never any question in which order commands were issued or how a particular result was derived. Secondly, it has been previously mentioned that much of the design process is

iterative in nature. The command log can be edited to create "procedures" from any designated portion of the file. Once this is accomplished, the user need only to invoke the single procedure for execution, after using the reviewer to define the appropriate input data, rather than commanding step by step program execution.

Section 3
SAMPLE SESSION FOR A BASIC USER

It is expected that most users of the AIDE system will interact with it via the menu driven mode in one form or another. The purpose of this section is to describe, via a sample session, how a basic user might interact with AIDE for a given application. It should be noted that this example has been set up to illustrate some capabilities of the AIDE system but is not exhaustive. In addition, the menu design here has not been patterned to a specific model. It is possible for an AIDE system administrator to quickly develop a selection of menus to any desired format. A description of that process will follow in Section 7.

For this example, the user JLS1 has been provided with menus to help him execute programs of a space station sizing and development system known as SSCDM (Space Station Conceptual Design Model). User input is indicated in this example by underlined values followed by a carriage return. Copies of the screens themselves are given in Appendix A. Such screens are referenced by the notation Screen:An where n is the screen number.

The session commences with the user entering the following command sequence.

```
AIDE
ENTER USR-ID:  JLS1
```

The user will execute these commands from a personal directory. Copies of configuration databases will be brought into the user's area. The "golden" configuration database, the application programs, AIDE utilities, etc are all brought in as needed by the executive, without requiring the user to know their location. (For description of the directory structure, see the EASIE Installation Manual). At this point the system begins to define the user's working environment by soliciting the user's choice of configuration database, application program, and view of the database. (Screen A1). This is the first screen presented to the user. The user has the choice of making a copy of a master configuration database (just BLKDB here) or using or copying one of the previously defined user databases (SSCDM1, NEWDB or NUDB here). The master configurations are the same as the "golden" databases discussed previously. User's have "read only" access to that data; that is, it may be copied but no changes may be stored there. In this instance the user has decided not to make another copy of the configuration data but to work with the data existing in SSCDM1. After this selection, the user is presented (Screen A2) with all applications programs that are associated with the chosen design system. The user then identifies the program which is to be applied to the current data. In the example, the designer has chosen the electrical power sizing program ELECT.

Once again the user is given a menu (Screen A3) of objects from which to choose. This time a list of templates is provided. A template is merely a predefined set of variables from the database which have been grouped together for display and editing by the "reviewer". In this instance, each template can be seen to be associated with either the input or the output variables of the application programs listed in Screen A2. AIDE will not automatically assume the choice of any of these templates, even though it has already been given the name of a particular application program. This is because the AIDE executive cannot know whether you are interested in the program's input template, output template, or some other template created as a subset of one of these predefined I/O templates. E.IN is a user-generated template corresponding to a subset of the variables used as input for ELECT. After this selection, Screen A4 will automatically appear to the user.

From here on in this sample execution, the flow of the design will be controlled by choices made from this menu. (Menus such as this can be quickly built and modified by an experienced user.) The first choice is to see all the input corresponding to ELECT.

Screens A5 and A6 show just a few of the variables (as presented by the "reviewer") required for input to the ELECT program; any or all of these may be modified by the user.

The AIDE reviewer is arguably the most powerful tool that is provided to the designer. With it the designer can automatically access and modify any portion of the configuration database without resorting to the usually cumbersome database command language.

For this example the user merely chooses the menu selection to review the input data for the chosen program, ELECT. If such a command had to be issued directly to the database it would run something like

```
SELECT * FROM SSCDM1 WHERE PROGDATA EQ ELECT.IN.
```

Similar statements are avoided for each database operation performed.

The reviewer presents data to the user much like the popular spread sheet programs. Basic reviewer commands follow. When it appears in a reviewer command "n" represents an integer.

- Nn - Display NEXT screen of data, "n" is used to advance more than one screen if positive, or backup screens if negative.
- R - REDISPLAY the current screen, this is particularly helpful on Tektronix terminals.
- Ln - LENGTH of screen should be "n" lines, allows user to modify for terminal screen size.

n - Modify the value of the "n"th variable. Once this is specified, the current value is accessed and a user changes the value by typing over those positions needing change.
H - Provides help.
PA - Prints all the variables retrieved for this reviewing session.
E - Exit and save modifications.
Q - Quit, do not save the modifications.

Screens A7 and A8 are presented to help contrast the difference between program templates and user-defined templates. Recall here the user template E.IN had been selected earlier. This template is a subtemplate of the total ELECT program input template; perhaps the user feels a need only to modify these variables. The building of a subtemplate like E.IN requires the full AIDE environment and will be discussed in section 3, but for now the novice user can assume he will be provided with any such necessary subtemplates.

Now, assume the user has made menu choices "7" to execute the program and "6" to review all of the data put back into the database by the program ELECT. Screen A9 shows some of the output variables placed back into the database via the output template ELECT.OUT.

The user has completed the current task and exits the AIDE environment (Screen A10).

Screen A11 is a copy of the log that was kept during the previous session. This log will continue as the design progresses. In addition, it will be seen later that portions of the log can be packaged into a procedure and executed via a single command. Lines preceded by a > symbol are those commands actually issued, and lines preceded by an * are comments placed in the log in response to the commands. Only the > lines would be used in a procedure.

This sample session, as has been stated, is not representative of the full AIDE environment, but is a preplanned set of options (see the menu of screen A4) provided to the new user. In fact, this has been accomplished by automatically putting the user into a "procedure" (screen A12) at login. This means, an experienced user can define a set of options (via the menu) for a new user on the system. Screen A12, though not explained in detail until Section 7, shows the actual command procedure controlling the user's activities during this simple example, depicted in screens A1 to A11.

Section 4
USING COMMANDS IN THE FULL AIDE ENVIRONMENT

AIDE is intended to be as flexible as possible for the user. Commands can be issued via menu selection or directly typed. Various levels of menu, display and help text are available to the designer. Version 1 of the AIDE system has 7 different standard menus in addition to a "permanent" menu of commands that can be executed at any point in a work session. These menus will be presented below with brief explanation, and this will then be followed by an explanation of the AIDE command format which can be used instead of the menus.

The Permanent (see Figure 1) menu provides the user basic informational access. The Utility Selection (see Figure 2) menu provides access to the other six functionally organized menus. The right-hand column on this and all succeeding menus provides the command which accomplishes the same result as its opposite menu choice. For example, the command "ACT UTL WSC" activates the Workspace Control Utility menu just as a menu choice of "1" will. The difference is that "ACT UTL WSC" can be given any time, not just when using the Utility Selection menu.

The Workspace Control menu (see Figure 3) aids the user in defining the workspace environment. The Data Review/Modification menu (see Figure 4) aids the user in selecting a view of the data and executing the "reviewer". The Application Execution menu (see Figure 5) provides the user with control over program execution. The Procedure Execution menu (see Figure 6) provides the user control over procedure execution. A procedure is a user-defined collection of AIDE statements covering some portion of design sequence. The Procedure Building menu (see Figure 7) aids the user in defining a procedure. The Template Building menu (see Figure 8) aids the user in defining a new view into the configuration database.

All AIDE commands take on the following format:

<command verb> [<object type>][<object name>] [other info as required]

Here [] indicates optional information and <> represents the name of a command or object.

The following sections provide a detailed description of the command verb and object types used to run the full AIDE system.

4.1 Command #1: Permanent Menu Commands

All permanent menu commands can be given by a single letter abbreviation.

H - Help
This command will call up AIDE help files relevant to the current situation.

D - Directory
This command will list all configuration databases, data templates and saved workspaces accessible to the user. "Master" files are those existing in the central project directory and are available on a "read only" basis.

I - Inventory
This command provides a list of all application programs and procedures accessible to the user. "Master" again implies "read only" access.

S - System Command
The system is currently implemented on Prime computers. This command indicates that what follows should be interpreted as a PRIMOS operating system command.

ex.: In order to invoke the interactive database manager RIM, one needs to issue the command:
RUN RIM.
Through AIDE this becomes
S RUN RIM.

C - Add a Comment to the Command Log.
This allows the user to make notes in the log for later reference and clarity.

T - Toggle the Amount of Menu Information Displayed.
This allows the user to select how much or how little information should be displayed. There are four possible settings:

1. Full information,
2. Status variables and utility menus only,
3. Status varialbes only
4. Prompt only

Each time the command T is issued the next state in the circular sequence is chosen.

R - Return to the Previous Menu
As users move through the development of a design different menus will be accessed in some sequence. This command allows the user to return to the previous menu.

Q - Quit this Sequence of Menus
This command allows the user to dispose of the current sequence of menus and return immediately to the basic utility selection menu.

L - Logout
This command gives an orderly closeout of the AIDE system and returns the user to the computer's operating system.

<CR> - Carriage Return
This will clear the screen for the user or reprint any menus or status information desired.

0 - Zero will cancel a command when given as a response during a query process for more information.

```
ex.:  ENTER COMMAND:  ACT APPL
      ENTER FILE TO BE ACTIVATED:  0
      ** COMMAND CANCELLED **
```

<BREAK> or <^P> - Send a Break Character
This will cause an application program to halt execution and ask the user instructions concerning how to proceed.

```
PERMANENT MENU:

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
    RETURN to the MAIN MENU
L - LOGOUT

<CR> - Clear the screen and relist the menu
```

Figure 1. Permanent Menu

```
UTILITY SELECTION (MAIN)

                                        COMMAND FORMAT
1 - WORKSPACE CONTROL UTILITY                ACT UTL  <WSC>
2 - DATA MODIFICATION UTILITY                ACT UTL  <DATA>
3 - APPLICATION EXECUTION UTILITY            ACT UTL  <APEX>
4 - PROCEDURE EXECUTION UTILITY              ACT UTL  <PREX>
5 - PROCEDURE BUILDING UTILITY               ACT UTL  <PBLD>
6 - TEMPLATE BUILDING UTILITY                ACT UTL  <TBLD>

ENTER COMMAND:
```

Figure 2. Utility Selection Menu

```
WORKSPACE CONTROL

                                        COMMAND FORMAT
 1 - READ DESCRIPTION   -  WORKSPACE        RD WS      <name>
 2 -                    -  CONFIGURATION    RD CFG     <name>
 3 -                    -  TEMPLATE         RD TPL     <name>
 4 -                    -  APPL. PROG.      RD APPL    <name>
 5 -                    -  PROCEDURE        RD PROC    <name>
 6 - CLEAR LOG OF OLD INFORMATION           CL
 7 - TYPE               -  COMMAND LOG      TY LOG     <name>
 8 -                    -  PROCEDURE        TY PROC    <name>
 9 - NEW                -  WORKSPACE        N WS
10 -                    -  CONFIGURATION    N CFG      <name>
11 - COPY               -  WORKSPACE        CP WS      <f,to>
12 -                    -  PROCEDURE        CP PROC    <f,to>
13 - ACTIVATE           -  WORKSPACE        ACT WS     <name>
14 -                    -  CONFIGURATION    ACT CFG    <name>
15 -                    -  TEMPLATE         ACT TPL    <name>
16 -                    -  APPL. PROG.      ACT APPL   <name>
17 -                    -  UTILITY          ACT UTL    <menu>
18 -                    -  INPUT TEMPL      ACT ITPL
19 -                    -  OUTPUT TEMPL     ACT OTPL
20 -                    -  PROCEDURE        ACT PROC   <name>
21 -                    -  PROGRAM UFD      ACT PUFD   <path>
22 - SAVE TEMPORARY     -  WORKSPACE        SA WS      <name>
23 -                    -  PROCEDURE        SA PROC    <name>
24 - REMOVE FROM UFD    -  WORKSPACE        RM WS      <name>
25 -                    -  CONFIGURATION    RM CFG     <name>
26 -                    -  TEMPLATE         RM TPL     <name>
27 -                    -  PROCEDURE        RM PROC    <name>
28 - SET USER LOGIN CHARACTERISTICS         SLOG

ENTER COMMAND:
```

Figure 3. Workspace Control Menu

```
DATA REVIEW/MODIFICATION


                                                COMMAND FORMAT
 1 - READ DESCRIPTION   - TEMPLATE              RD TPL    <name>
 2 -                    - CONFIGURATION         RD CFG    <name>
 3 - ACTIVATE           - CONFIGURATION         ACT CFG   <name>
 4 -          VIEW AS   - INPUT TMPL.           ACT ITPL
 5 -          VIEW AS   - OUTPUT TEMPL.         ACT OTPL
 6 -          VIEW AS   - ALT. TMPL.            ACT TPL   <name>
 7 - COPY               - CONFIGURATION         CP CFG    <f,to>
 8 -                    - TEMPLATE              CP TPL    <f,to>
 9 - ACTIVIATE TEMPLATE BUILDING UTILITY        ACT UTL   <TBLD>
10 - REVIEW             - INPUT DB              RVU IDB
11 -                    - OUTPUT DB             RVU ODB
12 - REVIEW DB WITH ACTIVE TEMPLATE             RVU
13 - PRINT              - INPUT DB              PRVU IDB
14 -                    - OUTPUT DB             PRVU ODB
15 - PRINT DB WITH ACTIVE TEMPLATE              PRVU
16 - CHANGE CONFIGURATION DESCRIPTION           CD CFG
17 - REMOVE ALL DATA    - INPUT & OUTPUT        RM CFG
18 - TYPE CURRENT COMMANG LOG                   TY LOG    <-    >

ENTER COMMAND:
```

Figure 4. Data/Review Modification Menu

APPLICATION EXECUTION

			COMMAND FORMAT	
1	- READ DESCRIPTION	- APPL. PROG.	RD APPL	<name>
2	-	- CONFIGURATION	RD CFG	<name>
3	-	- TEMPLATE	RD TPL	<name>
4	- NEW	- CONFIGURATION	N CFG	<base>
5	- ACTIVATE	- APPL. PROG.	ACT APPL	<name>
6	-	- CONFIGURATION	ACT CFG	<name>
7	- ACTIVATE DB VIEW AS	- INPUT TEMPL.	ACT ITPL	
8	-	- OUTPUT TEMPL.	ACT OTPL	
9	-	- ALT. TEMPL.	ACT TPL	<name>
10	- COPY	- CONFIGURATION	CP CFG	<f,to>
11	- REVIEW DB WITH ACTIVE TEMPLATE		RVU	
12	- REVIEW	- INPUT DB	RVU IDB	
13	-	- OUTPUT DB	RVU ODB	
14	- PRINT DB WITH ACTIVE TEMPLATE		PRVU	
15	- PRINT	- INPUT DB	PRVU IDB	
16	-	- OUTPUT DB	PRVU ODB	
17	- EXECUTE CURRENT APPLICATION PROGRAM		EX APPL	<- >
18	- CHANGE DESCRIPTION	- CONFIGURATION	CD CFG	<name>
19	- REMOVE ALL DATA	- INPUT & OUTPUT	RM CFG	
20	- TYPE CURRENT COMMAND LOG		TY LOG	<- >

ENTER COMMAND:

Figure 5. Application Execution Menu

PROCEDURE EXECUTION

```
                                          COMMAND FORMAT
 1 - READ DESCRIPTION     - PROCEDURE       RD PROC    <name>
 2 -                      - CONFIGURATION   RD CFG     <name>
 3 -                      - APPL. PROG.     RD APPL    <name>
 4 -                      - TEMPLATE        RD TPL     <name>
 5 - TYPE A GIVEN COMMAND LOG               TY LOG     <name>
 6 - ACTIVATE             - PROCEDURE       ACT PROC   <name>
 7 -                      - CONFIGURATION   ACT CFG    <name>
 8 - ACTIVATE DB VIEW AS  - INPUT TEMPL.    ACT ITPL
 9 -                      - OUTPUT TEMPL.   ACT OTPL
10 -                      - ALT. TEMPL.     ACT TPL    <name>
11 - COPY                 - PROCEDURE       CP PROC    <f,to>
12 -                      - CONFIGURATION   CP CFG     <f,to>
13 - NEW                  - CONFIGURATION   N CFG      <base>
14 - PROCEDURE BUILDING UTILITY             ACT UTL    <PBLD>
15 - REVIEW THE DB WITH THE ACTIVE TMPL.    RVU
16 - REVIEW                                 RVU IDB
17 -                       - OUTPUT DB      RVU ODB
18 - PRINT THE DB WITH THE ACTIVE TEMPLAT   PRVU
19 - PRINT                 - INPUT DB       PRVU IDB
20 -                       - OUTPUT DB      PRVU ODB
21 - EXECUTE THE CURRENT COMMAND FILE       EX PROC    <-   >
22 - CHANGE DESCRIPTION  - PROCEDURE       CD PROC    <name>
23 -                     - CONFIGURATION   CD CFG     <name>
24 - REMOVE ALL DATA     - INPUT & OUTPUT  RM CFG
25 - TYPE THE CURRENT COMMAND LOG           TY LOG     <-   >

ENTER COMMAND:
```

Figure 6. Procedure Execution Menu

```
PROCEDURE BUILDING

                                          COMMAND FORMAT
 1 - READ DESCRIPTION      - PROCEDURE     RD PROC    <name>
 2 - TYPE                  - COMMAND LOG   TY LOG     <name>
 3 -                       - PROCEDURE     TY PROC    <name>
 4 - CLEAR LOG OF OLD INFORMATION          CL
 5 - ACTIVATE              - PROCEDURE     ACT PROC   <name>
 6 - COPY                  - PROCEDURE     CP PROC    <f,to>
 7 - DEFINE A NEW PROCEDURE                N PROC
 8 - EDIT AN EXISTING PROCEDURE            ED PROC    <name>
 9 - EDIT A LOG TO BUILD PROCEDURE         ED LOG     <name>
10 - SAVE A TEMPORARY PROCEDURE            SA PROC    <new>
11 - CHANGE THE PROCEDURE DESCRIPTION      CD PROC
12 - REMOVE THE PROCEDURE                  RM PROC
13 - TYPE THE CURRENT COMMAND LOG          TY LOG     <-   >

ENTER COMMAND:
```

Figure 7. Procedure Building Menu

TEMPLATE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION OF	TEMPLATE	RD TPL	<name>
2 - ACTIVATE	- INPUT TEMPLATE	ACT ITPL	
3 -	- OUTPUT TEMPLATE	ACT OTPL	
4 -	- ALT TEMPLATE	ACT TPL	<name>
5 - COPY	- TEMPLATE	CP TPL	<f,to>
6 - DEFINE A NEW TEMPLATE		N TPL	
7 - EDIT THE ACTIVE TEMPLATE		ED TPL	<- >
8 - CHANGE THE TEMPLATE DESCRIPTION		CD TPL	
9 - REMOVE TEMPLATE FROM UFD		RM TP	<name>
10 - TYPE THE CURRENT COMMAND LOG		TY LOG	<- >

ENTER COMMAND:

Figure 8. Template Building Menu

4.2 Core Object Types

The following form a collection of object types necessary for execution of AIDE commands. Illustrations of these types will be cited from the sample sessions contained in Appendices A and B.

APPL - Application Program
Any application program which can be executed from AIDE. Screens A2 and B11 illustrate an inventory listing which includes the application programs associated with the SSCDM project team.

CFG - Configuration Database
A configuration database is a collection of all constants, parameters and variables pertinent to a given model. Screen A1 illustrates a directory listing including a master or "golden" configuration database and three user configuration databases.

LOG - Command Log
The collection of all commands executed to the current moment. Screens A11, B19, B22 and B29 illustrate various stages in the development of a command log.

PROC - Procedure Command File
A procedure command file is a collection of AIDE system commands, Screen A12, that can be set up for processing by the AIDE command interpreter. Screen B20 illustrates one method of deriving a procedure file from the command log. The directory listing, as illustrated on Screen B11, will list all available procedure files.

UTL - Utility Menu
A utility menu is a group of AIDE commands typically used to accomplish a specific task, combined together in a menu format for selection by the user. Screen B1-B7 illustrate the seven "standard" menus.

TPL - Data Template
A list of variables (and their location in the configuration database) created for use by the reviewer program. A directory listing as on B10 illustrates the template available to the user. Note, for the example of A3 or B10, all templates listed are either input or output templates of a SSCDM application program.

ITPL - Input Data Template
Represents the input data template of an application program.

OTPL - Output Data Template
Represents the output data template of an application program.

WS - Workspace
Represents a collection of information relevant to the current status of some design configuration. The directory command (illustrated on Screen B10) provides a sample listing of the currently available workspaces. Screens B26 and B27 illstrate the activation of a previously saved workspace for further analysis.

IDB - Input Database
Represents that portion of the configuration database associated with the input of some application program, that is, that portion defined by the input template ITPL.

ODB - Output Database
Represents that portion of the configuration database associated with the output of some application program, that is, that portion defined by the output template OTPL.

Note: At times a user will be creating a fresh copy of an object; such an object will be assigned the temporary name T$. For example at login the user is working with a workspace T$. Any such temporary objects may be saved as permanent at anytime during an AIDE session. If a user attempts to logout of an AIDE session with temporary objects, AIDE will solicit whether the user wants to save or dispose of those objects.

4.3 Workspace Variables: The Usual Command Objects

One of the purposes of providing a workspace to the designer is to track a set of state variables. Once defined, these variables provided act as default object names to which commands are applied, unless otherwise specified. As previously mentioned, design activities in the AIDE environment are carried out through operations on a collection of files. The names of these files are the basic workspace variables. A full list is provided in Section 7. In the case of a basic user, it has been demonstrated how management of the workspace can be carried out automatically via menus provided to the designer. For example, in Screen A1 the response SSCDM1 causes the workspace variable defining the configuration database to be set to SSCDM1. After that, any program requiring data will use the SSCDM1 database.

On the other hand, designers who access the full AIDE environment are presented with a more complete and flexible control of the active workspace variables through the commands described in Section 4.4 and 4.5.

Below, the user will find a brief description of the types of variables associated with the workspace, how they are used and what values are allowable.

Configuration Database

Any file of type CFG in the user's file directory. "Master" or "golden" configuations cannot be associated with the workspace. Once specified, all programs will refer to this database for data. A list of available databases may be obtained with the D command. This workspace variable is accessed via the commands ACTIVATE, NEW and EDITVAR.

Application Program

Any file of the type APPL. Once defined, any reference to execution will default to this program. Input and output databases and templates will refer specifically to those variables required for input or output for this program. A list of available application programs can be obtained with the I command. This workspace variable is accessed via the commands ACTIVATE and EDITVAR.

Data Template

A file of type TPL that contains a list of some of the variables associated with this design project. This file may contain a user selected subset of particular interest in the design. Once associated with the workspace it becomes the default selection mechanism for variables in the database to be displayed to the designer. A list of available templates is obtained via the D command. This workspace variable is accessed via the ACTIVATE and EDITVAR commands.

Procedure Command File

A file of type PROC which contains a sequence of AIDE commands. This file allows designers to prepackage sequences of design steps and execute them as a group without further user intervention. Once defined, any reference to procedure execution will default to this file. A list of procedures can be obtained via the I command. This workspace variable is accessed via the ACTIVATE, NEW and EDITVAR commands.

Active Workspace

A file of type WS which defines all associated objects and states of execution. It is possible to switch, save, attach or delete any given workspace in the user file directory. Saving a workspace records the status of that particular project. Attaching a workspace restores the previously saved status. A list of saved workspaces is available through the D command. This workspace variable is accessed through the ACTIVATE, NEW and EDITVAR commands. The workspace T$ is a temporary workspace.

Program File Directory

This is the directory where the project application programs, configuration databases, and input/output templates are found. Access to this workspace variable is via the ACTIVATE and EDITVAR commands.

Command Autodefault

When a command is only partially specified by a user, the value of this variable determines how it will be completed. If the value is true the command is completed by using the value of the current reference in the workspace. For example EX APPL would execute the current application program. If the value is false the system will prompt the user for the rest of the command. For example, EX APPL would cause the response ENTER NAME OF PROGRAM:. This workspace variable can be accessed via the ACTIVATE and EDITVAR commands.

User Knowledge Level

This variable indicates to the system how much help should be provided to the user. Allowable values are integers from 1 to 3, where 1 represents a novice user and 3 represents an expert user. Access to this workspace variable is through EDITVAR.

Output Print Level

This variable indicates to the system how much menu information should be routinely provided. The allowable values are integers from 1 to 4, where 1 prints status and all menus, 2 prints status and utility menu, 3 prints status, and 4 prints a prompt only. Access to this workspace variable is through the T and EDITVAR commands.

4.4 Command #2: Aide Core Commands

This section contains a list of commands with which the designer, who wishes to uses the full AIDE environment, should become familiar.

ACTIVATE, ACT

Form: ACT <type> [<filename>]
Command used to activate any type of object for the user's workspace. This command will define the object indicated as the current reference object of that type.

EX: ACT CFG SSCDM1

Discussion: This command will cause the configuration database, SSCDM1 to become the current reference database. Not only is this important for later analysis, but it allows the use of defaults. That is, subsequent to the above activation, any command which refers to a configuration database will refer specifically to SSCDM1.

Allowable Object Types
APPL, CFG, ITPL, OTPL, PROC, TPL, WS, UTL, BUFD, PUFD, DFLT (An explanation of BUFD, PUFD and DFLT is available in the section for advanced object types.)

Notes:

1. APPL, CFG, PROC,TPL
   If the indicated file does not exist or is not given, the user will be prompted to supply an existing file of the proper type.

2. ITPL, OTPL
   No filename is required and will be ignored if supplied. This command defines the value of the current reference template to be the same as the input or output template of the current reference application program. If no program is currently referenced, that information will be returned to the user.

3. UTL
   The allowable filenames for this object type are MAIN, WSC, DATA, APEX, PREX, PBLD, TBLD (the standard utility menus) and any special utility menus that may have been built for the design. Page B1 illustrates such a menu, namely LASS. The result of this command is to present the user with the indicated utility menu.

4. WS
   This command will restore the characteristics and configuration of a previously defined workspace.
   If no workspace name is referenced, the user will be asked to provide one.

CAUTION: Before using this command, save the currently active workspace, if it has not been saved previously (i.e., if it is listed as temporary).

COPY, CP

Form: CP <type>[<fromfilename><tofilename>]

Command used to make a copy of an object.

EX: CP CFG SSCDM1 SSCDM2

Discussion: This command will make a copy of the configuration database SSCDM1, naming the result SSCDM2.

Allowable Object Types:
APPL, CFG, PROC, TPL, WS, FILE

Notes:

1. If the filenames are not supplied, AIDE will solicit them from the user. If the from-filename does not exist or if the to-filename does exist, the user will be queried further concerning proper naming of these files.

2. FILE
This object type allows the user to copy non-AIDE object files without going back to the operating system. Note: files must be named with paths and extensions here, which is not required for AIDE files.

EDIT, ED

Form: ED <type> [<filename>]

Command used to invoke the system editor for certain operations.

EX: ED PROC NEWPROC

This command allows the user to modify the procedure NEWPROC via the system editor.

Allowable Object Types:
LOG, PROC, TPL

Notes:

1. LOG
   Makes a copy of a log file so that the user can turn it into a procedure. All system comments and prompts are deleted before the copy is made. If no log file is specified, the log attached to the current workspace is used.

2. PROC
   Invokes the system editor to modify the indicated procedure file. If no file is given the reference procedure is used, if defined.

3. TPL
   Invokes an AIDE editor designed for defining a user subtemplate.
   CAUTION: The template edited will be altered. Thus it should generally begin as a user copy of some other desired template (see the copy command). Template editor commands follow. See Screen B30 for an example.

   Sn - Save the variable with reference number "n" for the subtemplate.
   Fn - Free previously marked variable from the subtemplate
   Nn - move "n" pages throught the template, note n maybe "+" or "-"
   T - move to the top page of variables.
   B - move to the bottom page of variables.
   R - reprint the current page.
   E - end edit and save results
   Q - quit edit, do not save results, restore original template

EXECUTE, EX

Form: EX <type> [<filename>]

Command to execute a given application program or procedure command file.

EX: EX APPL ATMOS

Discussion: This command will execute the application program ATMOS.

Allowable Object Types:
APPL, PROC

Notes:
If no filename is given, the current application program or reference procedure, if defined, will be executed.

PRINT, PR

Form: PR <type> [<filename>]

Command used to print the indicated file at a local hard copy printer.

EX: PR LOG T$

Discussion: This command will cause the log associated with the temporary workspace, T$, to be printed.

Allowable Object Types:
LOG, PROC, FILE

Notes:

1. LOG, PROC - If no filename is given then the currently active log or reference procedure command file, if defined, will be printed.

2. FILE - This allows the user to print non-AIDE object files without going back to the operating system.

PRINTVU, PRVU

Form: PRVU [<type>]

Command to print a "view" of a configuration database. A view of a database is defined as the collection of variables defined by a data template.

EX: PRVU IDB

Discussion: This command will obtain the values of the input variables for the current application program from the current configuration database and print them along with their database description.

Allowable Object Types:
IDB, ODB

Notes:

1. If no object type is listed the system will print variables indicated by the current template and the current configuration database.

2. If there is no active configuration database the user will be so informed and the command will terminate.

RD, (Read Description)

Form: RD <type> [<filename>]

Command to read the description file associated with any workspace, program procedure, template, or database.

EX: RD APPL ATMOS

Discussion: This command will send the description file of the application program ATMOS to the user's terminal.

Allowable Object Types:
APPL, CFG, ITPL, OTPL, PROC, TPL, WS

Notes:

1. APPL, CFG, PROC, TPL, WS - If no filename is given then the description for the current value in the workspace, if defined, will be presented.

2. ITPL, OTPL - No filename is necessary; the current input or output template description will be presented.

REMOVE, RM

Form: RM <type> [<filename>]

Command to remove a file from the user's file directory.

EX: RM CFG SSCDM1

Discussion: This command will remove the configuration database SSCDM1 from the user's file directory.

Allowable Object Types:
APPL, CFGT, PROC, TPL, WS, FILE

Notes:

1. The user will need to verify the remove command to help prevent accidental deletion. The user may only remove files from a personal file directory.

2. APPL, CFG, PROC, TPL, WS. The user will be warned if the object to be removed is one currently active in the workspace.

3. FILE - This type will allow the user to remove any personal file without leaving the AIDE system.

REVIEW, RVU

Form: RVU [<type>]

Command to invoke the interactive "Reviewer" program. This command will display for possible modification a "view" of a configuration database. A veiw of a database is defined as the collection of variables defined by a data template.

EX: RVU IDB

Discussion: This command will obtain values of the input variables for the current application program from the current data base. Interactive commands may then be issued to modify this data.

Allowable Object Types:
IDB, ODB

Notes:

1. If no object type is given the reviewer will be executed using the current reference data template and the current configuration database.

2. If there is no active configuration database, the user will be so informed and the command will terminate.

SAVE, SA

Form: SA <type> [<filename>]

Command to save a current temporary object for later work.

EX: SA PROC NEWPROC

Discussion: This command will save the current temporary procedure as NEWPROC.

Allowable Object Types:
WS, PROC

Notes:

1. In general it is assumed that the current reference procedure or work space is temporary, that is, has the name T$.

2. WS - Save may also be used to copy the current status of the workspace into a new workspace.

TYPE, TY

Form: TY <type> [<filename>]

Command used to display a file at the user's terminal. Please note that lines are displayed in groups of 50 for the user's convenience.

EX: TY PROC NEWPROC

Discussion: This command will display the procedure command file NEWPROC for the user.

Allowable Object Types:
LOG, PROC, FILE, BAT

Notes:

1. LOG, PROC - If no filename is specified then the currently active log or procedure command file, if defined, will be displayed.

2. FILE - This allows the user to display non-AIDE object files without going back to the operating system.

3. BAT - This allows the typing of a BATCH JOB event file

   e.g., TY BAT BAT3.

4.5 Commands #3: Advanced AIDE Commands

This section contains the reviewing AIDE commands which provide the designer with full control of the design environment.

CD, (Change Description)

Form: CD <type> [<filename>]

Command which allows the user to expand, modify, or update description files using the system editor.

EX: CD WS EXAMPLEWS

Discussion: This command would allow the user to expand or update the description file for the user's workspace EXAMPLE.

Allowable Object Types:
APPL, CFG, ITPL, OTPL, PROC, TPL, WS

Notes:

1. APPL, CFG, PROC, TPL, WS - If no filename is given then the description for the current value in the workspace, if defined, will be available for change.

2. ITPL, OTPL - No filename is necessary. The description of the current input or output template will be made available for change.

CLEARLOG, CL

Form: CL [D T]

Command which allows the user to remove old information cluttering the log file of the attached workspace.

EX: CL T

Discussion: This command would clear the log completely

Allowable Types:
D - clear information prior to a given date
T - clear the log totally

Notes:

1. D - A menu of all logged dates and times are presented to the user, all log information prior to the date selected will be deleted.

2. T - All information in the log will be deleted.

CNAME, CN, (Change Name)

Form: CN <type> [<oldname> <new name>]

Command which allows the user to change the name of any of the user files, workspaces, configuration, templates or procedures.

EX: CN TPL OLDTPL NUTPL

Discussion: This command will change the name of the user template OLDTPL to NUTPL.

Allowable Object Types:
APPL, CFG, PROC, TPL, WS, FILE

Notes:

1. APPL, CFG, PROC, TPL, WS - If the user does not supply the old and new file names they will be solicted.

2. FILE - Allows the user to change the name of any file in the home directory without returning to the operating system.

GET

Form: GET <type> [<options>][<messages>]

Command used by programmers to build a procedure. It is used to interactively request input data during the execution of that procedure. See Section 6 for an in-depth discussion of the GET command.

EX: GET CFG

Discussion: This command presents a listing of both master and user database configurations and requests that the user choose one. The chosen configuration is then activated in the current workspace.

Allowable Object Types:
APPL, CFG, JMPC, JMPL, LACT, MENU, MODE, PROC, QRYC, QRYL, TPL, WS

Notes:

1. APPL, CFG, PROC, TPL, WS - An appropriate list of master and user objects are presented to the user. The user's choice will be automatically activated in the current workspace. If a "master" CFG is chosen it will, in addition, be copied into the user's file directory.

   e.g. GET PROC

   ** The object types that follow are being discussed here for the first time. That is, they were not included in Section 5.3.2.

2. JMPC, JMPL - Jump statements that cause an unconditional change in the value of the procedure counter. JMPC causes a counted, relative change in the procedure counter. The jump may be either positive or negative.

   e.g. GET JMPC 7 (pc = pc +7)
   GET JMPC -2 (pc = pc -2)

   JMPL causes the procedure counter to be set to the statement following the indicated label. Aside label statements must have the form C LABEL: <label_id>

   e.g. GET JMPL A5 (pc is set to statement After C LABEL:A5)

3. QRYC, QRYL - Conditional jump statements that can cause the procedure counter to change based upon the user's answer to a query. QRYC causes a counted, relative change in the procedure counter. The jump may be either positive or negative. The jump occurs only if the response to the query is "Y".

    e.g. GET QRYC -11 RERUN THE APPLICATION (Y/N)?

    (If the user answers "Y", pc = pc-11

    QRYL causes a jump to the indicated label when the answer to the query is "Y".

    e.g. GET QRYL LAST EXIT THE PROCEDURE (Y/N)?

    (If user responds "Y", the pc would be set to the statement after C LABEL:LAST)

4. LACT - Causes the current date and time to be entered into the log. It is generally recommended that each procedure begin with a GET LACT so that the activation date will be placed in the log.

5. MENU - Causes the indicated menu to be presented to the user. Once the user makes a choice, the pc will be reset to execute the portion of the procedure corresponding to that choice. An in depth discussion of this statement appears in the following section.

    e.g. GET MENU 2

6. MODE - This statement is used to insure that AIDE recognizes the appropriate input and output modes. An in depth discussion appears in the following section.

    e.g. GET MODE

NEW, N

Form: N <type> [<name>]

Command used to get a fresh object

EX: N WS

Discussion: This command will cause the replacement of the current workspace with a new, temporary workspace.

Allowable Object Types:
CFG, PROC, TPL, UTL, WS

Notes:

1. CFG, TPL - Since new configuration databases and templates must use some existing configuration or template as a base, the name of the base configuration or template will be requested if it has not been provided. The result will be attached to the current work space.

2. PROC - A new (empty) procedure file will be attached to the current workspace.

3. UTL - A new utility menu will be attached to the current workspace. This is equivalent to the ACT UTL.

4. WS - A new (empty) temporary workspace will be provided after the status of the current workspace is determine that is saved or deleted.

SETLOGIN, SLOG

Form: SLOG

Command which allows a user to modify the login environment. For example a designer may always wish to begin a session with a particular workspace and configuration database active. This command allows those objects to be specified. Upon issuing the SLOG command, the user is placed in an interactive mode for making the appropriate changes.

EX: SLOG causes the following menu to appear.

```
NUMBER      LOGIN VARIABLE          VALUE
-------------------------------------------------
  1         USER KNOWLEDGE LEVEL      2
  2          OUTPUT PRINT LEVEL       2
  3          COMMAND AUTODEFAULT      T
  4      CONFIGURATION DATABASE
  5          USER DATA TEMPLATE
  6      PROCEDURE COMMAND FILE
  7          APPLICATION PROGRAM
  8          ATTACHED WORKSPACE     T$
  9      PROGRAM FILE DIRECTORY     VAB2>CSC
 10          MENU PRESENTED         UTILITY SELECTION
 11      BEGIN EXECUTING PROCEDURE          F
 12        PROCEDURE TO EXECUTE
-------------------------------------------------
n - modify "n"th variable  S - save changes
R - reprint choices        Q - quit, delete changes
>
```

STATUS, STAT

Form: STAT

Command which returns the status of all jobs which have been submitted to the batch processor. Logs for jobs that have executed appear in the "BATCH JOB event files". These files can be read with the TYPE command. Active BATCH JOB are listed in the Active BATCH Queue.

EX: STAT will cause the following response.

```
BATCH JOB event files

BAT3
BAT1
BAT2

 Active BATCH Queue

[JOB rev 19.2.9]

Job status listing for user 787645E:

Jobid$    State        External name      Queue
------  ---------   ------------------   ------
$00133  executing   BAT2.CPL             BQUEUE
```

SUBMIT, SUB

Form: SUB <type> [<filename>]

Command used to send a given application program to the BATCH processor for execution in the background.

EX: SUB APPL ATMOS

Discussion: This command will submit the application program ATMOS to the BATCH processor for execution.

Allowable Object Types:
APPL

Notes:

1. There must be a configuration database active before this command is issued so that the BATCH processor will know where to find necessary data.

2. The user will be notified when processing is completed for a given submission.

3. All submitted jobs take on a name like BAT<n> for some integer n.

EDITVAR, VAR

Form: VAR

Command used to interactively edit changes in the active variables of a workspace.

EX: VAR causes the following menu to appear.

```
NUMBER      LOGIN VARIABLE                  VALUE
-------------------------------------------------
   1      USER KNOWLEDGE LEVEL                2
   2        OUTPUT PRINT LEVEL                4
   3        COMMAND AUTODEFAULT               T
   4     CONFIGURATION DATABASE      SSCDM1
   5       USER DATA TEMPLATE        E.IN
   6     PROCEDURE COMMAND FILE
   7       APPLICATION PROGRAM       WASTE
   8       ATTACHED WORKSPACE        T$
   9     PROGRAM FILE DIRECTORY      VAB2>CSC
-------------------------------------------------
n - modify "n"th variable             "0" - to cancel
R - reprint the choices
>
```

Section 5
A SAMPLE SESSION FOR THE FULL AIDE ENVIRONMENT

The following AIDE session is included as a sample for the beginning user to follow. The screens given in Appendix B were recorded during the session. In the listing below, each command line must be followed with a carriage return. The point at which an Appendix B screen is taken will be denoted by Screen Bn where n represents the screen number. User inputs will be underlined and comments will be enclosed in parenthesis.

(To initialize an AIDE session after logging onto the computer.)

AIDE
ENTER USER-ID: JLS (enter your preferred design system ID)

Screen B1

(The STATUS represents the current values of the major variables associated with this workspace. Notice this workspace is temporary, T$, at initialization.)

1 (This user chooses to activate the workspace control utility. Equivalently ACT UTL WSC)

Screen B2 (The menu now corresponds to the new utility, as does the status variable UTIL-IN-USE.
Note: Screens B3, B4, B5, B6, B7 are included to illustrate the format of the other basic utility menus.)

T (This toggles the menu printing mode to STATUS and UTILITY MENU only.)

Screen B8

T (Once again the printing mode is toggled now only STATUS appears.)

Screen B9

T (To get back to the original printing mode, Screen B2.)

D (Ask for a Directory)

Screen B10 (Recall that MASTER implies that the files exist in the design team's project directory, while USER implies the files are in the user's personal directory.)

I (Asks for an inventory - like a directory for programs and procedures.)

Screen B11

14 (Choose to activate a configuration database)

ENTER CFG TO BE ACTIVATED: SSCDM1

Screen B12 (alternately; ACT CFG SSCDM1)

16 (Choose to activate an application program)

ENTER APPLICATION PROGRAM NAME: ATMOS

Screen B13 (Alternately: ACT APPL ATMOS)
R (Return to the selection utility)
3 (Got the Application Execution Utility, alternately ACT UTL APEX)
RVU IDB (This command takes advantage of the two previous. RVU IDB reviews the input data variables for the program ATMOS, whose values are continued in the configuration database SSCDM1. Note the values of the STATUS variables. From the menu this command could be issued as 12.)
Screen B15 (This is a typical screen from the reviewer; values may be modified interactively of this print.)

17 (Selects execution. Again the current STATUS variables supply missing arguments, namely, execute ATMOS using values from SSCDM1.
Screen B16 Alternately EX APPL)

RVU ODB (Similar to RVUIDB, but now review the output data)

Screens B17,B18
(Sample output data)

TY LOG (Display what has happened in this temporary workspace, T$, since login)

Screen B19 (Notice numeric menu commands have been properly interpreted and all commands have been properly extended to include values of STATUS variables. Lines in the LOG beginning with a "*" are AIDE system comments.)

R (Return to the Utility Selection Menu)
ACT UTL PBLD (Choose the Procedure Building Menu)

9 (Edit the current log in order to create a procedure. This initializes the system editor.

Screen B20 Note all system comments and prompts are removed from the file prior to editing. Editing commands are then issued to reduce the file to the commands:

```
                    RVU IDB
                    EX APPL ATMOS
                    RVU ODB

               The final command, FIL, saves the file as a
               temporary procedure, T$, and exits the editor.)

TY LOG         (Get ready to display all commands since login
               Again note the STATUS values of variables)
Screens B21,B22

L              (Logout but save the work that has been done for
               later)

TEMPORARY PROCEDURE EXISTS AT LOGOUT
DO YOU WISH TO SAVE(Y/N):Y
INPUT PROCEDURE NAME:  NEWPROC
SAVE CURRENT WORKSPACE STATUS(Y/N):Y
WORKSPACE HAS ONLY TEMP NAME
INPUT WORKSPACE NAME:  EXAMPLEWS
Screen B23

AIDE           (A later AIDE Session)
ENTER USER-ID:JLS
1              (Get Workspace Control Utility
D              (Get a directory)
Screen B24, B25
               (Note the new workspace saved from the previous
               session EXAMPLEWS.)

13             (Activate an old workspace, namely EXAMPLEWS,
               alternately ACT WS EXAMPLEWS)
ENTER FILE TO BE ACTIVATED:  EXAMPLEWS
Screen B26, B27

               (Notice that the STATUS variables are restored
               exactly as they existed prior to the last logout.)

EX PROC        (This command will execute the procedure previously
               saved as NEWPROC.  That is, it runs the following
Screen B28     commands in sequence.

                    RVU IDB
                    EX APPL ATMOS
                    RVU ODB)

TY LOG         (After execution, look at the log to see what has
               happened.  Notice the new activation of the work-
Screen B29     space, the execution of the procedure and all
               corresponding commands, and the exiting from the
               procedure.)
```

(This final portion of the example illustrates how the user edits a template to produce a subtemplate containing only those variables in which the user is interested.)

Q (Return to the main utility selection menu.)
6 (Select Template Building)

5 (Make a copy of the templat to edit)
ENTER SOURCE TEMPLATE: OUTP.IN
ENTER DESINATION TEMPLATE: NUOUT
Screen B30

4 (Activate "NUOUT" is the template to edit.)
ENTER TEMPLATE TO ACTIVATE: NUOUT

7 (Edit the activated template)

Screen B31

S 12 ... 25 (Save certain variables for the subtemplate)

Screen B32

R (Reprint the page with saved variables marked)

B (Advance to the bottom page)

S 201 213 210 (Save some more variables)

Screen B33

R (Reprint page)

E (Exit the edit session and save the results)

RVU (Review the database with the activated template; that is, the newly edited template.)

Screens B34, B35

(The reviewer has retrieved only that data indicated by the edited template.)

This concludes the sample session.

Section 6
Aide Procedures

Given the ease with which an introductory user can learn to manipulate data and execute programs in the automatic execution mode, it is envisioned that this mode will become the predominant choice of system users. As previously described, control of the system during the automatic execution mode is governed by a procedure. Thus it becomes important to consider construction of such procedures in order to fit the needs of designers who will not use the full AIDE environment.

6.1 Basic Procedures

The example below illustrates that it is possible to build and execute procedures which do not specify the names of configuration, application programs and so on. Let the following procedure be called EX1.

```
RVU IDB
EX APPL
RVU ODB
```

Naturally for this procedure to make any sense, several conditions must be satisfied. First, the status of the workspace must be set to the AUTODEFAULT mode so that partial commands like these can be completed with defaults from the workspace variables. Secondly, those workspace variables must already be defined.

To complete this discussion, let us assume that the AUTODEFAULT mode has been selected and that workspace variables assume the indicated values.

```
Configuration Database:  CASEA3
Application Program:  AERO
Input Template:  AERO.IN
Output Template:  AERO.OUT
```

Under these assumptions let us consider the execution of procedure EX1.

The statement RVU IDB causes the database review to be executed for input data. This means that the values of variables listed in the template AERO.IN will be found in the configuration database CASE3A. The review, of course, allows the designer to modify these values as appropriate.

The statement EX APPL causes the execution of the application program AERO to commence. The process begins by automatically extracting the values of input data, those just reviewed, from CASE3A. Execution procedes as usual and then final results, if any, are automatically stored back in CASE3A.

The final statement RVU ODB once again executes the database reviewer. This time the values presented correspond to those listed in the output data template AERO.OUT.

The idea of not requiring the names of application programs and the configuration database to be specified in the procedure adds significant flexibility to the utility of the procedure. By simply changing the values of variables associated with the workspace, a totally different design execution would be addressed.

6.2 Get Command: The Basics

The procedure EX1 in the previous example executes only under the assumptions of the AUTODEFAULT mode being set and certain workspace variables being defined. That is no real problem for the user of the full AIDE environment, as AIDE commands such as ACTIVATE allow such a user to set up to meet those assumptions prior to execution of EX1. Such an AIDE command does not address the problem facing a procedure writer attempting to address the needs of designers executing AIDE in the automatic execution mode.

Certainly the procedure writer could add a command such as ACT APPL AERO. This would cause workspace variables related to the application program to take on appropriate values associated with AERO. However the flexibility of not naming workspace variables mentioned at the end of the last section is completely lost. That is, suppose a user wants to run the WEIGHTS program instead.

The GET command was built to address this situation directly. When included in a procedure file, it is set up to obtain information from the procedure user for use later in the procedure. For example, the addition of the statement GET APPL to EX1 would initiate the following actions. First, a list of all project associated application programs is presented and the user's choice is solicited. Once made that choice is automatically activated as the current workspace value. Below is a collection of GET statements that cause a similar response.

```
GET APPL        - for application programs
GET CFG         - for configuration databases
GET PROC        - for a new procedure
GET TPL         - for a database reviewing template
GET WS          - for a new workspace
```

With this in mind consider the procedure EX2.

```
GET CFG
GET APPL
RVU IDB
EX APPL
RVU ODB
```

The first two lines guarantee that the appropriate workspace variables have been defined with the procedure user's choice.

6.3 Get Command: Procedure Flow Control

Another typical need to be addressed when building procedures is the need to repeat portions of the procedure or, at the procedure users choice, jump to other portions of the procedure. During the execution of a procedure, AIDE keeps track of its position within the procedure via a procedure counter. The GET command provides several techniques for modifying the value of the procedure counter.

The procedure counter can be changed by the addition of a numerical offset. The offset may be either positive or negative. Alternately the procedure counter may be reset by jumping to a labelled statement within the procedure. In either case the jump can be made conditional upon the response of the procedure user. Examples of each type of procedure modification follow.

The form for an unconditional jump via a numerical offset is GET JMPC <n> where <n> represents the offset. That is pc = pc + n.

Labels are placed into a procedure as a comment statement, since they are non-executable, with a special format: C LABEL <label_id> where <label_id> is any sequence of alphanumeric information. The statement GET JMPL <label_id> will search the procedure for the appropriate label and set the counter to the statement following that label.

The conditional jumps depend on the user's response to a procedure generated query. The format for a conditional jump based upon a numeric offset is GET QRYC <n> <query_text> where <n> is again the offset and <query_text> is text for which the procedure user is expect to provide a yes or no response. When the first character of the response is Y the jump is pc = pc + n, otherwise pc = pc + 1.

Finally a conditional jump to a label is made in a similar fashion, GET QRY <label_id> <query_text> where <label_id> and <query_text> are as defined above.

As an example, consider procedure EX3 (an extension of EX2).

```
GET CFG
GET APPL
C LABEL:  EXECUTE
RVU IDB
EX APPL
RVU ODB
GET QRYC 4       NEW PROGRAM (Y=YES):
GET QRYC 5       NEW DATABASE (Y=YES):
```

```
GET QRYL EXECUTE     RERUN PROGRAM SEQUENCE (Y=YES):
GET JMPL  DONE
GET APPL
GET JMPC  -4
GET CFG
GET JMPC   -5
C LABEL:  DONE
L
```

Note that there are two label statements: EXECUTE and DONE. The seventh statement, GET QRYC 4 NEW PROGRAM (Y=yes):, causes NEW PROGRAM (Y=yes) to be displayed and the program to wait for user input. If the first character of that user input is Y then pc = pc + 4. That is, the procedure will execute statement GET APPL next. Any other response causes pc = pc + 1. The ninth statement, GET QRYL EXECUTE RERUN PROGRAM SEQUENCE (Y=yes):, works in a similar fashion except that a Y response causes the pc to reset so that RVU IDB is the next statement executed. The GET JMPC and GET JMPL statements work in the obvious manner. Finally the L commands causes the user to logout.

6.4 Get Command: Menu Manipulation

The trouble with procedure EX3 is that it always forces a user through a series of three queries whether they are necessary or not. This can become quite tedious and it is generally more user friendly to present the procedure user a list of choices and execute the portion of the procedure that corresponds to the user choice.

The GET MENU <n> command provides this service. Here <n> represents a menu number. Before continuing it is necessary to briefly consider some of the files required to define a procedure, say EX4. The procedure commands themselves must be in a file named EX4.PROC. For this example, the statement GET MENU 3 refers to the menu contained in the file EX4.PROC.3. In addition EX4.PROC.3 would have lines which follow the following format <label_id><menu_text_choice>. Here <label_id> is a label identification, maximum of 8 characters, appearing in EX4. <Menu text_choice> is one of the choices presented to the user.

A GET MENU 3 statement would execute the following sequence of events. First the menu choices contained in EX4.PROC.3 would be presented in order to the user. The AIDE system would then await the users choice. An automatic GET JMPL would be performed to the <label_id> associated with that menu choices.

What follows, procedure EX4, is a menu modified version of EX3.

```
GET CFG
GET APPL
C LABEL:  MENU
GET MENU 3
```

```
GET JMPL MENU
C LABEL:  DATAB
GET CFG
GET JMPL MENU
C LABEL:  PROG
GET APPL
GET JMPL MENU
C LABEL:  INPUT
RVU IDB
GET JMPL MENU
C LABEL:  OUTPUT
RVU ODB
GET JMPL MENU
C LABEL:  EXE
EX APPL
GET JMPL MENU
C LABEL:DONE
L
```

In addition to the procedure file EX4.PROC, the menu file EX4.PROC.3 should be built to contain

```
DATAB           CHANGE THE CURRENT CONFIGURATION DATABASE
PROG            CHANGE THE CURRENT APPLICATION PROGRAM
INPUT           REVIEW THE INPUT DATA
OUTPUT          REVIEW THE OUTPUT DATA
EXE             EXECUTE THE APPLICATION PROGRAM
DONE            EXIT THE PROCEDURE
```

Again the statement GET MENU 3 causes the following menu to be presented.

```
1 - CHANGE THE CURRENT CONFIGURATION DATABASE
2 - CHANGE THE CURRENT APPLICATION PROGRAM
3 - REVIEW THE INPUT DATA
4 - REVIEW THE OUTPUT DATA
5 - EXECUTE THE APPLICATION PROGRAM
6 - EXIT THE PROCEDURE
```

A choice of, say, 4 would cause RVU ODB to be the next statement executed.

6.5 GET COMMAND: EXTRAS

There are a few other small points to be made concerning the GET statement and the development of procedures.

The command GET LACT puts an activation time into the workspace log. It is recommended that procedures written for use in the automatic execution mode begin with this command, as this will simplify the intepretation of command sequencing in the log.

The command GET MODE is used to insure that AIDE expects interactive input from the procedure user. In general, when a

procedure is active, commands are being sent to the AIDE command processor from the procedure, not expecting feedback from the user. Thus with the exception of the GET command itself, other AIDE commands expecting input from the user should be preceded by a GET MODE Command.

Finally as noted earlier in this section, the AUTODEFAULT should be set. For users in the automatic execution mode this should be preset in their AIDE login variable file to true. Aside: This file will have the name <user_id>.VAR. Users of the full AIDE environment may set AUTODEFAULT with the ACTIVATE, EDITVAR or SETLOGIN commands.

Section 7
OF FILES, DATABASES AND OTHER THINGS

An AIDE user file directory will contain a large variety of files. Though an explanation of each of these files follows, there would generally be little reason for a general user to become involved with any of the internal details or naming conventions used in these files. Such details can and generally should be left for the AIDE system to monitor, freeing the user to concentrate upon the design process itself.

7.1 The Login Files

There are two files whose presence is required prior to a user being allowed to log into the AIDE system. Designers interested in using the AIDE system should contact an AIDE system administrator to ensure the proper definition and setup of these files. Let the users identification be represented by <user id>. The two files are <user_id>.VAR and <user_id>_VARS; details follow.

A. The <user id>.VAR File

<user_id>.VAR - holds values as defined for the following list of variables. A "*" indicates the variables must be defined at login.

*USRID	-	the user's login identification
*USRLVL	-	the user's level of familarity with the AIDE system, range: 1..3 where 3=> experienced
MENUCMD	-	boolean indicates if the command last issued was selected from a menu
REFCFG	-	the name of the reference configuration database
APPLTPL	-	the name of a special data template for a given application program
REFTMPL	-	the name of a user selected data template.
APPLTMPLIN APPLTMPLOUT	-	The names of input and output templates associated with the given application program.
REFPCF	-	The name of the reference procedure command file
CURPROG	-	The name of the reference application program
*WKSP	-	The name of the workspace. If no specific workspace is used a temporary workspace should be listed with the name T$.
*CURMENU	-	The name of the menu to be displayed, usually "UTILITY SELECTION (MAIN)" at login.

*CURMENUIDX	-	The index of the login menu, usually "1".
*INUNIT	-	The logical unit number from which input should be received, usually "1" for the PRIMOS system.
*OUTUNIT	-	The logical unit number to which output would be directed, usually "1" for the PRIMOS system.
*PRINTLEVL	-	Indicates the amount of extra help a user desires from AIDE, range: 1-4 where 1=> full menus, help, etc and 4=> prompt only.
*HOMEUFD *BASEUFD *PROGUFD	-	The file directories which contain information pertinent to this project. HOME=> user, BASE=> AIDE specific information & PROG=> related programs and data.
*MENULIST	-	An encoded stack listing the values of the menus used, usually intialized to "!".
*SEQEXEC	-	Boolean, used to indicate when a sequence of program is executing, usually initialized to "F".
*SEQNUM	-	The number of the program being executed, usually initialized to "0".
*PROCSP	-	Procedure stack pointer, since procedures may call other procedures, a stack for tracking the executing procedures has been implemented usually initialized to "0".
*PROCPC	-	Procedure counter, the point of current execution in a given procedure, usually initialized to "0".
*PROCEX	-	Boolean, used to indicate when a procedure is in execution, usually initialized to "F".
*EMPTYPROCSTACK	-	Boolean, used to indicate the existence of stacked procedures, usually initialized to "T".
*AUTODEFAULT	-	Boolean, used to indicate that AIDE commands given without specifying object names should default to values defined by the current workspace, usually initialized to "T".
EXPROCFILE	-	The name of the currently executing procedure file.

B. The <user id> VARS File

<user id>_VARS - a PRIMOS global varialbe file used to pass information between programs.

REF_CFG - same as "REFCFG" above
LOG_UNIT - logical unit number on which the logfile is operating.
AIDE_UNIT - logical unit number on which the AIDE program is operating.
SYS_CMD - will contain the value of the last command to be passed to the system
*USR_ID - same as "USRID" above
CUR_CMD - same as "CURCMD" above

C. Sample Login Files for a Basic User

For this sample, the <user_id> is JLS1 and represents login files of the type required for the automatic execution mode.

```
SLIST JLS1.VAR
USRID                    JLS1
USRLVL                      1
MENUCMO                     F
REFCFG
APPLTMPL
REFTMPL
APPLTMPLIN
APPLTMPLOUT
REFPCF                   EX4
CURPROG
WKSP                     JLS1
CURMENU                  UTILITY SELECTION (MAIN)
CURMENUIDX                  1
INUNIT                      1
OUTUNIT                     1
PRINTLEVL                   4
HOMEUFD                  VAR>JLS>JLS.DIR
BASEUFD                  VAB>AIDE
PROGUFD                  VAB2>CSC
MENULIST                 !
SEQEXEC                     F
SEQNUM                      0
PROCSP                   0
PROCPC                   1
PROCEX                      T
EMPTYPROCSTACK              T
AUTODEFAULT                 T
EXPROCFILE               EX4.PROC
OK,

DEFINE_GVAR JLS1_VARS
LIST_VAR
REF_CFG
LOG_UNIT                    5
AIDE_UNIT                 126
SYS_CMD                  R VAB>AIDE>REVIEW NUDB
.USR_ID                  JSL1
```

```
.CUR_CMD_____________________L
OK,
```

D. Sample Login Files for the Full AIDE Environment for sample the <user_id> is JLS.

```
LIST JLS.VAR
USRID_____________________JLS
USRLVL_____________________2
MENUCMD____________________E
REFCFG
APPLTMPL
REFTMPL
APPLTMPLIN
APPLTMPLOUT
REFPCF
CURPROG
WKSP_____________________T$
CURMENU_____________UTILITY SELECTION (MAIN)
CURMENUIDX_____________1
INUNIT_____________1
OUTUNIT_____________1
PRINTLEVL_____________2
HOMEUFD_____________VAB>JLS>JLS.DIR
BASEUFD_____________VAB>AIDE
PROGUFD_____________VAB2>CSC
MENULIST_____________!
SEQEXEC_____________F
SEQNUM_____________0
PROCSP_____________0
PROCPC_____________0
PROCEX_____________F
EMPTYPROCSTACK_____________T
AUTODEFAULT_____________T
EXPROCFILE_____________
OK,

DEFINE GVAR JLS_VARS
LIST VAR
REF_CFG_____________SSCDM1
LOG_UNIT_____________5
AIDE_UNIT_____________126
SYS_CMD_____________JOB -ST
USR_ID_____________JLS
CUR_CMD_____________L
OK,
```

7.2 Workspace Files

An AIDE user should refer to a workspace only with its identifying name. Let <WS_name> represent that name. The following types of files would always appear in the user's file directory for each workspace.

<WS_name>.WS - A file which has a structure identical with that of the <user_id>.VAR file

previously discussed. This information is sufficient to represent the current state of a configuration development.

<WS_name>.WS.D - A description file to which the user may add information concerning the specifics of project development found in this workspace (see the RD and CD commands).

<WS_name>.LOG - A file containing the sequence of commands issued by the user along with potential system and user comments.

7.3 Configuration Database Files

An AIDE user should refer to a configuration database only by its identifying name. Let <cfg_name> represent that name. The following types of files may appear in the user's file directory and are associated with a user configuration database. These types may also appear in the project file directory where they are associated with master configuration databases. AIDE permits users only to copy master databases. User databases may be copied, modified by the user or program, or deleted.

<cfg_name>.CFG - represents a subdirectory. This subdirectory contains all database information, schema and data for the given configuration.

<cfg_name>.CFG.D - A description file to which the user may add information concerning the project associated data.

7.4 Procedure Files

An AIDE user should refer to a procedure only by its identifying name. Let <proc_name> represent that name. The following types of files may appear in the user's file directory and are associated with user procedures. These types may also appear in the project file directory where they are associated with master procedures.

<proc_name>.PROC - A file which contains a collection of AIDE commands for processing by the AIDE command interpretor.

<proc_name>.PROC.D - A description file in which a user may describe uses and parameters for the associated procedure.

<proc_name>.PROC.<n> - A menu file associated with this procedure. See the previous section for a complete explanation.

7.5 Template Files

An AIDE user should refer to a template only with its identifying name. Let <tpl_name> represent that name. The following types of files may appear in the user's file directory and are associated with a user template.

<tpl_name>.REV - A file which contains configuration data locations in the database in a format appropriate for the reviewer program.

<tpl_name>.TPL.D - A description file the user may use to describe the type of data identified by this template.

In addition to the above file types the project file directory may contain several additional types of template files. Let <appl_name> represent the name of an application program.

<appl_name>.IN.REV, <appl_name>.OUT.REV - These files contain configuration data location for input and output, respectively, in a format appropriate for the reviewer program.

<appl_name>.IN.TPL.D, <appl_name>.OUT.TPL.D - Description files of the input and output of the indicated application program.

7.6 Application Program Files

An AIDE user should refer to an application program only by its identifying name. Let <appl_name> represent that name. The following files would generally appear in the main project file directory.

<appl_name>.APPL.D - A description file detailing the use of the associated program.

<appl_name>
<appl_name>.SEG - One or more of these will comprise the executable elements of the application program.
<appl_name>.CPL

7.7 Batch Files

Batch Files are created when time consuming application programs are submitted to the system's batch processors for executing, freeing the AIDE processor for further work. An AIDE user should refer to a batch file only by its identifying name. Let BAT <n> represent that name, where <n> is some integer. The following files may appear in the user's file directory.

BAT<n>.CPL - A file containing instructions for the batch processor.

BAT<n>.BAT - A log file containing information regarding the results of the batch execution. Note: this file may be displayed in full AIDE via the command TY BAT BAT<n>.

7.8 Temporary Files

The following files may appear from time to time an AIDE session. They will be deleted at logout and should not be referred to or deleted by an AIDE user, as they keep track of valuable intermediate data.

T$.VAR - file which contains the current values of all status variables throughout an AIDE session.

T$.LOG - the current command log, if the user has not assigned or recalled a permanent workspace.

T$.PROC - file used to store a new procedure until it is saved and named by the user.

T$.APPL - file used to keep track of an execution sequence for a given application.

T$.PROCSTACK - file used to store the stack; formed as a procedure to call other procedures.

T$.TPL,T$,S$ - files used during the interactive editing of a data template

Table of AIDE Associated Files

FILE TYPE	GENERIC NAME	FILENAME	USER FILE	MASTER FILE
LOGIN FILES	<USER_ID>	<USER_ID>.VAR	HOMEUFD	
	<USER_ID>_VARS	HOMEUFD		
WORKSPACE FILES	<WS_NAME>	<WS_NAME>.WS	HOMEUFD	
	<WS_NAME>.WS.D	HOMEUFD		
	<WS_NAME>.LOG	HOMEUFD		
CONFIGURATION DATABASES	<CFG_NAME>	<CFG_NAME>.CFG	HOMEUFD	PROGUFD
	<CFG_NAME>.CFG.D	HOMEUFD	PROGUFD	
PROCEDURE FILES	<PROC_NAME>	<PROC_NAME>.PROC	HOMEUFD	PROGUFD
	<PROC_NAME>.PROC.D	HOMEUFD	PROGUFD	
	<PROC_NAME>.PROC.<n>	HOMEUFD	PROGUFD	
TEMPLATE FILES	<TPL_NAME>	<TPL_NAME>.REV	HOMEUFD	
	<TPL_NAME>.TPL.D	HOMEUFD	PROGUFD	
	<APPL_NAME>.IN.REV		PROGUFD	
	<APPL_NAME>.OUT.REV		PROGUFD	
APPLICATION PROGRAMS	<APPL_NAME>	<APPL_NAME>		PROGUFD
	<APPL_NAME>.SEG		PROGUFD	
	<APPL_NAME>.CPL		PROGUFD	
	<APPL_NAME>.APPL.D		PROGUFD	

FIGURE 7-8

Section 8
Aide Commands Summary

8.1 Permanent Menu Commands

H - Help

D - Directory: gives a listing of currently available files for configuration databases, data templates and workspaces.

I - Inventory: gives a listing of currently available application programs and procedures.

S - System Command: allows the user to pass a command to the operating system.

C - Comment: allows the user to place a comment in the command log.

T - Toggle the display mode; rotate among:

1. Full information
2. Status variables and current utility menu
3. Status variables only
4. Prompt only.

R - Return to Previous Menu

Q - Quit this Sequence of Menus: initialize back to the selection menu

L - Logout.

<cr> - Carriage Return: clears the screen and reprints menu information if active

0 - zero: cancels a command sequence.

<Break> - halts an execution sequence.

8.2 Full AIDE Commands

Short Form	Full Form	
ACT	ACTIVATE	Associate the indicated object with the user's workspace. Ex: ACT CFG SSCDM1 Allowable Objects: APPL, CFG, OTPL, OTPL, PROC, TPL, WS, UTL, BUFD, PUFD, DFLT.
CD	CD	Change description of the indicated object. Ex: CD TPL USERINFO Allowable Objects: APPL, CFG, ITPL, TPL, PROC, TPL, WS.
CL	CLEARLOG	Removes prior information from a cluttered command log. Ex: CL D Allowable Objects: D - prior to a given date T - total, a new log started
CN	CNAME	Change the name of a file as indicated. Ex: CN TPL OLDTPL NEWTPL Allowable Objects: APPL, CFG, PROC, TPL, WS.
CP	COPY	Copy one file to another. Ex: CP CFG SSCDM1 SSCDM2 Allowable Objects: APPL, CFG, PROC, TPL, WS, FILE
ED	EDIT	Invoke an editor. Ex: ED PROC ITERATE Allowable Objects: LOG, PROC, TPL
EX	EXECUTE	Executes an indicated program or procedure Ex: EX APPL ATMOS Allowable Objects: APPL, PROC.
GET	GET	Allows interactive input during procedure execution. Ex: GET PROC Allowable Objects: APPL, CFG, JMPC, JMPL, LACT, MENU, MODE, PROC, QRYC, QRYL, TPL, WS

N	NEW	Create a new object. Ex: NEW WS Allowable Objects: CFG, PROC, TPL, UTL, WS.
PR	PRINT	Print a file Ex: PR LOG MYWS Allowable Objects: LOG, PROC, FILE, BAT
PRVU	PRINTVU	Print a view of the database. Ex: PRVU IDB Allowable Objects: IDB, ODB, TPL
RD	RD	Read a file description Ex: RD APPL ELECT Allowable Objects: APPL, CFG, ITPL, OTPL, PROC, TPL, WS
RM	REMOVE	Remove a file from your directory Ex: RM CFG SSCDM1 Allowable Objects: APPL, CFG, PROC, TPL, WS, FILE
RVU	REVIEW	Review data from the configuraton database. Ex: RVU IDB Allowable Object: IDB, ODB, TPL.
SA	SAVE	Save the indicated object Ex: SA PROC MYPROC Allowable Objects: PROC, WS.
SLOG	SETLOGIN	Define your login characteristics Ex: SLOG No objects used in this command.
STAT	STATUS	Get the status of batch jobs. Ex: STAT No objects used in this command.
SUB	SUBMIT	Submit a job for batch processing. Ex: SUB APPL ELECT Allowable Objects: APPL
TY	TYPE	Type the indicated file. Ex: TY PROC MYPROC Allowable Objects: LOG, PROC, FILE, BAT

VAR	EDITVAR	Change values of the current status variables in the workspace Ex: VAR No objects used in this command.

References

1. Jacky, J. P. and I. J. KaPet, "A General Purpose Data Entry Program", CACM, V. 26, No. 6, pp. 409-417, June 1983.

2. Dube, R. P. and M. R. Smith, "Managing Geometric Information With a Database Management System", IEEE Computer Graphics and Applications, V. 3, No. 7, pp. 57-62, October 1983.

3. Wilhite, A. W. and V. Crisp, A Relational Information System, ARIS User Manual, NASA/Langley Research Center, 1983.

Appendix A

Screens for the Sample Session of a Basic User

```
MASTER CONFIGURATIONS

DLKDB


USER CONFIGURATIONS

SSCDM1
MUDB
NEUDB

 SHOULD A NEW DATABASE BE CREATED FROM
 A MASTER CONFIGURATION (Y = yes):N
 CHOOSE A 'USER' CONFIGURATION DATABASE:SSCDM1.
```

SCREEN A1

MASTER APPLICATION PROGRAMS

CONFIG
FOOD
ATMOS
COOLER
CONTAM
ELECT
REACT
STACON
STRUCT
THERM
WASTE
WATMAN
CADP
CREWEQ
COMPAT
OUTPUT

CHOOSE AN APPLICATION PROGRAM: ELECT

SCREEN A2

```
MASTER TEMPLATES

CONF.IN
CONF.OUT
FOOD.IN
FOOD.OUT
ATMO.IN
ATMO.OUT
COMM.IN
COMM.OUT
WATR.IN
WATR.OUT
CONT.IN
CONT.OUT
ELEC.IN
ELEC.OUT
REAC.IN
REAC.OUT
STAC.IN
STAC.OUT
STRU.IN
STRU.OUT
THER.IN
THER.OUT
WAST.IN
WAST.OUT
CREW.IN
CREW.OUT
CADP.IN
CADP.OUT
COMP.OUT
COMP.IN
OUTP.IN
OUTP.OUT

USER TEMPLATES

A
AA
E.IN

CHOOSE A TEMPLATE: E.IN


     1 - CHANGE THE CURRENT CONFIGURATION DATABASE
     2 - CHANGE THE CURRENT APPLICATION PROGRAM
     3 - CHOOSE A USER DATA TEMPLATE
     4 - REVIEW DATA WITH THE USER'S TEMPLATE
     5 - REVIEW THE PROGRAM'S INPUT DATA
     6 - REVIEW THE PROGRAM'S OUTPUT DATA
     7 - EXECUTE THE APPLICATION PROGRAM
     8 - ENTER THE FULL 'AIDE' ENVIRONMENT
     9 - EXIT THE PROGRAM

 Input: # - 'n'th choice, R - reprint menu
```

SCREEN A3

C-2

L0	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	0.300000	0.300000		A1	SCALING FACTOR FOR PARASITIC LOAD	-
2	0.400000	0.400000		AA	FRACT. POW. FOR EMERG. ECLSS OPERATION	-
3	0.100000	0.100000		B	SYSTEM FRACT. EMERG. OPERATING POWER	-
4	10.0000	10.0000		B1	IB POWER CONVERSION UNIT SIZING CONSTANT	-
5	0.870000	0.870000		C1	DC CONVERTER SPECIFIC VOLUME	FT3/KW
6	0.300000	0.300000		C2	AC CONVERTER SPECIFIC VOLUME	FT3/KW
7	0.300000	0.300000		C3	VARIABLE FREQ. INVERTER SPEC. VOL.	FT3/KW
8	0.300000	0.300000		C4	HI VOLTAGE RECTIFIER SPECIFIC VOL.	FT3/KW
9	0.300000	0.300000		C5	SPECIFIC VOLUME OF BATTERY	FT3/KW
10	0.170000	0.170000		C6	BATTERY CHARGER-REGULATOR SPEC. VOL	FT3/KW
11	0.300000E-01	0.300000E-01		C7	IB SYS. PARASITIC LOAD SPEC. VOL.	FT3/KW
12	4.10000	4.10000		C8	IB PWR. CONVERSION UNIT SPEC. VOL.	FT3/KW
13	30.0000	30.0000		CHARWT	BATTERY CHARGER-REGULATOR WEIGHT	LB
14	0.500000	0.500000		DD	ALLOWABLE BATTERY DISCHARGE DEPTH	%
15	150.000	150.000		EAC	EXPERIMENTAL PROGRAM AC ENERGY REQ.	KWH
16	10.0000	10.0000		EDC	EXPERIMENTAL PROGRAM DC ENERGY REQ.	KWH
17	0.850000	0.850000		ETAA	SOLAR PANEL AREA UTILIZATION FACTOR	-
18	0.942000	0.942000		ETAAC	AC INVERTER EFFICIENCY	-
19	0.640000	0.640000		ETAB	BATTERY EFFICIENCY	-
20	0.910000	0.910000		ETABR	BATTERY REGULATOR & CHARGER EFFICIENCY	-
21	0.120000	0.120000		ETACES	SOLAR CELL EFFICIENCY AT 85 DEGREES C	-
22	0.500000	0.500000		ETACN	SOLAR CONCENTRATION EFFICIENCY	-
23	0.970000	0.970000		ETAD	POWER DISTRIBUTION EFFICIENCY	-
24	0.850000	0.850000		ETADC	DC CONVERTER EFFICIENCY	-
25	0.680000	0.680000		ETAFC	FUEL CELL FULL LOAD EFFICIENCY	-
26	0.920000	0.920000		ETAHF	ISOTOPE BRAYTON HI FREQUENCY CKT. EFF.	-
27	0.950000	0.950000		ETAHV	HIGH VOLTAGE RECTIFIER EFFICIENCY	-
28	0.700000	0.700000		ETAS	ESTIMATED EXPERIMENT SCHEDULING EFF.	-
29	0.930000	0.930000		ETAVF	VARIABLE FREQUENCY INVERTER EFFICIENCY	-
30	27.0000	27.0000		ETPCSR	IB PWR. SYS. PCU RATED AT PCR EFFICIENCY	-

Mn - Next page, R - Reprint page, n - modify L0 n, Ln - n L0 per page, E - End and savemods, Q - Quit, H - Help, PA - Print all

EDIT
>

SCREEN A4

LS	PRESENT VALUE	OLD VALUE	SUBSCRIPT	NAME	DESCRIPTION	UNITS
31	0.810811E-01	0.810811E-01		FIXAG		
32	0.000000	0.000000		FIXAA		
33	1.00000	1.00000		FP	FRACT. PEAK DC POWER SUPPLIED BY BATTERY	-
34	200.000	200.000		H	ORBITAL ALTITUDE	NMI
35	1	1	1, 1, 1	IOPT	PRINTOUT/DISPLAY OPTIONS	-
36	1	1	1, 1, 2			
37	1	1	1, 1, 3			
38	0	0	1, 1, 4			
39	3	3	1, 1, 5			
40	0	0	1, 1, 6			
41	0	0	1, 1, 7			
42	1	1	1, 1, 8			
43	1	1	1, 1, 9			
44	1	1	1, 1, 10			
45	5	5	1, 1, 11			
46	5	5	1, 1, 12			
47	5	5	1, 1, 13			
48	5	5	1, 1, 14			
49	5	5	1, 1, 15			
50	5	5	1, 1, 16			
51	5	5	1, 1, 17			
52	5	5	1, 1, 18			
53	5	5	1, 1, 19			
54	5	5	1, 1, 20			
55	1	1	1, 1, 21			
56	1	1	1, 1, 22			
57	1	1	1, 1, 23			
58	1	1	1, 1, 24			
59	1	1	1, 1, 25			
60	1	1	1, 1, 26			

Mn - Next page, R - Reprint page, n - modify LS n, Ln - n LS per page, E - End and savemods, Q - Quit; H - Help, PA - Print all

EDIT
)

SCREEN A5

LS#	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
101	0.300000	0.300000		Z	SOLAR PANEL EQUILIB. TEMPERATURE FACTOR	-
102	71.2000	71.2000		ZETAG	DENSITY OF LOX	LB/FT3
103	0.3000000E-01	0.3000000E-01		ZETAFC	FUEL CELL POWER DENSITY	KW/LB
104	4.50000	4.50000		ZETAH	LH2 DENSITY	LB/FT3
105	2.00000	2.00000		CONR	CONCENTRATOR RATIO	-
106	90.0000	90.0000		T	RESUPPLY INTERVAL	DAYS
107	20.0000	20.0000		T3	PRESSURIZED UNMANNED CHECKOUT DAYS	DAYS
108	90.0000	90.0000		T4	STATION EXPENDABLES SUPPLY AT LAUNCH	DAYS
109	24.0000	24.0000		TC	STEADY STATE & PEAK LOAD CYCLE DURATION	HR
110	5.00000	5.00000		TE	EMERGENCY POWER SUPPLY DURATION	HR
111	11.0000	11.0000		TP	DURATION OF PEAK LOAD	HR
112	0.500000	0.500000		TS	ISOTOPE BRAYTON STARTUP DURATION	HR

Mn - Next page, R - Reprint page, n - modify LS n, Ln - n LS per page, E - End and savecode, Q - Quit, H - Help, PA - Print all

EDIT
>

LOI	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	0.570000	0.570000		C1	DC CONVERTER SPECIFIC VOLUME	FT3/KW
2	0.300000	0.300000		C2	AC CONVERTER SPECIFIC VOLUME	FT3/KW
3	0.300000	0.300000		C3	VARIABLE FREQ. INVERTER SPEC. VOL.	FT3/KW
4	0.300000	0.300000		C4	HI VOLTAGE RECTIFIER SPECIFIC VOL.	FT3/KW
5	0.300000	0.300000		C5	SPECIFIC VOLUME OF BATTERY	FT3/KW
6	0.170000	0.170000		C6	BATTERY CHARGER-REGULATOR SPEC. VOL	FT3/KW
7	30.0000	30.0000		T3	PRESSURIZED UNMANNED CHECKOUT DAYS	DAYS
8	90.0000	90.0000		T4	STATION EXPENDABLES SUPPLY AT LAUNCH	DAYS
9	24.0000	24.0000		TC	STEADY STATE & PEAK LOAD CYCLE DURATION	HR
10	5.00000	5.00000		TE	EMERGENCY POWER SUPPLY DURATION	HR
11	11.0000	11.0000		TP	DURATION OF PEAK LOAD	HR
12	0.500000	0.500000		TS	ISOTOPE BRAYTON STARTUP DURATION	HR

Mn - Next page, R - Reprint page, n - modify LO n, Ln - n LO per page, E - End and savecode, Q - Quit, H - Help, PA - Print all

EDIT
>

```
 8 STARTING EXECUTION OF ELECT
ENTER DATA BASE NAME
DATA08
. SUBROUTINE ELECT INTERMEDIATE PRINT DUMP


   ETAPC =        0.8884000E+00
      TA =        0.1381848E+01
     PSS =        0.1084177E+03
    P(1) =        0.1000331E+03
    PDCS =        0.5530657E+02
    PACS =        0.4711218E+02
   PKDCS =        0.6136218E+02
   PKACS =        0.5229449E+02
   PSDCE =        0.7990866E-03
   PSACE =        0.1198630E-01
    PDCE =        0.7607331E-03
    PACE =        0.1141100E-01
  PPKDC =        0.8444136E-03
  PPKAC =        0.1266620E-01
     PAC =        0.4718367E+02
     PDC =        0.5530633E+02
  PPKACP =        0.5230715E+02
  PPKDCP =        0.6139001E+02
      PE =        0.4025170E+02
      TD =        0.5304678E+00
      TB =        0.7907740E+00
     SPA =        0.3402238E+03
     EB1 =        0.5720293E+02
    TEMP =        0.8740135E+02
    ETAC =        0.7701749E-01
 ONEGAK =        0.1000000E+01
      AS =        0.7362479E+04
     CUC =        0.1472496E+04
      AP =        0.8861740E+04
     UUP =        0.9094826E+04
     ACN =        0.3464696E+05
     UCN =        0.3464696E+04
     VOE =        0.2106802E+04
1 SUBROUTINE ELECT INTERMEDIATE PRINT DUMP


     EB2 =        0.0000000E+00
    W(3) =        0.0000000E+00
     EB3 =        0.2012585E+03
      EB =        0.4025170E+03
    W(6) =        0.3858870E+05
    W(7) =        0.6020202E+03
    W(8) =        0.0000000E+00
     WSP =        0.5332963E+05
     VSP =        0.2070000E+03
     GSP =        0.6949778E+05
     WLB =        0.1000000E+03
$$$$ STOP


    1 - CHANGE THE CURRENT CONFIGURATION DATABASE
    2 - CHANGE THE CURRENT APPLICATION PROGRAM
    3 - CHOOSE A USER DATA TEMPLATE
    4 - REVIEW DATA WITH THE USER'S TEMPLATE
    5 - REVIEW THE PROGRAM'S INPUT DATA
    6 - REVIEW THE PROGRAM'S OUTPUT DATA
    7 - EXECUTE THE APPLICATION PROGRAM
    8 - ENTER THE FULL 'AIDE' ENVIRONMENT
    9 - EXIT THE PROGRAM

Input: n - 'n'th choice, R - reprint menu
```

LS	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	24847.0	24847.0		AOM		
2	8681.74	8681.74		AP		
3	7368.48	7368.48		AS		
4	365.000	365.000		BYD		
5	0.000000	0.000000		CAPWFC		
6	0.000000	0.000000		CAPWHT		
7	0.000000	0.000000		CAPWOT		
8	1478.50	1478.50		CUC		
9	0.000000	0.000000		DLFC		
10	0.000000	0.000000		DLFCT		
11	0.000000	0.000000		DOTL0		
12	0.000000	0.000000		DOTL0T		
13	0.000000	0.000000		DOTLH		
14	0.000000	0.000000		DOTLHT		
15	0.000000	0.000000		DTLFCS		
16	0.000000	0.000000		DTLFCT		
17	402.517	402.517		EB		
18	57.2029	57.2029		EB1		
19	0.000000	0.000000		EB2		
20	201.258	201.258		EB3		
21	0.582400	0.582400		ETABC		
22	0.770175E-01	0.770175E-01		ETAC		
23	0.000000	0.000000		ETAPCS		
24	1	1		INDEX		
25	0	0		IX		
26	1.00000	1.00000		OMEGAK	SOLAR CELL AREA ORIENTATION FACTOR	-
27	100.033	100.033	1, 1, 1	P		
28	0.000000	0.000000	1, 1, 2			
29	0.000000	0.000000	1, 1, 3			
30	0.000000	0.000000	1, 1, 4			

Nn - Next page, R - Reprint page, n - modify LS n, Ln - n LS per page, E - End and savemods, Q - Quit, H - Help, PA - Print all

EDIT
)

ORIGINAL PAGE IS
OF POOR QUALITY

LN	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
31	0.000000	0.000000	1, 1, 5	P		
32	0.000000	0.000000	1, 1, 6			
33	0.000000	0.000000	1, 1, 7			
34	0.000000	0.000000	1, 1, 8			
35	0.000000	0.000000	1, 1, 9			
36	0.000000	0.000000	1, 1, 10			
37	47.1836	47.1836		PAC		
38	0.114110E-01	0.114110E-01		PACE		
39	0.000000	0.000000		PACP		
40	47.1188	47.1188		PACS		
41	0.000000	0.000000		PAU8		
42	0.119863E-01	0.119863E-01		PBACE		
43	0.799087E-03	0.799087E-03		PBDCE		
44	55.3063	55.3063		PDC		
45	0.780733E-03	0.780733E-03		PDCE		
46	0.000000	0.000000		PDCP		
47	55.3056	55.3056		PDCS		
48	40.2517	40.2517		PE		
49	0.000000	0.000000		PHU		
50	52.2945	52.2945		PKACS		
51	61.3892	61.3892		PKDCS		
52	0.128682E-01	0.128682E-01		PPKAC		
53	52.3072	52.3072		PPKACP		
54	0.844414E-03	0.844414E-03		PPKDC		
55	61.3900	61.3900		PPKDCP		
56	0.000000	0.000000		PPUF		
57	102.418	102.418		PSB		
58	0.000000	0.000000		PUF		
59	0.000000	0.000000		QFC		
60	0.000000	0.000000		QPC		

Nn - Next page, R - Reprint page, n - modify L0 n, Ln - n L0 per page, E - End and savemode, Q - Quit, H - Help, PA - Print all

EDIT
)

ORIGINAL PAGE IS
OF POOR QUALITY

```
$ US ACTIVATED ON : 03 Nov 86 13:59:52 Monday
$ US ACTIVATED ON : 03 Nov 86 13:59:56 Monday
>D
$ DIRECTORY
>ACT CFG SDCDM1
$ ACTIVATING CONFIGURATION DATABASE $
>I
$ INVENTORY
>ACT APPL ELECT
>D
$ DIRECTORY
>ACT TPL E.IN
C LABEL: AA
>RUU IDB
>RUU
>EX APPL ELECT
$    EXECUTING ELECT
>RUU ODB
$ EXITING PROCEDURE EX3.PROC
$ SAVED CURRENT STATUS OF ATTACHED WORKSPACE
>SA US JLS1
>TY LOG JLS1


ENTER COMMAND:
```

SCREEN A11

ORIGINAL PAGE IS
OF POOR QUALITY

SCREEN A12

```
GET LAST
GET CFG
GET APPL
GET TPL
C LABEL: A
GET MENU
GET JNPL
C LABEL: B
GET CFG
GET JNPL
C LABEL: C
GET APPL
GET JNPL
C LABEL: D
RVU IDB
GET JNPL
C LABEL: E
EX APPL
GET JNPL
C LABEL: F
RVU ODB
GET JNPL
C LABEL: G
GET MODE
C LABEL: H
GET TPL
GET JNPL
C LABEL: I
RVU
GET JNPL
C LABEL: J
GET MODE

ENTER COMMAND:
```

Appendix B

Screens for the Sample Session Using the Full AIDE Environment

STATUS

WORKSPACE: T6 REF_CONFIG: UTIL_IN_USE: UTILITY SELECTI
APPLIC_PROG: REF_TEMPLATE: REF_PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
(CR) - Clear the screen and relist the menu

UTILITY SELECTION (MAIN) COMMAND FORMAT

1 - WORKSPACE CONTROL UTILITY ACT UTL <WSC >
2 - DATA MODIFICATION UTILITY ACT UTL <DATA>
3 - APPLICATION EXECUTION UTILITY ACT UTL <APEX>
4 - PROCEDURE EXECUTIOIN UTILITY ACT UTL <PREX>
5 - PROCEDURE BUILDING UTILITY ACT UTL <PBLD>
6 - TEMPLATE BUILDING UTILITY ACT UTL <TBLD>

STATUS

WORKSPACE: TS REF_CONFIG: UTIL_IN_USE: WORKSPACE CONTR
APPLIC PROG: REF TEMPLATE: REF PROFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
    RETURN to the MAIN MENU
L - LOGOUT
@ - CANCEL the current command
(CR) - Clear the screen and relist the menu

WORKSPACE CONTROL

			COMMAND FORMAT	
1 -	READ DESCRIPTION	- WORKSPACE	RD WS	<name>
2 -		- CONFIGURATION	RD CFG	<name>
3 -		- TEMPLATE	RD TPL	<name>
4 -		- APPL. PROG.	RD APPL	<name>
5 -		- PROCEDURE	RD PROC	<name>
6 -	CLEAR LOG OF OLD INFORMATION		CL	
7 -	TYPE	- COMMAND LOG	TY LOG	<name>
8 -		- PROCEDURE	TY PROC	<name>
9 -	NEW	- WORKSPACE	N WS	
10 -		- CONFIGURATION	N CFG	<base>
11 -	COPY	- WORKSPACE	CP WS	<f,to>
12 -		- PROCEDURE	CP PROC	<f,to>
13 -	ACTIVATE	- WORKSPACE	ACT WS	<name>
14 -		- CONFIGURATION	ACT CFG	<name>
15 -		- TEMPLATE	ACT TPL	<name>
16 -		- APPL. PROG.	ACT APPL	<name>
17 -		- UTILITY	ACT UTL	<menu>
18 -		- INPUT TEMPL	ACT ITPL	
19 -		- OUTPUT TEMPL	ACT OTPL	
20 -		- PROCEDURE	ACT PROC	<name>
21 -		- PROGRAM UFD	ACT PUFD	<path>
22 -	SAVE TEMPORARY	- WORKSPACE	SA WS	<name>
23 -		- PROCEDURE	SA PROC	<name>
24 -	REMOVE FROM UFD	- WORKSPACE	RM WS	<name>
25 -		- CONFIGURATION	RM CFG	<name>
26 -		- TEMPLATE	RM TPL	<name>
27 -		- PROCEDURE	RM PROC	<name>
28 -	SET USER LOGIN CHARACTERISTICS		SLOG	

STATUS

WORKSPACE: T8 REF_CONFIG: UTIL_IN_USE: DATA REVIEW/MOD
APPLIC PROG: REF TEMPLATE: REF PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
$ - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

DATA REVIEW/MODIFICATION

		COMMAND	FORMAT
1 -	READ DESCRIPTION -TEMPLATE	RD TPL	<name>
2 -	-CONFIGURATION	RD CFG	<name>
3 -	ACTIVATE - CONFIGURATION	ACT CFG	<name>
4 -	VIEW AS - INPUT TMPL.	ACT ITPL	
5 -	VIEW AS - OUTPUT TEMPL.	ACT OTPL	
6 -	VIEW AS - ALT. TMPL.	ACT TPL	<name>
7 -	COPY - CONFIGURATION	CP CFG	<f,to>
8 -	- TEMPLATE	CP TPL	<f,to>
9 -	ACTIVATE TEMPLATE BUILDING UTILITY	ACT UTL	<TBLD>
10 -	REVIEW - INPUT DB	RVU IDB	
11 -	- OUTPUT DB	RVU ODB	
12 -	REVIEW DB WITH ACTIVE TEMPLATE	RVU	
13 -	PRINT - INPUT DB	PRVU IDB	
14 -	- OUTPUT DB	PRVU ODB	
15 -	PRINT DB WITH ACTIVE TEMPLATE	PRVU	
16 -	CHANGE CONFIGURATION DESCRIPTION	CD CFG	
17 -	REMOVE ALL DATA - INPUT & OUTPUT	RM CFG	
18 -	TYPE CURRENT COMMAND LOG	TY LOG	(-)

STATUS

WORKSPACE: T$ REF_CONFIG: UTIL_IN_USE: APPLICATION EXE
APPLIC PROG: REF TEMPLATE: REF PROFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

APPLICATION EXECUTION

			COMMAND FORMAT	
1	READ DISCRIPTION	- APPL. PROG.	RD APPL	<name>
2		- CONFIGURATION	RD CFG	<name>
3		- TEMPLATE	RD TPL	<name>
4	NEW	- CONFIGURATION	N CFG	<base>
5	ACTIVATE	- APPL. PROG.	ACT APPL	<name>
6		- CONFIGURATION	ACT CFG	<name>
7	ACTIVATE DB VIEW AS	- INPUT TEMPL.	ACT ITPL	
8		- OUTPUT TEMPL.	ACT OTPL	
9		- ALT. TEMPL.	ACT TPL	<name>
10	COPY	- CONFIGURATION	CP CFG	<f,to>
11	REVIEW DB WITH ACTIVE TEMPLATE		RVU	
12	REVIEW	- INPUT DB	RVU IDB	
13		- OUTPUT DB	RVU ODB	
14	PRINT DB WITH ACTIVE TEMPLATE		PRVU	
15	PRINT	- INPUT DB	PRVU IDB	
16		- OUTPTU DB	PRVU ODB	
17	EXECUTE CURRENT APPLICATION PROGRAM		EX APPL	<- >
18	CHANGE DESCRIPTION	- CONFIGURATION	CD CFG	<name>
19	REMOVE ALL DATA	- INPUT & OUTPUT	RM CFG	
20	TYPE CURRENT COMMAND LOG		TY LOG	<- >

SCREEN B4

STATUS

WORKSPACE: T$ REF_CONFIG: UTIL_IN_USE: PROCEDURE EXECU
APPLIC PROG: REF_TEMPLATE: REF_PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

PROCEDURE EXECUTION

			COMMAND FORMAT	
1	READ DESCRIPTION	- PROCEDURE	RD PROC	<name>
2		- CONFIGURATION	RD CFG	<name>
3		- APPL. PROG.	RD APPL	<name>
4		- TEMPLATE	RD TPL	<name>
5	TYPE A GIVEN COMMAND LOG		TY LOG	<name>
6	ACTIVATE	- PROCEDURE	ACT PROC	<name>
7		- CONFIGURATION	ACT CFG	<name>
8	ACTIVATE DB VIEW AS-	INPUT TEMPL.	ACT ITPL	
9		- OUTPUT TEMPL.	ACT OTPL	
10		- ALT. TEMPL.	ACT TPL	<name>
11	COPY	- PROCEDURE	CP PROC	<f,to>
12		- CONFIGURATION	CP CFG	<f,to>
13	NEW	- CONFIGURATION	N CFG	<base>
14	PROCEDURE BUILDING UTILITY		ACT UTL	<PBLD>
15	REVIEW THE DB WITH THE ACTIVE TMPL.		RVU	
16	REVIEW		RVU IDB	
17		- OUTPUT DB	RVU ODB	
18	PRINT THE DB WITH THE ACTIVE TEMPLAT		PRVU	
19	PRINT	- INPUT DB	PRVU IDB	
20		- OUTPUT DB	PRVU ODB	
21	EXECUTE THE CURRENT COMMAND FILE		EX PROC	<- >
22	CHANGE DESCRIPTION	- PROCEDURE	CD PROC	<name>
23		- CONFIGURATION	CD CFG	<name>
24	REMOVE ALL DATA	- INPUT & OUTPUT	RM CFG	
25	TYPE THE CURRENT COMMAND LOG		TY LOG	<- >

SCREEN B5

STATUS

WORKSPACE: T$ REF_CONFIG: UTIL_IN_USE: PROCEDURE BUILD
APPLIC PROG: REF TEMPLATE: REF PROFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

PROCEDURE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION	- PROCEDURE	RD PROC	<name>
2 - TYPE	- COMMAND LOG	TY LOG	<name>
3 -	- PROCEDURE	TY PROC	<name>
4 - CLEAR LOG OF OLD INFORMATION		CL	
5 - ACTIVATE	- PROCEDURE	ACT PROC	<name>
6 - COPY	- PROCEDURE	CP PROC	<f,to>
7 - DEFINE A NEW PROCEDURE		N PROC	
8 - EDIT AN EXISTING PROCEDURE		ED PROC	<name>
9 - EDIT A LOG TO BUILD A PROCEDURE		ED LOG	<name>
10 - SAVE A TEMPORARY PROCEDURE		SA PROC	<new >
11 - CHANGE THE PROCEDURE DESCRIPTION		CD PROC	
12 - REMOVE THE PROCEDURE		RM PROC	
13 - TYPE THE CURRENT COMMAND LOG		TY LOG	<- >

STATUS

WORKSPACE: T$ REF_CONFIG: UTIL_IN_USE: TEMPLATE BUILDI
APPLIC_PROG: REF_TEMPLATE: REF_PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

TEMPLATE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION OF TEMPLATE		RD TPL	<name>
2 - ACTIVATE	- INPUT TEMPLATE	ACT ITPL	
3 -	- OUTPUT TEMPLATE	ACT OTPL	
4 -	- ALT TEMPLATE	ACT TPL	<name>
5 - COPY	- TEMPLATE	CP TPL	<f,to>
6 - TEMPLATE BUILDING UTILITY		ACT UTL	<TBLD>
7 - EDIT THE ACTIVE TEMPLATE		ED TPL	<- >
8 - CHANGE THE TEMPLATE DESCRIPTION		CD TPL	
9 - REMOVE TEMPLATE FROM UFD		RM TPL	<name>

SCREEN B7

STATUS

```
WORKSPACE:    T$                REF_CONFIG:
UTIL_IN_USE:  WORKSPACE CONTR   APPLIC_PROG:
REF_TEMPLATE:                   REF_PROFILE:
```

WORKSPACE CONTROL

			COMMAND FORMAT	
1	READ DESCRIPTION	- WORKSPACE	RD WS	<name>
2		- CONFIGURATION	RD CFG	<name>
3		- TEMPLATE	RD TPL	<name>
4		- APPL. PROG.	RD APPL	<name>
5		- PROCEDURE	RD PROC	<name>
6	CLEAR LOG OF OLD INFORMATION		CL	
7	TYPE	- COMMAND LOG	TY LOG	<name>
8		- PROCEDURE	TY PROC	<name>
9	NEW	- WORKSPACE	N WS	
10		- CONFIGURATION	N CFG	<base>
11	COPY	- WORKSPACE	CP WS	<f,to>
12		- PROCEDURE	CP PROC	<f,to>
13	ACTIVATE	- WORKSPACE	ACT WS	<name>
14		- CONFIGURATION	ACT CFG	<name>
15		- TEMPLATE	ACT TPL	<name>
16		- APPL. PROG.	ACT APPL	<name>
17		- UTILITY	ACT UTL	<menu>
18		- INPUT TEMPL	ACT ITPL	
19		- OUTPUT TEMPL	ACT OTPL	
20		- PROCEDURE	ACT PROC	<name>
21		- PROGRAM UFD	ACT PUFD	<path>
22	SAVE TEMPORARY	- WORKSPACE	SA WS	<name>
23		- PROCEDURE	SA PROC	<name>
24	REMOVE FROM UFD	- WORKSPACE	RM WS	<name>
25		- CONFIGURATION	RM CFG	<name>
26		- TEMPLATE	RM TPL	<name>
27		- PROCEDURE	RM PROC	<name>
28	SET USER LOGIN CHARACTERISTICS		SLOG	

SCREEN B8

STATUS

```
WORKSPACE:     T$                REF_CONFIG:
UTIL_IN_USE:   WORKSPACE CONTR   APPLIC_PROG:
REF_TEMPLATE:                    REF_PROCFILE:
```

SCREEN B9

MASTER CONFIGURATIONS

BLKDB

MASTER TEMPLATES

CONF.IN
CONF.OUT
FOOD.IN
FOOD.OUT
ATMO.IN
ATMO.OUT
CO2R.IN
CO2R.OUT
WATM.IN
WATM.OUT
CONT.IN
CONT.OUT
ELEC.IN
ELEC.OUT
REAC.IN
REAC.OUT
STAC.IN
STAC.OUT
STRU.IN
STRU.OUT
THER.IN
THER.OUT
WAST.IN
WAST.OUT
CREW.IN
CREW.OUT
CADP.IN
CADP.OUT
COMP.OUT
COMP.IN
OUTP.IN
OUTP.OUT

USER CONFIGURATIONS

SSCDM1
NUDB
NEWDB

USER WORKSPACES

JLS1
NEW

USER TEMPLATES

A
AA
E.IN

:

MASTER APPLICATION PROGRAMS

CONFIG
FOOD
ATMOS
CO2REM
CONTAM
ELECT
REACT
STACON
STRUCT
THERM
WASTE
WATMAN
CADP
CREWEQ
COMPAT
OUTPUT

USER PROCEDURES

SPSTA
EX
IEM

EX4
EX2
EX3

STATUS

WORKSPACE: T8 REF_CONFIG: UTIL_IN_USE: WORKSPACE CONTR
APPLIC PROG: REF TEMPLATE: REF PROFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

WORKSPACE CONTROL

			COMMAND FORMAT	
1 -	READ DESCRIPTION	- WORKSPACE	RD WS	<name>
2 -		- CONFIGURATION	RD CFG	<name>
3 -		- TEMPLATE	RD TPL	<name>
4 -		- APPL. PROG.	RD APPL	<name>
5 -		- PROCEDURE	RD PROC	<name>
6 -	CLEAR LOG OF OLD INFORMATION		CL	
7 -	TYPE	- COMMAND LOG	TY LOG	<name>
8 -		- PROCEDURE	TY PROC	<name>
9 -	NEW	- WORKSPACE	N WS	
10 -		- CONFIGURATION	N CFG	<base>
11 -	COPY	- WORKSPACE	CP WS	<f,to>
12 -		- PROCEDURE	CP PROC	<f,to>
13 -	ACTIVATE	- WORKSPACE	ACT WS	<name>
14 -		- CONFIGURATION	ACT CFG	<name>
15 -		- TEMPLATE	ACT TPL	<name>
16 -		- APPL. PROG.	ACT APPL	<name>
17 -		- UTILITY	ACT UTL	<menu>
18 -		- INPUT TEMPL	ACT ITPL	
19 -		- OUTPUT TEMPL	ACT OTPL	
20 -		- PROCEDURE	ACT PROC	<name>
21 -		- PROGRAM UFD	ACT PUFD	<path>
22 -	SAVE TEMPORARY	- WORKSPACE	SA WS	<name>
23 -		- PROCEDURE	SA PROC	<name>
24 -	REMOVE FROM UFD	- WORKSPACE	RM WS	<name>
25 -		- CONFIGURATION	RM CFG	<name>
26 -		- TEMPLATE	RM TPL	<name>
27 -		- PROCEDURE	RM PROC	<name>
28 -	SET USER LOGIN CHARACTERISTICS		SLOG	

:14
ENTER CFG TO BE ACTIVATED : SSCDM1

STATUS

WORKSPACE: T$ REF_CONFIG: SSCDM1 UTIL_IN_USE: WORKSPACE CONTR
APPLIC_PROG: REF_TEMPLATE: REF_PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

WORKSPACE CONTROL

			COMMAND FORMAT	
1	- READ DESCRIPTION	- WORKSPACE	RD WS	<name>
2	-	- CONFIGURATION	RD CFG	<name>
3	-	- TEMPLATE	RD TPL	<name>
4	-	- APPL. PROG.	RD APPL	<name>
5	-	- PROCEDURE	RD PROC	<name>
6	- CLEAR LOG OF OLD INFORMATION		CL	
7	- TYPE	- COMMAND LOG	TY LOG	<name>
8	-	- PROCEDURE	TY PROC	<name>
9	- NEW	- WORKSPACE	N WS	
10	-	- CONFIGURATION	N CFG	<base>
11	- COPY	- WORKSPACE	CP WS	<f,to>
12	-	- PROCEDURE	CP PROC	<f,to>
13	- ACTIVATE	- WORKSPACE	ACT WS	<name>
14	-	- CONFIGURATION	ACT CFG	<name>
15	-	- TEMPLATE	ACT TPL	<name>
16	-	- APPL. PROG.	ACT APPL	<name>
17	-	- UTILITY	ACT UTL	<menu>
18	-	- INPUT TEMPL	ACT ITPL	
19	-	- OUTPUT TEMPL	ACT OTPL	
20	-	- PROCEDURE	ACT PROC	<name>
21	-	- PROGRAM UFD	ACT PUFD	<path>
22	- SAVE TEMPORARY	- WORKSPACE	SA WS	<name>
23	-	- PROCEDURE	SA PROC	<name>
24	- REMOVE FROM UFD	- WORKSPACE	RM WS	<name>
25	-	- CONFIGURATION	RM CFG	<name>
26	-	- TEMPLATE	RM TPL	<name>
27	-	- PROCEDURE	RM PROC	<name>
28	- SET USER LOGIN CHARACTERISTICS		SLOG	

:16
** NO DEFAULT ALLOWED FOR ACTIVATE APPLICATION **
ENTER APPLICATION PROGRAM NAME :ATMOS

STATUS

WORKSPACE: T$ REF_CONFIG: SSCDM1 UTIL_IN_USE: APPLICATION EXE
APPLIC_PROG: ATMOS. REF_TEMPLATE: REF_PROFILE:

PERMANENT MENU

```
H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
    RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu
```

APPLICATION EXECUTION

			COMMAND FORMAT	
1 -	READ DISCRIPTION	- APPL. PROG.	RD APPL	<name>
2 -		- CONFIGURATION	RD CFG	<name>
3 -		- TEMPLATE	RD TPL	<name>
4 -	NEW	- CONFIGURATION	N CFG	<base>
5 -	ACTIVATE	- APPL. PROG.	ACT APPL	<name>
6 -		- CONFIGURATION	ACT CFG	<name>
7 -	ACTIVATE DB VIEW AS	- INPUT TEMPL.	ACT ITPL	
8 -		- OUTPUT TEMPL.	ACT OTPL	
9 -		- ALT. TEMPL.	ACT TPL	<name>
10 -	COPY	- CONFIGURATION	CP CFG	<f,to>
11 -	REVIEW DB WITH ACTIVE TEMPLATE		RVU	
12 -	REVIEW	- INPUT DB	RVU IDB	
13 -		- OUTPUT DB	RVU ODB	
14 -	PRINT DB WITH ACTIVE TEMPLATE		PRVU	
15 -	PRINT	- INPUT DB	PRVU IDB	
16 -		- OUTPTU DB	PRVU ODB	
17 -	EXECUTE CURRENT APPLICATION PROGRAM		EX APPL	<- >
18 -	CHANGE DESCRIPTION	- CONFIGURATION	CD CFG	<name>
19 -	REMOVE ALL DATA	- INPUT & OUTPUT	RM CFG	
20 -	TYPE CURRENT COMMAND LOG		TY LOG	<- >

:RVU IDB

L#	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	6.00000	6.00000	1, 1, 1	A	AIRLOCK LENGTH	FT
2	6.00000	6.00000	1, 1, 2			
3	6.00000	6.00000	1, 1, 3			
4	6.00000	6.00000	1, 1, 4			
5	6.00000	6.00000	1, 1, 5			
6	6.00000	6.00000	1, 1, 6			
7	6.00000	6.00000	1, 1, 7			
8	6.00000	6.00000	1, 1, 8			
9	6.00000	6.00000	1, 1, 9			
10	6.00000	6.00000	1, 1, 10			
11	6.00000	6.00000	1, 1, 11			
12	6.00000	6.00000	1, 1, 12			
13	8.00000	8.00000		C	STATION CREW SIZE	-
14	0.350000E-04	0.350000E-04		CAPKI	CRYOGENIC TANK INSULATION CONDUCTIVITY	BTU/HRFTDR
15	0.347000E-05	0.347000E-05		CAPKM1	STAINLESS STEEL MATERIAL CONSTANT	-
16	0.210000E-01	0.210000E-01		CAPKW1	CRYO. ATMOS. STORAGE SYSTEM CONTR.POW.	KW
17	0.970000	0.970000		CAPM	METABOLIC WATER PRODUCTION	LB/MAN/DAY
18	26.0000	26.0000	1, 1, 1	CAPN	AIRLOCK OPERATIONS RESUPPLY INTERVAL	-
19	26.0000	26.0000	1, 1, 2			
20	26.0000	26.0000	1, 1, 3			

Nn - Next page, R - Reprint page, n - modify L# n, Ln - n L# per page, E - End and savemods, Q - Quit, H - Help, PA - Print all

EDIT
)

SCREEN B15

STATUS

WORKSPACE: T$ REF_CONFIG: SSCDM1 UTIL_IN_USE: APPLICATION EXE
APPLIC PROG: AIMOS REF_TEMPLATE:
REF_PROFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

APPLICATION EXECUTION

		COMMAND FORMAT	
1 - READ DISCRIPTION	- APPL. PROG.	RD APPL	<name>
2 -	- CONFIGURATION	RD CFG	<name>
3 -	- TEMPLATE	RD TPL	<name>
4 - NEW	- CONFIGURATION	N CFG	<base>
5 - ACTIVATE	- APPL. PROG.	ACT APPL	<name>
6 -	- CONFIGURATION	ACT CFG	<name>
7 - ACTIVATE DB VIEW AS-	INPUT TEMPL.	ACT ITPL	
8 -	- OUTPUT TEMPL.	ACT OTPL	
9 -	- ALT. TEMPL.	ACT TPL	<name>
10 - COPY	- CONFIGURATION	CP CFG	<f,to>
11 - REVIEW DB WITH ACTIVE TEMPLATE		RVU	
12 - REVIEW	- INPUT DB	RVU IDB	
13 -	- OUTPUT DB	RVU ODB	
14 - PRINT DB WITH ACTIVE TEMPLATE		PRVU	
15 - PRINT	- INPUT DB	PRVU IDB	
16 -	- OUTPTU DB	PRVU ODB	
17 - EXECUTE CURRENT APPLICATION PROGRAM		EX APPL	<- >
18 - CHANGE DESCRIPTION -	CONFIGURATION	CD CFG	<name>
19 - REMOVE ALL DATA -	INPUT & OUTPUT	RM CFG	
20 - TYPE CURRENT COMMAND LOG		TY LOG	<- >

117

L#	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	58.0050	58.0050		CAPL		
2	325.931	325.931		CAPW1		
3	573.566	573.566		CAPW2		
4	0.000000	0.000000		CAPW3		
5	0.000000	0.000000		CAPW4		
6	0.000000	0.000000		CAPW5		
7	0.000000	0.000000		CAPW6		
8	0.173496	0.173496		CHI0		
9	0.968305E-01	0.968305E-01		CHIN		
10	0.856002E-01	0.856002E-01		DELTR0		
11	0.342182	0.342182		DELTRN		
12	0.218211E-01	0.218211E-01		FKWAS		
13	0.000000	0.000000		FKWEA		
14	18.6119	18.6119		FL1		
15	2.28545	2.28545		FL2		
16	3.62146	3.62146		FL3		
17	0.000000	0.000000		FL5		
18	0.000000	0.000000		FL6		
19	25.0188	25.0188		FLAS		
20	2251.70	2251.70		FLAST		

Nn - Next page, R - Reprint page, n - modify L# n, Ln - n L# per page, E - End and savemods, Q - Quit, H - Help, PA - Print all

EDIT
)

L#	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
21	0.000000	0.000000		FLEA		
22	0.000000	0.000000		FLEAT		
23	0.000000	0.000000		FLP4		
24	0.000000	0.000000		FNE		
25	0.000000	0.000000		FNT	NUMBER OF PROPELLANT TANK PAIRS	DAYS
26	2	2		IX		
27	7.10000	7.10000		PA		
28	0.000000	0.000000		QA		
29	71.3711	71.3711		QA0		
30	5.72806	5.72806		QAN		
31	-77.0991	-77.0991		QAS		
32	0.000000	0.000000		QCF		
33	0.000000	0.000000		QEA		
34	0.000000	0.000000		QFC		
35	0.000000	0.000000		QPC		
36	0.000000	0.000000		QSP		
37	1.46009	1.46009		R0		
38	0.814901	0.814901		RN		
39	0.000000	0.000000		RW		
40	62.0000	62.0000		SUM		

Nn - Next page, R - Reprint page, n - modify L# n, Ln - n L# per page, E - End and saveaods, Q - Quit, H - Help, PA - Print all

EDIT
>

SCREEN B18

```
* WS ACTIVATED ON : 21 Nov 86 12:55:24 Friday
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>ACT UTL  WSC
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  DATA
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  APEX
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  PREX
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  PBLD
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  TBLD
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  WSC
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>D
* DIRECTORY
>I
* INVENTORY
>ACT CFG SSCDM1
* ACTIVATING CONFIGURATION DATABASE *
>ACT APPL ATMOS
>R
* RETURN TO THE PREVIOUS MENU
>ACT UTL  APEX
>RVU IDB
>EX APPL ATMOS
*    EXECUTING ATMOS
>RVU ODB
>TY LOG T$

*
```

```
.5
.NULL.
T
T
T
ACT UTL  USC
Q
ACT UTL  DATA
Q
ACT UTL  APEX
Q
ACT UTL  PREX
Q
ACT UTL  PBLD
Q
ACT UTL  TBLD
Q
ACT UTL  USC
T
T
T
D
I
ACT CFG SSCDM1
ACT APPL ATMOS
R
ACT UTL  APEX
RVU IDB
EX APPL ATMOS
RVU ODB
TY LOG T$
R
ACT UTL PBLD
ED LOG T$
BOTTOM
```

SCREEN B20

```
* US ACTIVATED ON : 21 Nov 86 12:55:24 Friday
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>ACT UTL  USC
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  DATA
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  APEX
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  PREX
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  PBLD
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  TBLD
>Q
* QUIT THIS MENU, RETURN TO MAIN
>ACT UTL  USC
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>T
* TOGGLE TTY PRINT MODE
>D
* DIRECTORY
>I
* INVENTORY
>ACT CFG SSCDM1
* ACTIVATING CONFIGURATION DATABASE *
>ACT APPL ATMOS
>R
* RETURN TO THE PREVIOUS MENU
>ACT UTL  APEX
>RVU IDB
>EX APPL ATMOS
*    EXECUTING ATMOS
>RVU ODB
>TY LOG T$
>R
* RETURN TO THE PREVIOUS MENU
>ACT UTL PBLD

                    ***  FOR MORE INFORMATION, <CR>  ***
                    ***       OR TO QUIT      ,   Q    ***
```

```
>ED LOG T$
>TY LOG T$
```

SCREEN B22

STATUS

WORKSPACE: T$ REF_CONFIG: SSCDM1 UTIL_IN_USE: PROCEDURE BUILD
APPLIC_PROG: ATMOS REF_TEMPLATE: REF_PROFILE: T$

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
(CR) - Clear the screen and relist the menu

PROCEDURE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION	- PROCEDURE	RD PROC	(name)
2 - TYPE	- COMMAND LOG	TY LOG	(name)
3 -	- PROCEDURE	TY PROC	(name)
4 - CLEAR LOG OF OLD INFORMATION		CL	
5 - ACTIVATE	- PROCEDURE	ACT PROC	(name)
6 - COPY	- PROCEDURE	CP PROC	(f,to)
7 - DEFINE A NEW PROCEDURE		N PROC	
8 - EDIT AN EXISTING PROCEDURE		ED PROC	(name)
9 - EDIT A LOG TO BUILD A PROCEDURE		ED LOG	(name)
10 - SAVE A TEMPORARY PROCEDURE		SA PROC	(new)
11 - CHANGE THE PROCEDURE DESCRIPTION		CD PROC	
12 - REMOVE THE PROCEDURE		RM PROC	
13 - TYPE THE CURRENT COMMAND LOG		TY LOG	(-)

```
:L
 TEMPORARY PROCEDURE EXISTS AT LOGOUT
DO YOU WISH TO SAVE (Y/N):Y

INPUT PROCEDURE NAME: NEWPROC
SAVE CURRENT WORKSPACE STATUS (Y/N):Y

 WORKSPACE HAS ONLY TEMP NAME
INPUT WORKSPACE NAME: EXAMPLEWS
* LOGOUT - BYE *
OK,
```

MASTER CONFIGURATIONS

BLKDB

MASTER TEMPLATES

CONF.IN
CONF.OUT
FOOD.IN
FOOD.OUT
ATMO.IN
ATMO.OUT
CO2R.IN
CO2R.OUT
WATM.IN
WATM.OUT
CONT.IN
CONT.OUT
ELEC.IN
ELEC.OUT
REAC.IN
REAC.OUT
STAC.IN
STAC.OUT
STRU.IN
STRU.OUT
THER.IN
THER.OUT
WAST.IN
WAST.OUT
CREW.IN
CREW.OUT
CADP.IN
CADP.OUT
COMP.OUT
COMP.IN
OUTP.IN
OUTP.OUT

USER CONFIGURATIONS

SSCDM1
NUDB
NEWDB

USER WORKSPACES

JLS1
NEW
EXAMPLEWS

USER TEMPLATES

A
AA
E.IN

1

SCREEN B24

STATUS

WORKSPACE: T$ REF_CONFIG: UTIL_IN_USE: WORKSPACE CONTR
APPLIC_PROG: REF_TEMPLATE: REF_PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
(CR) - Clear the screen and relist the menu

:D

WORKSPACE CONTROL

			COMMAND FORMAT	
1	READ DESCRIPTION	- WORKSPACE	RD WS	(name)
2		- CONFIGURATION	RD CFG	(name)
3		- TEMPLATE	RD TPL	(name)
4		- APPL. PROG.	RD APPL	(name)
5		- PROCEDURE	RD PROC	(name)
6	CLEAR LOG OF OLD INFORMATION		CL	
7	TYPE	- COMMAND LOG	TY LOG	(name)
8		- PROCEDURE	TY PROC	(name)
9	NEW	- WORKSPACE	N WS	
10		- CONFIGURATION	N CFG	(base)
11	COPY	- WORKSPACE	CP WS	(f,to)
12		- PROCEDURE	CP PROC	(f,to)
13	ACTIVATE	- WORKSPACE	ACT WS	(name)
14		- CONFIGURATION	ACT CFG	(name)
15		- TEMPLATE	ACT TPL	(name)
16		- APPL. PROG.	ACT APPL	(name)
17		- UTILITY	ACT UTL	(menu)
18		- INPUT TEMPL	ACT ITPL	
19		- OUTPUT TEMPL	ACT OTPL	
20		- PROCEDURE	ACT PROC	(name)
21		- PROGRAM UFD	ACT PUFD	(path)
22	SAVE TEMPORARY	- WORKSPACE	SA WS	(name)
23		- PROCEDURE	SA PROC	(name)
24	REMOVE FROM UFD	- WORKSPACE	RM WS	(name)
25		- CONFIGURATION	RM CFG	(name)
26		- TEMPLATE	RM TPL	(name)
27		- PROCEDURE	RM PROC	(name)
28	SET USER LOGIN CHARACTERISTICS		SLOG	

STATUS

WORKSPACE: T$ REF_CONFIG: UTIL_IN_USE: WORKSPACE CONTR
APPLIC_PROG: REF_TEMPLATE:
REF_PROCFILE:

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

WORKSPACE CONTROL

			COMMAND FORMAT	
1	- READ DESCRIPTION	- WORKSPACE	RD WS	<name>
2	-	- CONFIGURATION	RD CFG	<name>
3	-	- TEMPLATE	RD TPL	<name>
4	-	- APPL. PROG.	RD APPL	<name>
5	-	- PROCEDURE	RD PROC	<name>
6	- CLEAR LOG OF OLD INFORMATION		CL	
7	- TYPE	- COMMAND LOG	TY LOG	<name>
8	-	- PROCEDURE	TY PROC	<name>
9	- NEW	- WORKSPACE	N WS	
10	-	- CONFIGURATION	N CFG	<base>
11	- COPY	- WORKSPACE	CP WS	<f,to>
12	-	- PROCEDURE	CP PROC	<f,to>
13	- ACTIVATE	- WORKSPACE	ACT WS	<name>
14	-	- CONFIGURATION	ACT CFG	<name>
15	-	- TEMPLATE	ACT TPL	<name>
16	-	- APPL. PROG.	ACT APPL	<name>
17	-	- UTILITY	ACT UTL	<menu>
18	-	- INPUT TEMPL	ACT ITPL	
19	-	- OUTPUT TEMPL	ACT OTPL	
20	-	- PROCEDURE	ACT PROC	<name>
21	-	- PROGRAM UFD	ACT PUFD	<path>
22	- SAVE TEMPORARY	- WORKSPACE	SA WS	<name>
23	-	- PROCEDURE	SA PROC	<name>
24	- REMOVE FROM UFD	- WORKSPACE	RM WS	<name>
25	-	- CONFIGURATION	RM CFG	<name>
26	-	- TEMPLATE	RM TPL	<name>
27	-	- PROCEDURE	RM PROC	<name>
28	- SET USER LOGIN CHARACTERISTICS		SLOG	

:13

ENTER FILE TO BE ACTIVATED :EXAMPLEWS

STATUS

WORKSPACE:	EXAMPLEWS	REF_CONFIG:	SSCDM1	UTIL_IN_USE:	PROCEDURE BUILD
APPLIC_PROG:	ATMOS	REF_TEMPLATE:		REF_PROCFILE:	NEWPROC

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

PROCEDURE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION	- PROCEDURE	RD PROC	<name>
2 - TYPE	- COMMAND LOG	TY LOG	<name>
3 -	- PROCEDURE	TY PROC	<name>
4 - CLEAR LOG OF OLD INFORMATION		CL	
5 - ACTIVATE	- PROCEDURE	ACT PROC	<name>
6 - COPY	- PROCEDURE	CP PROC	<f,to>
7 - DEFINE A NEW PROCEDURE		N PROC	
8 - EDIT AN EXISTING PROCEDURE		ED PROC	<name>
9 - EDIT A LOG TO BUILD A PROCEDURE		ED LOG	<name>
10 - SAVE A TEMPORARY PROCEDURE		SA PROC	<new >
11 - CHANGE THE PROCEDURE DESCRIPTION		CD PROC	
12 - REMOVE THE PROCEDURE		RM PROC	
13 - TYPE THE CURRENT COMMAND LOG		TY LOG	<- >

:

STATUS

WORKSPACE: EXAMPLEUS REF_CONFIG: SSCDM1 UTIL_IN_USE: PROCEDURE BUILD
APPLIC_PROG: ATMOS REF_TEMPLATE: REF_PROCFILE: NEWPROC

PERMANENT MENU

```
H - HELP
D - DIRECTORY of CFG's, TPL's and US's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
    RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu
```

:EX PROC

PROCEDURE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION	- PROCEDURE	RD PROC	<name>
2 - TYPE	- COMMAND LOG	TY LOG	<name>
3 -	- PROCEDURE	TY PROC	<name>
4 - CLEAR LOG OF OLD INFORMATION		CL	
5 - ACTIVATE	- PROCEDURE	ACT PROC	<name>
6 - COPY	- PROCEDURE	CP PROC	<f,to>
7 - DEFINE A NEW PROCEDURE		N PROC	
8 - EDIT AN EXISTING PROCEDURE		ED PROC	<name>
9 - EDIT A LOG TO BUILD A PROCEDURE		ED LOG	<name>
10 - SAVE A TEMPORARY PROCEDURE		SA PROC	<new >
11 - CHANGE THE PROCEDURE DESCRIPTION		CD PROC	
12 - REMOVE THE PROCEDURE		RM PROC	
13 - TYPE THE CURRENT COMMAND LOG		TY LOG	<- >

SCREEN B28

```
>ED LOG T$
>TY LOG T$
>L
* LOGOUT
>SA PROC NEWPROC
>SA WS EXAMPLEWS
* WS ACTIVATED ON : 21 Nov 86 13:21:16 Friday
>ACT APPL ATMOS
>ACT CFG SSCDM1
* ACTIVATING CONFIGURATION DATABASE *
>TY LOG EXAMPLEWS
>EX PROC NEWPROC
* BEGIN PROCEDURE EXECUTION ON NEWPROC
>RVU IDB
>EX APPL ATMOS
*    EXECUTING ATMOS
>RVU ODB
* EXITING PROCEDURE NEWPROC.PROC
* SAVED CURRENT STATUS OF ATTACHED WORKSPACE
>SA WS EXAMPLEWS
>TY LOG EXAMPLEWS
```

STATUS

WORKSPACE: EXAMPLEWS REF_CONFIG: SSCDM1 UTIL_IN_USE: TEMPLATE BUILDI
APPLIC_PROG: ATMOS REF_TEMPLATE: REF_PROFILE: NEWPROC

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
0 - CANCEL the current command
<CR> - Clear the screen and relist the menu

TEMPLATE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION OF TEMPLATE		RD TPL	<name>
2 - ACTIVATE	- INPUT TEMPLATE	ACT ITPL	
3 -	- OUTPUT TEMPLATE	ACT OTPL	
4 -	- ALT TEMPLATE	ACT TPL	<name>
5 - COPY	- TEMPLATE	CP TPL	<f,to>
6 - TEMPLATE BUILDING UTILITY		ACT UTL	<TBLD>
7 - EDIT THE ACTIVE TEMPLATE		ED TPL	<- >
8 - CHANGE THE TEMPLATE DESCRIPTION		CD TPL	
9 - REMOVE TEMPLATE FROM UFD		RM TPL	<name>

:5
ENTER SOURCE TEMPLATE :OUTP.IN
ENTER DESTINATION TEMPLATE :NUOUT

SCREEN B30

STATUS

WORKSPACE:	EXAMPLEWS	REF_CONFIG:	SSCDM1	UTIL_IN_USE:	TEMPLATE BUILDI
APPLIC PROG:	ATMOS	REF TEMPLATE:	NUOUT	REF PROFILE:	NEWPROC

PERMANENT MENU

H - HELP
D - DIRECTORY of CFG's, TPL's and WS's
I - INVENTORY of APPL's and PROC's
S - SYSTEM COMMANDS for PRIMOS
C - Add a COMMENT to the command log
T - TOGGLE the MENU PRINT setting
R - RETURN to the PREVIOUS MENU
Q - QUIT this sequence of menus and
RETURN to the MAIN MENU
L - LOGOUT
@ - CANCEL the current command
<CR> - Clear the screen and relist the menu

TEMPLATE BUILDING

		COMMAND FORMAT	
1 - READ DESCRIPTION OF TEMPLATE		RD TPL	<name>
2 - ACTIVATE	- INPUT TEMPLATE	ACT ITPL	
3 -	- OUTPUT TEMPLATE	ACT OTPL	
4 -	- ALT TEMPLATE	ACT TPL	<name>
5 - COPY	- TEMPLATE	CP TPL	<f,to>
6 - TEMPLATE BUILDING UTILITY		ACT UTL	<TBLD>
7 - EDIT THE ACTIVE TEMPLATE		ED TPL	<- >
8 - CHANGE THE TEMPLATE DESCRIPTION		CD TPL	
9 - REMOVE TEMPLATE FROM UFD		RM TPL	<name>

:7

REFERENCE NUMBER	SAVED FOR NEW TEMPLATE	PARAMETER NAME	DESCRIPTION	UNITS
1		A4	EXPERIMENT MOUNTING RACK WIDTH	FT
2		AC		
3		ACC		
4		ACP		
5		ACPC		
6		ACS1		
7		ACS2		
8		ACS3		
9		ACS4		
10		AE0		
11		AEFA		
12		ALPHAR	RATIO OF NOSE CONE LENGTH TO DIAMETER	-
13		ALT	ORBITAL ALTITUDE	NMI
14		AMPC		
15		AP		
16		ASA		
17		B4	EXPERIMENT WORK TABLE WIDTH	FT
18		BWDP		
19		BWPDP		
20		C	STATION CREW SIZE	-
21		CAPLC		
22		CAPLCP		
23		CAPLM		
24		CAPLMT		
25		CAPVHT		

S n - save n, F n - free n, N n - next n pages, T - top, B - bottom
R - reprint page, E - exit edit and save, Q - quit edit - no save
>S 12 14 16 18 20 21 22 23 24 25

SCREEN B32

REFERENCE NUMBER	SAVED FOR NEW TEMPLATE	PARAMETER NAME	DESCRIPTION	UNITS
1		A4	EXPERIMENT MOUNTING RACK WIDTH	FT
2		AC		
3		ACC		
4		ACP		
5		ACPC		
6	.	ACS1		
7		ACS2		
8		ACS3		
9		ACS4		
10	.	AE0		
11		AEFA		
12	X	ALPHAR	RATIO OF NOSE CONE LENGTH TO DIAMETER	-
13		ALT	ORBITAL ALTITUDE	NMI
14	X	AMPC		
15		AP		
16	X	ASA		
17		B4	EXPERIMENT WORK TABLE WIDTH	FT
18	X	BWDP		
19		BWPDP		
20	X	C	STATION CREW SIZE	-
21	X	CAPLC		
22	X	CAPLCP		
23	X	CAPLM		
24	X	CAPLMT		
25	X	CAPVHT		

S n - save n, F n - free n, N n - next n pages, T - top, B - bottom
R - reprint page, E - exit edit and save, Q - quit edit - no save
>

REFERENCE NUMBER	SAVED FOR NEW TEMPLATE	PARAMETER NAME	DESCRIPTION	UNITS
201	X	WSPP		
202		WST		
203		WTC		
204	.	WTWAS		
205		WWD		
206		WWL		
207		WWM		
208	.	WWP		
209		WWR		
210	X	WWR1		
211		WWRT		
212		WWRX		
213	X	WZC		

S n - save n, F n - free n, N n - next n pages, T - top, B - bottom
R - reprint page, E - exit edit and save, Q - quit edit - no save
)

L#	PRESENT VALUE	OLD VALUE	SUBSCRIPT	PNAME	DESCRIPTION	UNITS
1	1.50000	1.50000		ALPHAR	RATIO OF NOSE CONE LENGTH TO DIAMETER	-
2	50.0000	50.0000		AMPC		
3	39.9600	39.9600		ASA		
4	1137.09	1137.09		BWDP		
5	8.00000	8.00000		C	STATION CREW SIZE	-
6	4.22858	4.22858		CAPLC		
7	17.6899	17.6899		CAPLCP		
8	50.0000	50.0000		CAPLM		
9	50.0000	50.0000		CAPLMT		
10	0.000000	0.000000		CAPVHT		
11	53329.6	53329.6		WSPP		
12	50.0000	50.0000		WWR1		
13	0.000000	0.000000		WZC		

Nn - Next page, R - Reprint page, n - modify L# n, Ln - n L# per page, E - End and savemods, Q - Quit, H - Help, PA - Print all

EDIT
)

SCREEN B35