INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 34 - Application Interface Unit Test Plan

General Electric Company
Production Resources Consulting
One River Road
Schenectady, New York 12345

November 1985
Approved for public release; distribution is unlimited.

MATERIALS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AFB, OH 45433-6533
This unit test plan establishes the methodology and procedures used to adequately test the capabilities of the computer program identified as the Application Interface (AI). The AI consists of a set of callable execution time routines that allows an application program to send/receive formatted screens to/from various terminals and to perform terminal control functions independent of the terminal type. The AI translates the application's call for form processing into a message which is then sent to the User Interface Monitor of the Form Processor.

The computer software contained herein are theoretical and/or references that in no way reflect Air Force-owned or -developed computer software.
11. Title

Integrated Information Support System (IISS)
Vol VII - User Interface Subsystem
Part 34 - Application Interface Unit Test Plan

A S D 86 0046
9 Jan 1986
This unit test plan covers the work performed under Air Force Contract F33615-80-C-5155 (ICAM Project 6201). This contract is sponsored by the Materials Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Gerald C. Shumaker, ICAM Program Manager, Manufacturing Technology Division, through Project Manager, Mr. David Judson. The Prime Contractor was Production Resources Consulting of the General Electric Company, Schenectady, New York, under the direction of Mr. Allan Rubenstein. The General Electric Project Manager was Mr. Myron Hurlbut of Industrial Automation Systems Department, Albany, New York.

Certain work aimed at improving Test Bed Technology has been performed by other contracts with Project 6201 performing integrating functions. This work consisted of enhancements to Test Bed software and establishment and operation of Test Bed hardware and communications for developers and other users. Documentation relating to the Test Bed from all of these contractors and projects have been integrated under Project 6201 for publication and treatment as an integrated set of documents. The particular contributors to each document are noted on the Report Documentation Page (LD1473). A listing and description of the entire project documentation system and how they are related is contained in document FTR620100001, Project Overview.

The subcontractors and their contributing activities were as follows:

**TASK 4.2**

<table>
<thead>
<tr>
<th>Subcontractors</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>Reviewer</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>Responsible for IDEF support, state-of-the-art literature search</td>
</tr>
<tr>
<td>General Dynamics/ Ft. Worth</td>
<td>Responsible for factory view function and information models</td>
</tr>
<tr>
<td>Subcontractors</td>
<td>Role</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Illinois Institute of Technology</td>
<td>Responsible for factory view function research (IITRI) and information models of small and medium-size business</td>
</tr>
<tr>
<td>North American Rockwell</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Northrop Corporation</td>
<td>Responsible for factory view function and information models</td>
</tr>
<tr>
<td>Pritsker and Associates</td>
<td>Responsible for IDEF2 support</td>
</tr>
<tr>
<td>SofTech</td>
<td>Responsible for IDEF0 support</td>
</tr>
</tbody>
</table>

**TASKS 4.3 - 4.9 (TEST BED)**

<table>
<thead>
<tr>
<th>Subcontractors</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>Responsible for consultation on applications of the technology and on IBM computer technology.</td>
</tr>
<tr>
<td>Computer Technology Associates (CTA)</td>
<td>Assisted in the areas of communications systems, system design and integration methodology, and design of the Network Transaction Manager.</td>
</tr>
<tr>
<td>Control Data Corporation (CDC)</td>
<td>Responsible for the Common Data Model (CDM) implementation and part of the CDM design (shared with DACOM).</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>Responsible for the overall CDM Subsystem design integration and test plan, as well as part of the design of the CDM (shared with CDC). DACOM also developed the Integration Methodology and did the schema mappings for the Application Subsystems.</td>
</tr>
</tbody>
</table>
**Subcontractors** | **Role**
---|---
Digital Equipment Corporation (DEC) | Consulting and support of the performance testing and on DEC software and computer systems operation.
McDonnell Douglas Automation Company (McAuto) | Responsible for the support and enhancements to the Network Transaction Manager Subsystem during 1984/1985 period.
On-Line Software International (OSI) | Responsible for programming the Communications Subsystem on the IBM and for consulting on the IBM.
Rath and Strong Systems Products (RSSP) (In 1985 became McCormack & Dodge) | Responsible for assistance in the implementation and use of the MRP II package (PIOS) that they supplied.
SofTech, Inc. | Responsible for the design and implementation of the Network Transaction Manager (NTM) in 1981/1984 period.
Software Performance Engineering (SPE) | Responsible for directing the work on performance evaluation and analysis.
Structural Dynamics Research Corporation (SDRC) | Responsible for the User Interface and Virtual Terminal Interface Subsystems.

Subcontractors and other prime contractors under other projects who have contributed to Test Bed Technology, their contributing activities and responsible projects are as follows:  

**Subcontractors** | **Role**
---|---
General Dynamics/ Ft. Worth | Responsible for factory view
<table>
<thead>
<tr>
<th>Contractors</th>
<th>ICAM Project</th>
<th>Contributing Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>1701, 2201, 2202</td>
<td>Enhancements for IBM node use. Technology Transfer to Integrated Sheet Metal Center (ISMC)</td>
</tr>
<tr>
<td>Control Data Corporation (CDC)</td>
<td>1502, 1701</td>
<td>IISS enhancements to Common Data Model Processor (CDMP)</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>1502</td>
<td>IISS enhancements to Integration Methodology</td>
</tr>
<tr>
<td>General Electric</td>
<td>1502</td>
<td>Operation of the Test Bed and communications equipment.</td>
</tr>
<tr>
<td>Hughes Aircraft Company (HAC)</td>
<td>1701</td>
<td>Test Bed enhancements</td>
</tr>
<tr>
<td>Structural Dynamics Research Corporation (SDRC)</td>
<td>1502, 1701, 1703</td>
<td>IISS enhancements to User Interface/Virtual Terminal Interface (UI/VTI)</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS

SECTION 1.0 GENERAL ............................................... 1-1
  1.1 Purpose .................................................... 1-1
  1.2 Project References ......................................... 1-1
  1.3 Terms and Abbreviations .................................... 1-2

SECTION 2.0 DEVELOPMENT ACTIVITY ............................... 2-1
  2.1 Statement of Pretest Activity ............................... 2-1
  2.2 Pretest Activity Results .................................... 2-1

SECTION 3.0 SYSTEM DESCRIPTION ................................. 3-1
  3.1 System Description .......................................... 3-1
  3.2 Testing Schedule ............................................ 3-3
  3.3 First Location Testing ....................................... 3-3
  3.4 Subsequent Location Testing .................................. 3-4

SECTION 4.0 SPECIFICATIONS AND EVALUATIONS ..................... 4-1
  4.1 Test Specifications .......................................... 4-1
  4.2 Test Methods and Constraints ................................ 4-5
  4.3 Test Progression ............................................... 4-6
  4.4 Test Evaluation ............................................... 4-6

SECTION 5.0 TEST PROCEDURES ................................... 5-1
  5.1 Test Description ............................................. 5-1
  5.2 Test Control .................................................. 5-1
  5.3 Test Procedures .............................................. 5-1

APPENDICES

  A Commands for ARTEST ........................................... A-1
  B FORMS FOR TESTING THE FORM PROCESSOR ...................... B-1

FIGURES

  3-1 Interface Block Diagram ..................................... 3-2
  4-1 Table of Functionality Testing ............................. 4-3
  5-1a BEFORE .................................................... 5-3
  5-1b AFTER ..................................................... 5-4
  5-1c BEFORE .................................................... 5-5
  5-2a BEFORE .................................................... 5-6
  5-2b AFTER ..................................................... 5-7
  5-3a BEFORE .................................................... 5-8
<table>
<thead>
<tr>
<th>Section No.</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-27a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-27b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-27c</td>
<td>AFTER</td>
<td>5-27b PF key</td>
</tr>
<tr>
<td>5-28a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-28b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-29a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-29b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-30a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-30b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-31a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-31b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-32a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-32b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-33a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-33b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-34a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-34b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-35a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-35b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-36a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-36b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-37a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-37b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-38a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-38b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-39a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-39b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-39c</td>
<td>AFTER</td>
<td>5-39b</td>
</tr>
<tr>
<td>5-40a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-40b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-40c</td>
<td>AFTER</td>
<td>5-40b</td>
</tr>
<tr>
<td>5-40d</td>
<td>AFTER</td>
<td>5-40c</td>
</tr>
<tr>
<td>5-41a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-41b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-42a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-42b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-43a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-43b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-44a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-44b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-45a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-45b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-46a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-46b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-47a</td>
<td>BEFORE</td>
<td></td>
</tr>
<tr>
<td>5-47b</td>
<td>AFTER</td>
<td></td>
</tr>
<tr>
<td>5-48a</td>
<td>BEFORE</td>
<td></td>
</tr>
</tbody>
</table>
SECTION 1

GENERAL

1.1 Purpose

This unit test plan establishes the methodology and procedures used to adequately test the capabilities of the computer program identified as the Application Interface known in this document as the AI. The AI is one configuration item of the Integrated Information Support System (IISS) User Interface (UI). It consists of Application Interface callable routines.

1.2 Project References


1.3 Terms and Abbreviations

**Application Interface:** (AI), subset of the IISS User Interface that consists of the callable routines that are linked with applications that use the Form Processor or Virtual Terminal. The AI enables applications to be hosted on computers other than the host of the User Interface.

**Application Process:** (AP), a cohesive unit of software that can be initiated as a unit to perform some function or functions.

**Communication Services:** allows on host interprocess communication and inter-host communication between the various Test Bed subsystems.

**Communication Subsystem:** (COMM), IISS subsystem that provides communication services to the Test Bed and subsystems.

**Computer Program Configuration Item:** (CPCI), an aggregation of computer programs or any of their discrete portions, which satisfies an end-use function.

**Field:** two-dimensional space on a terminal screen.

**Field Pointer:** indicates the ITEM which contains the current cursor position.

**Form:** structured view which may be imposed on windows or other forms. A form is composed of fields. These fields may be defined as forms, items, and windows.

**Form Definition:** (FD), forms definition language after compilation. It is read at runtime by the Form Processor.
Forms Definition Language: (FDL), the language in which electronic forms are defined.

Form Hierarchy: a graphic representation of the way in which forms, items and windows are related to their parent form.

Form Processor: (FP), subset of the IISS User Interface that consists of a set of callable execution time routines available to an application program for form processing.

IISS Function Screen: the first screen that is displayed after logon. It allows the user to specify the function he wants to access and the device type and device name on which he is working.

Integrated Information Support System: (IISS), a test computing environment used to investigate, demonstrate and test the concepts of information management and information integration in the context of Aerospace Manufacturing. The IISS addresses the problems of integration of data resident on heterogeneous data bases supported by heterogeneous computers interconnected via a Local Area Network.

Message: descriptive text which may be returned in the standard message line on the terminal screen. They are used to warn of errors or provide other user information.

Message Line: a line on the terminal screen that is used to display messages.

Network Transaction Manager: (NTM). IISS subsystem that performs the coordination, communication and housekeeping functions required to integrate the Application Processes and System Services resident on the various hosts into a cohesive system.

Qualified Name: the name of a form, item or window preceded by the hierarchy path so that it is uniquely identified.

User Data: data which is either input by the user or output by the application programs to items.

User Interface: (UI). IISS subsystem that controls the user's terminal and interfaces with the rest of the system. The UI consists of two major subsystems: the User Interface Development System (UIDS) and the User Interface Management System (UIMS).
**User Interface Management System:** (UIMS), the runtime UI. It consists of the Form Processor, Virtual Terminal, Application Interface, the User Interface Services and the Text Editor.

**User Interface/Virtual Terminal Interface:** (UI/VTI), another name for the User Interface.

**Virtual Terminal:** (VT), subset of the IISS User Interface that performs the interfacing between different terminals and the UI. This is done by defining a specific set of terminal features and protocols which must be supported by the UI software which constitutes the virtual terminal definition. Specific terminals are then mapped against the virtual terminal software by specific software modules written for each type of real terminal supported.

**Window:** dynamic area of a terminal screen on which predefined forms may be placed at run time.
SECTION 2

DEVELOPMENT ACTIVITY

2.1 Statement of Pretest Activity

During system development, the computer programs were tested progressively. Functionality was incrementally tested, and as bugs were discovered by this testing, the software was corrected.

Each Application Interface callable routine was tested individually through Application Interface development. A test program, ARTEST, was developed as an easy means of testing changes to the Application Interface. This test program allows a developer to type in commands that are translated into the appropriate Application Interface calls. With this test program all Application Interface callable routines may be executed.

All pretesting activity was conducted by the individual program developer in a manual mode. The developer would manually enter data onto the screen and observe the results. Any errors were noted by the developer, and corrections to the Application Interface software were then made after a testing session.

2.2 Pretest Activity Results

The pretest activity was very successful in the elimination of programming bugs so that at release time only a few bugs were found in the Application Interface. The development of the test program, ARTEST, has proved very beneficial since as new functionality was added to the Application Interface, ARTEST was also updated to test this functionality. ARTEST is the major test tool for the Unit Test Plan of the Application Interface.
3.1 **System Description**

The Application Interface consists of a set of callable execution time routines that allows an application program to send/receive formatted screens to/from various terminals and to perform terminal control functions independent of the terminal type. The Application Interface translates the application's call for form processing into a message which is then sent to the User Interface Monitor of the Form Processor.

The following block diagram illustrates the Application Interface Test Configuration used in the Unit Test Plan.
Figure 3-1 Interface Block Diagram

The required input and the resulting output of these tests are documented in detail in Section 5.3. The general testing method is the entry of commands on the ARTEST form Command Line item and the translation of this command by the ARTEST program into a call to the appropriate Application Interface routine. Each Application Interface routine as found in the Application Interface Development Specification is to be exercised. The resulting output is observed on the ARTEST form. Appendix A
outlines the ARTEST command format and the various types of commands and function keys.

The following keys are used to move within forms (using the VT100 terminal as an example): the ENTER key is used to activate all commands; the TAB key is used to move from field to field within the form; and the arrow keys are used to move within fields. In addition, ESC TAB is a reverse TAB.

3.2 Testing Schedule

The execution of the Application is dependent upon the NTM subsystem of the IISS. Testing of the Application Interface must be done only after the NTM has been successfully tested. In this unit test, the Application Interface is dependent on the Form Processor (FP) and on the Virtual Terminal (VT). In fact, all three Configuration Items are to be tested together.

3.3 First Location Testing

These tests of the Application Interface require the following:

Equipment: Air Force VAX, terminals supported by the Virtual Terminal as listed in the UI Terminal Operator's Guide.


Personnel: One integrator familiar with the IISS.

Training: The AI User manual has been previously provided with the current release.

Deliverables: The Application Interface component of the IISS UI/VTI.

Test Materials: This test may be run interactively by inputting the appropriate data and observing the output as outlined in this test plan. A script file has been created to run this unit test plan and save the resulting output.
Security considerations: None.

3.4 Subsequent Location Testing

The requirements as listed above need to be met; however, in subsequent testing it is advantageous to create a script file of the outlined tests and run this saving the output of the test for future comparisons.
4.1 Test Specification

The Unit Test Plan is based on covering specific functionality as outlined in the Application Interface DS. The test uses the test program ARTEST.

The following chart has the functional requirements as outlined in the Application Interface Development Specification listed vertically and the test activities in the Unit Test Plan that demonstrate each functional requirement’s testing listed horizontally. As can be seen in the figures in Section 5.3, the command line of the form used by ARTEST, has the actual Application Interface routine name specified or is annotated with what function key was pressed.
<table>
<thead>
<tr>
<th>Functional Requirements</th>
<th>Diagram Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>ABCDEFGHIJKLMNOPQRSTUVWXYZ</td>
</tr>
<tr>
<td>B</td>
<td>X</td>
</tr>
<tr>
<td>C</td>
<td>CHGLDV X</td>
</tr>
<tr>
<td>D</td>
<td>CLSFRM X</td>
</tr>
<tr>
<td>E</td>
<td>CLSLDV X</td>
</tr>
<tr>
<td>F</td>
<td>GDATA X</td>
</tr>
<tr>
<td>G</td>
<td>GETATT X</td>
</tr>
<tr>
<td>H</td>
<td>GETBAK X</td>
</tr>
<tr>
<td>I</td>
<td>GETCUR X</td>
</tr>
<tr>
<td>J</td>
<td>GPAGE X</td>
</tr>
<tr>
<td>K</td>
<td>GWINDO X</td>
</tr>
<tr>
<td>L</td>
<td>INITFP X</td>
</tr>
<tr>
<td>M</td>
<td>INQLDV X</td>
</tr>
<tr>
<td>N</td>
<td>OISCR X</td>
</tr>
<tr>
<td>O</td>
<td>OPNFRM X</td>
</tr>
<tr>
<td>P</td>
<td>OPNLDV X</td>
</tr>
<tr>
<td>Q</td>
<td>OUTSCR X</td>
</tr>
<tr>
<td>R</td>
<td>PARFQN X</td>
</tr>
<tr>
<td>S</td>
<td>PDATA X</td>
</tr>
<tr>
<td>T</td>
<td>PMSGLC X</td>
</tr>
<tr>
<td>U</td>
<td>PMSGLS X</td>
</tr>
<tr>
<td>V</td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td></td>
</tr>
</tbody>
</table>

-CONTINUED-
### Functional Requirements

<table>
<thead>
<tr>
<th>Functional Requirements</th>
<th>Diagram Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A B C D E F G H I J K L M N O P Q R S T U V W X Y Z</td>
</tr>
<tr>
<td>PUTATT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td>PUTBAK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td>PUTCUR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td>RMVPAG</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td>RPLFRM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td>TERMFP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

Figure 4-1 Table of Functionality Testing
The steps outlined in Section 5.3 presenting the BEFORE and AFTER forms of each test show the direct correspondence between the test and the functional requirements as listed in this section.

A - Figure 5-3-a thru Figure 5-4-b
B - Figure 5-43-a thru Figure 5-43-b
C - Figure 5-11-a thru Figure 5-12-b
D - Figure 5-69-a thru Figure 5-69-b
E - Figure 5-16-a thru Figure 5-18-b
F - Figure 5-19-a thru Figure 5-19-b
G - Figure 5-23-a thru Figure 5-23-b
H - Figure 5-27-a thru Figure 5-27-b
I - Figure 5-7-a thru Figure 5-7-b
J - Figure 5-8-a thru Figure 5-8-b
K - Figure 5-2-a thru Figure 5-2-b
L - Figure 5-41-a thru Figure 5-41-b
M - Figure 5-29-a thru Figure 5-29-b
N - Figure 5-4-a thru Figure 5-4-b
O - Figure 5-42-a thru Figure 5-42-b
P - Figure 5-29-a thru Figure 5-29-c
Q - Figure 5-28-a thru Figure 5-28-b
R - Figure 5-13-a thru Figure 5-13-b
S - Figure 5-39-c
T - Figure 5-40-c
U - Figure 5-20-a thru Figure 5-21-b, 5-24-a thru 5-25-b
V - Figure 5-22-a thru Figure 5-22-b, 5-26-a thru 5-26-b
W - Figure 5-27-a thru Figure 5-27-b
X - Figure 5-6-a thru Figure 5-6-b
Y - Figure 5-5-a thru Figure 5-5-b
Z - Figure 5-70-a thru the last figure

4.2 Testing Methods and Constraints

The tests as outlined in Section 5 must be followed. The required input is stated for each test. This testing uses the normal mode of operation of these functions and does not completely exercise all the error combinations that a user of the Application Interface might create by faulty entry of parameter information. Much of this testing has been done, however, through the normal testing done by the developer of these functions. No data recording is required. It is suggested that on further running of this test, scripting of the test may be done and the output from running the script be saved for future testing. No additional constraints are placed on this unit test besides those listed in Section 3.3 of this unit test plan.
4.3 Test Progression

The progression of testing of the Application Interface is fully outlined in Section 5 of this unit test plan. This progression should be followed exactly to insure the successful testing of this IISS configuration item.

4.4 Test Evaluation

The test results are evaluated by comparing the information returned on the various output screens to that specified as successful for the given test. As outlined in Section 5, each test of Application Interface functionality provides an input screen with the required data entry specified and the resulting output for a successful test. To speed up this testing and provide more accurate measurement of the test's success, scripting has been used. The resulting output of these tests is saved in a file FPUTP.SAV. The corresponding test script file is FPUTP.SCP. Both these files are under IISS Configuration Management. If scripting is used, these files should be copied over to the test directory. The .SAV file may be used for future comparison against subsequent running of this unit test using scripting. To compare the results use the command file DIFFILE.COM which was released as part of the acceptance testing done on the Air Force VAX and is under Configuration Management. The only differences should be the date/time stamps on the IISS function screen and the type of device on the window manager screen. The device type is given to the UIS by the NTM at runtime. Note that the test script used to test the Application Interface is exactly the same one as was used to test the Form Processor. The latter part of that script which tests window management processing is not necessary for the testing of the Application Interface.
SECTION 5

TEST PROCEDURES

5.1 Test Description

A general description of this unit test was provided in Section 3.

5.2 Test Control

As outlined, this unit test may be done manually or run automatically using a supplied script file. To manually perform this unit test would require the tester to be logged into the IISS system and enter SDARTESTZZ on the Function Select Form. In section 5.3 the required input data is specified for each function being tested and the resulting successful output is also specified. The order of the testing is also completely specified. The test control information is completely described by the sequence of the input and output screens presented in this section. The successfulness of the test may be determined by doing a comparison on the .SAV files produced against the ones provided under IISS Configuration Management.

5.3 Test Procedures

To run the unit test plan as outlined in this section on a VAX, one must be logged on to an IISS account. The NTM must be up and running and the UI group logical names IISSFLIB and IISSMLIB must be set properly. IISSFLIB points to the directory containing production form definitions (FD files). IISSMLIB points to the directory containing error messages (.MSG files).

This unit test uses the program ARTEST and its associated forms ffl through ff9. The fd1 source file for these forms is presented in Appendix B. The executable for ARTEST should exist in the NTM environment directory and the NTM dirtbl.dat should have its SD entry pointing to this directory. The NTM tables APITBL, APITBL, and ACTTBL should have ARTEST set up as a normal IISS application program.

Assuming the NTM is up and running, an IISS user may start up this unit test plan as follows:

5-1
$ SET DEF <to directory containing your NTM environment>
$ VT100 -RFPUTP.SCP -SFPTST.SAV

This starts up the VT100 device driver with a source script as input and specifies a save file for output. If the User Interface system has been installed at your site with a different device driver, then this step is amended as appropriate. The test begins executing on the terminal. The results of this test are saved in the current directory in the file FPTST.SAV. The Before and After Figures show not only the form input and output but also the sequencing of the test.
Figure 5-1a (BEFORE)
**Figure 5-1b (AFTER)**
Figure 5-1c (BEFORE)
Figure 5-2a (BEFORE)
Figure 5-2b (AFTER)
Figure 5-3a (BEFORE)
Display: ..........................................................................................

application

Figure 5-3b (AFTER)
**Figure 5-4a (BEFORE)**
Figure 5-4b (AFTER)
Figure 5-5a (BEFORE)
Figure 5-5b (AFTER)
Figure 5-6a (BEFORE)
Figure 5-6b  (AFTER)
Figure 5-7a  (BEFORE)
Figure 5-7b (AFTER)
Figure 5-8a (BEFORE)
Figure 5-8b (AFTER)
Figure 5-9a (BEFORE)
Figure 5-9b (AFTER)
Figure 5-10a (BEFORE)
Figure 5-10b (AFTER)
Figure 5-11a (BEFORE)
Figure 5-11b (AFTER)
Figure 5-12a (BEFORE)
Figure 5-12b (AFTER)
Figure 5-13a (BEFORE)
Figure 5-13b (AFTER)
Figure 5-14a (BEFORE)
Figure 5-14b (AFTER)
Figure 5-15a (BEFORE)
Figure 5-15b  (AFTER)
Figure 5-16a (BEFORE)
Figure 5-16b (AFTER)
Figure 5-17a (BEFORE)
Figure 5-17b (AFTER)
Figure 5-18a (BEFORE)
Figure 5-18b (AFTER)
Figure 5-19a (BEFORE)
Figure 5-19b (AFTER)
Figure 5-20a (BEFORE)
Figure 5-20b (AFTER)
Figure 5-21a (BEFORE)
Figure 5-22a (BEFORE)
Figure 5-22b (AFTER)
Figure 5-23a (BEFORE)
Figure 5-23b (AFTER)
Figure 5-24a (BEFORE)
Figure 5-24b (AFTER)
Figure 5-25a (BEFORE)
Figure 5-25b (AFTER)
Figure 5-26a (BEFORE)
Figure 5-26b (AFTER)
Figure 5-27a (BEFORE)
Figure 5-27b (AFTER)
Display: type = F, row = 1, col = 1, fn0 = SCREEN.SCREEN1, FFL, FFS

Figure 5-27c (AFTER 5-27b PF key)
Figure 5-28a (BEFORE)
Figure 5-28b (AFTER)
Figure 5-29a (BEFORE)
Figure 5-29b (AFTER)
Figure 5-30b (AFTER)
Figure 5-31a (BEFORE)
Figure 5-31b (AFTER)
Figure 5-32a (BEFORE)
Figure 5-32b (AFTER)
Figure 5-33b (AFTER)
Figure 5-34a (BEFORE)
Figure 5-34b (AFTER)
Figure 5-35a (BEFORE)
Figure 5-35b (AFTER)
Figure 5-36a (BEFORE)
Command Line

Display: ........................................................................................................

Mag: 0

Figure 5-36b (AFTER)
Figure 5-37a (BEFORE)
Figure 5-37b (AFTER)
Figure 5-38a (BEFORE)
Figure 5-38b (AFTER)
Figure 5-39a (BEFORE)
Figure 5-39b (AFTER)
Figure 5-39c  (AFTER 5-39b)
Figure 5-40a (BEFORE)
Figure 5-40b (AFTER)
Figure 5-40c  (AFTER 5-40b)
Figure 5-40d (AFTER 5-40c)
Figure 5-41a (BEFORE)
Figure 5-41b (AFTER)
Figure 5-42a (BEFORE)
Display: Opened and changed to logical device: 25 ........................
................................................................................
................................................................................

Figure 5-42b (AFTER)
Display: Opened and changed to logical device: 25

Figure 5-43a (BEFORE)
Figure 5-43b (AFTER)
Figure 5-44a (BEFORE)
Display: Opened and changed to logical device: 25 ..........................

Figure 5-44b (AFTER)
Figure 5-45a (BEFORE)
Figure 5-45b (AFTER)
Figure 5-46a (BEFORE)
Figure 5-46b (AFTER)
Figure 5-47a (BEFORE)
Figure 5-47b (AFTER)
Figure 5-48a (BEFORE)
Figure 5-48b (AFTER)
Figure 5-49a (BEFORE)
Figure 5-49b (AFTER)
Figure 5-50a (BEFORE)
Figure 5-50b (AFTER)
Figure 5-51a (BEFORE)
Figure 5-52a (BEFORE)
Figure 5-52b (AFTER)
Figure 5-53a (BEFORE)
Figure 5-53b (AFTER)
Figure 5-54a (BEFORE)
Figure 5-54b (AFTER)
Figure 5-55a (BEFORE)
Figure 5-55b (AFTER)
Figure 5-56a (BEFORE)
Figure 5-56b (AFTER)
Figure 5-57a (BEFORE)
### Window Manager

<table>
<thead>
<tr>
<th>Application</th>
<th>Type</th>
<th>Name</th>
<th>Pri</th>
<th>Window</th>
<th>Location</th>
<th>Display</th>
<th>Viewport</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDARTESTZZ</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>SCREEN</td>
<td>1 1 23</td>
<td>0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 60 10</td>
<td>0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td>VT100</td>
<td>TT:</td>
<td>2</td>
<td>SCREEN</td>
<td>1 1 24</td>
<td>0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 60 10</td>
<td>0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 60 10</td>
<td>0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>MENU</td>
<td>VT100</td>
<td>TT:</td>
<td>3</td>
<td>SCREEN</td>
<td>1 1 24</td>
<td>0 0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 5-57b (AFTER)*

5-121
<table>
<thead>
<tr>
<th>Application</th>
<th>Type</th>
<th>Name</th>
<th>Pri</th>
<th>Window</th>
<th>Location</th>
<th>Display</th>
<th>Viewport</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDARTESTZZ</td>
<td>BDDPRINTZZ</td>
<td>PRINT.DEV</td>
<td>1</td>
<td>SCREEN</td>
<td>6</td>
<td>17</td>
<td>34 11</td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>80 23</td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>60</td>
<td>10 8</td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td>VT100</td>
<td>TT:</td>
<td>2</td>
<td>SCREEN</td>
<td>1</td>
<td>1</td>
<td>80 24</td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>80 23</td>
</tr>
<tr>
<td>SDARTESTZZ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>60</td>
<td>10 8</td>
</tr>
<tr>
<td>MENU</td>
<td>VT100</td>
<td>TT:</td>
<td>3</td>
<td>SCREEN</td>
<td>1</td>
<td>1</td>
<td>80 24</td>
</tr>
<tr>
<td>MENU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>80 23</td>
</tr>
</tbody>
</table>

Figure 5-58a (BEFORE)
Figure 5-58b (AFTER)
Figure 5-59a (BEFORE)
Figure 5-59b (AFTER)
Figure 5-60a (BEFORE)
Figure 5-60b (AFTER)
Figure 5-61a (BEFORE)
Figure 5-61b (AFTER)
Figure 5-62a (BEFORE)
Figure 5-62b (AFTER)
Figure 5-63a (BEFORE)
**Figure 5-63b (AFTER)**

<table>
<thead>
<tr>
<th>Application</th>
<th>Type</th>
<th>Name</th>
<th>Pri</th>
<th>Location</th>
<th>Display</th>
<th>Viewport</th>
</tr>
</thead>
<tbody>
<tr>
<td>STARTESTZ2</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>6 17 36 11</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>STARTESTZ2</td>
<td>SCREEN</td>
<td></td>
<td></td>
<td>1 1 80 23</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>STARTESTZ2</td>
<td>K3</td>
<td></td>
<td></td>
<td>2 60 10 8</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>STARTESTZ2</td>
<td>VT100</td>
<td>TT:</td>
<td>2</td>
<td>1 1 80 24</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>STARTESTZ2</td>
<td>SCREEN</td>
<td></td>
<td></td>
<td>1 1 80 23</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>STARTESTZ2</td>
<td>K3</td>
<td></td>
<td></td>
<td>2 60 10 8</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>STARTESTZ2</td>
<td>K3</td>
<td></td>
<td></td>
<td>2 60 10 8</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>MENU</td>
<td>VT100</td>
<td>TT:</td>
<td>3</td>
<td>1 1 80 24</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>MENU</td>
<td>SCREEN</td>
<td></td>
<td></td>
<td>1 1 80 23</td>
<td>0 0</td>
<td></td>
</tr>
</tbody>
</table>

Reg: 0
## Window Manager

<table>
<thead>
<tr>
<th>Application</th>
<th>Type</th>
<th>Name</th>
<th>Pri</th>
<th>Location</th>
<th>Display</th>
<th>Viewport</th>
<th>Offset</th>
<th>Row Col</th>
<th>W</th>
<th>H</th>
<th>D</th>
<th>Raw Col</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>2</td>
<td>6</td>
<td>36</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>1</td>
<td>80</td>
<td>23</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>2</td>
<td>60</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>1</td>
<td>80</td>
<td>24</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>1</td>
<td>80</td>
<td>23</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>2</td>
<td>60</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>2</td>
<td>60</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>1</td>
<td>80</td>
<td>24</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SDARTEST22</td>
<td>VT100</td>
<td>TT:</td>
<td>1</td>
<td>1</td>
<td>80</td>
<td>23</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 5-64a (BEFORE)
Figure 5-64b (AFTER)
Figure 5-65a (BEFORE)
Figure 5-65b (AFTER)
Figure 5-66a (BEFORE)
### Window Manager

<table>
<thead>
<tr>
<th>Application</th>
<th>Type</th>
<th>Name</th>
<th>Pri</th>
<th>Row Col</th>
<th>W</th>
<th>D</th>
<th>Row Col</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDARTESTZ2</td>
<td>VT100</td>
<td>TT: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDARTESTZ2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDARTESTZ2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDARTESTZ2</td>
<td>VT100</td>
<td>TT: 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDARTESTZ2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDARTESTZ2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MENU</td>
<td>VT100</td>
<td>TT: 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MENU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5-66b (AFTER)**
### Window Manager

<table>
<thead>
<tr>
<th>Device</th>
<th>Application</th>
<th>Type</th>
<th>Name</th>
<th>Pri</th>
<th>Window Location</th>
<th>Display Size</th>
<th>Viewport Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>VT100</td>
<td>TT: 2</td>
<td>2</td>
<td>1 1 80 24 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>IVT100</td>
<td>TT: 2</td>
<td>1</td>
<td>1 1 80 23 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>M3</td>
<td></td>
<td>1</td>
<td>2 60 10 8 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>M3</td>
<td></td>
<td>1</td>
<td>2 60 10 8 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>VT100</td>
<td>TT: 1</td>
<td>1</td>
<td>6 17 36 11 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>SCREEN</td>
<td></td>
<td>1</td>
<td>1 1 80 23 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>M3</td>
<td></td>
<td>1</td>
<td>2 60 10 8 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDARTESTZZ</td>
<td>M3</td>
<td></td>
<td>1</td>
<td>1 1 80 24 0 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MENU</td>
<td>VT100</td>
<td>TT: 3</td>
<td>3</td>
<td>1 1 80 23 0 0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5-67a (BEFORE)
Figure 5-67b (AFTER)
Figure 5-68a (BEFORE)
Figure 5-68b (AFTER)
Figure 5-69a (BEFORE)
Figure 5-69b (AFTER)
Figure 5-70a (BEFORE)
Figure 5-70b (AFTER)
Figure 5-71a (BEFORE)
Now at Vax Command level

Figure 5-71b (AFTER)
APPENDIX A

COMMANDS FOR ARTEST

Commands are of the form:
command arg1 arg2 ... argn

Where command is the form processor procedure implementing the command and arg1, etc. are the input arguments. Arguments are separated by blank(s) and arguments which contain blanks are enclosed in double quotes.

Window/Form Manipulation
--------------------------
add form to a window
addfrm window_path form_name
delete pages from window
rmvpag window_path page_number
replace page in window
rplfrm window_path page_number form_name
close form
clsfrm form_name

Window/Form Information
-------------------------
get name of form on page
n gpage window_path page_number
get number of pages
in window
gwindo window_path

Attributes
----------
change attributes
putatt path dur attribute
get attributes
getatt path dur
put and get temp attributes
tmpatt path dur attribute
change background attributes
putbak path dur attribute
get background attributes
getbak path dur
get and get background
tmpbak path dur attribute
dur is 0) permanent
1) temporary
attributes are:
INPUT, OUTPUT, TEXT, HIDDEN, ERROR
background attributes are:
WHITE, BLACK, XPARNT
Data Manipulation

---

put data to form/item/array  pdata path data
get data from form/item/array  gdata instance path
instance is 0=previous, 1=current

Miscellaneous

---

put cursor to field  putcur path
window set(term within term)  oiscr window path
parse fully qualified name  parfqn level
   with level: 1=first, 2=second,...
   0=last, 1=next to last....
Must use pf16(0) first!!!

Function Keys

---

pf0(enter) - do command on command line
pf16(0)  - display path name of cursor position
pf1  - go to next form processor mode
pf2  - help
pf3  - display message screen
pf4  - quit

for scrolling, press pf1 until the mode is scrll/page,
then:

pf5(7)  - horizontal scroll forward
pf6(8)  - horizontal scroll backward
pf7(9)  - vertical scroll forward
pf8(-)  - vertical scroll backward
pf9(4)  - horizontal page forward
pf10(5) - horizontal page backward
pf11(6) - vertical page forward
pf12(. ) - vertical page backward
APPENDIX B

FORMS FOR TESTING THE FORM PROCESSOR

CREATE FORM ffl
PROMPT AT 1 2 "Command Line"
  item 10
  at 1 70
  size 8
  display as text
  value "form ffl"

  item 13
  at 21 11
  size 8
  display as text
  value "Display:"

  item 14
  at 1 20
  size 40
  display as input
  help pathcom

WINDOW w1 (2 v 1)
  AT 2 2
  SIZE 10 BY 4
  display as white

WINDOW w2 (2 v 1, 2 H 1)
  AT 2 15
  SIZE 10 BY 4
  display as xparnt

WINDOW w3 at 2 60
  size 10 by 8
  display as black
  form ff2 (2 h 4)
  at 12 1
  size 12 by 4

  form ff3
  at 12 32
  size 10 by 4
form ff4 (2 v 1)
at 16 2
size 10 by 2

form ff5
at 12 43
size 10 by 4

form ff6
at 12 55
size 10 by 4

form ff7
at 12 68
size 10 by 4

form ff8
at 16 13
size 10 by 4

item fqn
at 21 20
size 60 by 3
display as output

CREATE FORM ff2
prompt 1 2 "form ff2"

item 11 (2 h 1)
at 2 2
size 2
display as input

item 12
at 3 2
size 3
display as input

create form ff3
prompt 1 2 "form ff3"
item 11 (2 v 1, 2 h 1)
at 2 2
size 3
display as input
create form ff4
background black
prompt 1 2 "form ff4"

item il
at 2 2
size 4
display as input

create form ff5
background white
prompt 1 2 "form ff5"

item il
at 3 3
size 1 by 2
display as input

create form ff6
background xparnt
prompt 1 2 "form ff6"

item il
at 3 6
size 1 by 4
display as input

create form ff7
prompt 1 2 "form ff7"
item il((3 6 v 0. 2/4 h 1))
at 2 2
size 1
display as input

create form ff8
prompt 1 2 "form ff8"

item il
at 2
size 4
display as input

create form ff9
prompt 1 2 "form ff9"
item 2 2
size 8
display as input

window w4
at 3 1
size 10 by 5
display as black

create form pathcom
prompt 1 9 "Commands for ARTEST"
prompt 2 9 "-------- ---------
prompt 3 9 "add form to a window addrfm window form"
prompt 4 9 "delete pages from window rmvpag window page"
prompt 5 9 "replace page in window rplfrm window page form"
prompt 6 9 "close form clsfrm form"
prompt 7 9 "put data to form item array pdata path data"
prompt 8 9 "get data from form item array gdata
inst(0-prev,1-cur) path"
prompt 9 9 "change attributes: foreground putatt path
dur(prm-0.tmp-1) attrib"
prompt 10 9 "get attributes: foreground getatt path
dur(prm-0.tmp-1)"
prompt 11 9 "put and get temp attributes(f) tmpatt path dur
attrib"
prompt 12 9 "change attributes: background putbak path
dur(prm-0.tmp-1) attrib"
prompt 13 9 "get attributes: background getbak path
dur(prm-0.tmp-1)"
prompt 14 9 "put and get temp attributes(b) tmpbak path dur
attrib"
prompt 15 9 "get name of form on page n gpage window page"
prompt 16 9 "get number of pages in window gwindo window"
prompt 17 9 "put cursor to field putcur path"
prompt 18 9 "window set(term within term) oisrc path"
prompt 19 9 "parse fully qualified name parfqn
lev(0-1st,1=fst..1=nxt21st,etc)"
prompt 20 9 "first"
prompt 21 9 "Function Keys"
prompt 22 9 "-------- ---------
prompt 23 9 "pfl6(0) - display path name of cursor position"
END
8-87
DTIC