INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 21 - Forms Driven Forms Editor Unit Test Plan

S. Barker
Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209

September 1990

Final Report for Period 1 April 1987 - 31 December 1990

Approved for Public Release; Distribution is Unlimited
NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

This technical report has been reviewed and is approved for publication.

[Signature]
DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

DATE
25 July 91

FOR THE COMMANDER:

[Signature]
BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

DATE
25 July 91

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.
THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.
This unit test plan establishes the methodology and procedures used to test the Forms Driven Form Editor computer program.

**Block 11:**

**INTEGRATED INFORMATION SUPPORT SYSTEM**

*Vol VIII - User Interface Subsystem*

*Part 21 - Forms Driven Forms Editor Unit Test Plan*
FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

<table>
<thead>
<tr>
<th>SUBCONTRACTOR</th>
<th>ROLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Data Corporation</td>
<td>Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.</td>
</tr>
<tr>
<td>D. Appleton Company</td>
<td>Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.</td>
</tr>
<tr>
<td>ONTEK</td>
<td>Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.</td>
</tr>
<tr>
<td>Simpact Corporation</td>
<td>Responsible for Communication development.</td>
</tr>
<tr>
<td>Structural Dynamics</td>
<td>Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.</td>
</tr>
<tr>
<td>Research Corporation</td>
<td>Responsible for test bed operations and support.</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>SECTION 1.0 GENERAL</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Purpose</td>
<td>1-1</td>
</tr>
<tr>
<td>1.2 Project References</td>
<td>1-1</td>
</tr>
<tr>
<td>1.3 Terms and Abbreviations</td>
<td>1-2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION 2.0 DEVELOPMENT ACTIVITY</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Statement of Pretest Activity</td>
<td>2-1</td>
</tr>
<tr>
<td>2.2 Pretest Activity Results</td>
<td>2-1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION 3.0 SYSTEM DESCRIPTION</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 System Description</td>
<td>3-1</td>
</tr>
<tr>
<td>3.2 Testing Schedule</td>
<td>3-2</td>
</tr>
<tr>
<td>3.3 First Location Testing</td>
<td>3-3</td>
</tr>
<tr>
<td>3.3.1 Test Materials on VAX</td>
<td>3-3</td>
</tr>
<tr>
<td>3.4 Subsequent Location Testing on VAX</td>
<td>3-3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION 4.0 SPECIFICATIONS AND EVALUATIONS</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Test Specification</td>
<td>4-1</td>
</tr>
<tr>
<td>4.2 Test Methods and Constraints</td>
<td>4-3</td>
</tr>
<tr>
<td>4.3 Test Progression</td>
<td>4-3</td>
</tr>
<tr>
<td>4.4 Test Evaluation</td>
<td>4-4</td>
</tr>
<tr>
<td>4.4.1 Test Evaluation on the VAX</td>
<td>4-4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION 5.0 TEST PROCEDURES</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Test Description</td>
<td>5-1</td>
</tr>
<tr>
<td>5.2 Test Control</td>
<td>5-1</td>
</tr>
<tr>
<td>5.3 Test Procedures</td>
<td>5-1</td>
</tr>
<tr>
<td>5.3.1 Keypad Function</td>
<td>5-1</td>
</tr>
<tr>
<td>5.3.1.1 Keypad on VAX</td>
<td>5-2</td>
</tr>
<tr>
<td>5.3.1.2 Keypad on IBM</td>
<td>5-2</td>
</tr>
<tr>
<td>5.3.2 Test Procedures on VAX</td>
<td>5-2</td>
</tr>
<tr>
<td>5.3.3 IBM Test Procedures</td>
<td>5-3</td>
</tr>
<tr>
<td>5.3.4 Testing the FDPE</td>
<td>5-4</td>
</tr>
</tbody>
</table>
### LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-1</td>
<td>Interface Block Diagram</td>
<td>3-2</td>
</tr>
<tr>
<td>5-1</td>
<td>Keypad for VT100</td>
<td>5-2</td>
</tr>
<tr>
<td>5-2</td>
<td>IISS Logon Screen</td>
<td>5-4</td>
</tr>
<tr>
<td>5-3</td>
<td>IISS Function Screen</td>
<td>5-5</td>
</tr>
<tr>
<td>5-4</td>
<td>Invoking the FDFE</td>
<td>5-6</td>
</tr>
<tr>
<td>5-5</td>
<td>Test Screen 1</td>
<td>5-7</td>
</tr>
<tr>
<td>5-6</td>
<td>Test Screen 2</td>
<td>5-8</td>
</tr>
<tr>
<td>5-7</td>
<td>Test Screen 3</td>
<td>5-9</td>
</tr>
<tr>
<td>5-8</td>
<td>Test Screen 4</td>
<td>5-10</td>
</tr>
<tr>
<td>5-9</td>
<td>Test Screen 5</td>
<td>5-11</td>
</tr>
<tr>
<td>5-10</td>
<td>Test Screen 6</td>
<td>5-12</td>
</tr>
<tr>
<td>5-11</td>
<td>Test Screen 7</td>
<td>5-13</td>
</tr>
<tr>
<td>5-12</td>
<td>Test Screen 8</td>
<td>5-14</td>
</tr>
<tr>
<td>5-13</td>
<td>Test Screen 9</td>
<td>5-15</td>
</tr>
<tr>
<td>5-14</td>
<td>Test Screen 10</td>
<td>5-16</td>
</tr>
<tr>
<td>5-15</td>
<td>Test Screen 11</td>
<td>5-17</td>
</tr>
<tr>
<td>5-16</td>
<td>Test Screen 12</td>
<td>5-18</td>
</tr>
<tr>
<td>5-17</td>
<td>Test Screen 13</td>
<td>5-19</td>
</tr>
<tr>
<td>5-18</td>
<td>Test Screen 14</td>
<td>5-20</td>
</tr>
<tr>
<td>5-19</td>
<td>Test Screen 15</td>
<td>5-21</td>
</tr>
<tr>
<td>5-20</td>
<td>Test Screen 16</td>
<td>5-22</td>
</tr>
<tr>
<td>5-21</td>
<td>Test Screen 17</td>
<td>5-23</td>
</tr>
<tr>
<td>5-22</td>
<td>Test Screen 18</td>
<td>5-24</td>
</tr>
<tr>
<td>5-23</td>
<td>Test Screen 19</td>
<td>5-25</td>
</tr>
<tr>
<td>5-24</td>
<td>Test Screen 20</td>
<td>5-26</td>
</tr>
<tr>
<td>5-25</td>
<td>Test Screen 21</td>
<td>5-27</td>
</tr>
<tr>
<td>5-26</td>
<td>Test Screen 22</td>
<td>5-28</td>
</tr>
<tr>
<td>5-27</td>
<td>Test Screen 23</td>
<td>5-29</td>
</tr>
<tr>
<td>5-28</td>
<td>Test Screen 24</td>
<td>5-30</td>
</tr>
<tr>
<td>5-29</td>
<td>Test Screen 25</td>
<td>5-31</td>
</tr>
<tr>
<td>5-30</td>
<td>Test Screen 26</td>
<td>5-32</td>
</tr>
<tr>
<td>5-31</td>
<td>Test Screen 27</td>
<td>5-33</td>
</tr>
<tr>
<td>5-32</td>
<td>Test Screen 28</td>
<td>5-34</td>
</tr>
<tr>
<td>5-33</td>
<td>Test Screen 29</td>
<td>5-35</td>
</tr>
<tr>
<td>5-34</td>
<td>Test Screen 30</td>
<td>5-36</td>
</tr>
<tr>
<td>5-35</td>
<td>Test Screen 31</td>
<td>5-37</td>
</tr>
<tr>
<td>5-36</td>
<td>Test Screen 32</td>
<td>5-38</td>
</tr>
<tr>
<td>5-37</td>
<td>Test Screen 33</td>
<td>5-39</td>
</tr>
<tr>
<td>5-38</td>
<td>Test Screen 34</td>
<td>5-40</td>
</tr>
<tr>
<td>5-39</td>
<td>Test Screen 35</td>
<td>5-41</td>
</tr>
<tr>
<td>5-40</td>
<td>Test Screen 36</td>
<td>5-42</td>
</tr>
<tr>
<td>5-41</td>
<td>Test Screen 37</td>
<td>5-43</td>
</tr>
<tr>
<td>5-42</td>
<td>Test Screen 38</td>
<td>5-44</td>
</tr>
<tr>
<td>5-43</td>
<td>Test Screen 39</td>
<td>5-45</td>
</tr>
<tr>
<td>5-44</td>
<td>Test Screen 40</td>
<td>5-46</td>
</tr>
<tr>
<td>5-45</td>
<td>Test Screen 41</td>
<td>5-47</td>
</tr>
<tr>
<td>5-46</td>
<td>Test Screen 42</td>
<td>5-48</td>
</tr>
<tr>
<td>5-47</td>
<td>Test Screen 43</td>
<td>5-49</td>
</tr>
<tr>
<td>5-48</td>
<td>Test Screen 44</td>
<td>5-50</td>
</tr>
<tr>
<td>5-49</td>
<td>Test Screen 45</td>
<td>5-51</td>
</tr>
<tr>
<td>5-50</td>
<td>Test Screen 46</td>
<td>5-52</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-1</td>
<td>Matrix Mapping FDFE Functions with Test Plan</td>
<td>4-2</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 Forms Driven Form Editor known in this document as the FDFE. The FDFE is one configuration item of the Integrated Information Support System (IISS) User Interface (UI).

1.2 Project References


1.3 Terms and Abbreviations

American Standard Code for Information Interchange: (ASCII), the character set defined by ANSI X3.4 and used by most computer vendors.

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.

Attribute: field characteristic such as blinking, highlighted, black, etc. and various other combinations. Background attributes are defined for forms or windows only. Foreground attributes are defined for items. Attributes may be permanent, i.e., they remain the same unless changed by the application program, or they may be temporary, i.e., they remain in effect until the window is redisplayed.

Device Drivers: (DD), software modules written to handle I/O for a specific kind of terminal. The modules map terminal specific commands and data to a neutral format. Device Drivers are part of the UI Virtual Terminal.
Display List: a list of all the open forms that are currently being processed by the FP or the user.

Extended Binary Coded Decimal Interchange Code: (EBCDIC), the character set used by a few computer vendors (notably IBM) instead of ASCII.

Field: two dimensional space on a terminal screen.

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.

Forms Driven Form Editor: (FDFE), subset of the FE which consists of a forms driven application used to create Form Definition files interactively.

Form Editor: (FE), subset of the IISS User Interface that is used to create definitions of forms. The FE consists of the Forms Driven Form Editor and the Forms Language Compiler.

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

Forms Language Compiler: (FLAN), subset of the FE that consists of a batch process that accepts a series of forms definition language statements and produces form definition files as output.

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.

Form Processor Text Editor: (FPTE), subset of the Form Processor that consists of software modules that provide text editing capabilities to all users of applications that use the Form Processor.
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.

Item: non-decomposable area of a form in which hard-coded descriptive text may be placed and the only defined areas where user data may be input/output.

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.

Open List: a list of all the forms that are currently open for an application process.

Operating System: (OS), software supplied with a computer which allows it to supervise its own operations and manage access to hardware facilities such as memory and peripherals.

Page: instance of forms in windows that are created whenever a form is added to a window.

Paging and Scrolling: a method which allows a form to contain more data than can be displayed with provisions for viewing any portion of the data buffer.

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

Subform: a form that is used within another form.

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 Development System: (UIDS), collection of IISS User Interface subsystems that are used by applications programmers as they develop IISS applications. The UIDS includes the Form Editor and the Application Generator.

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 Monitor: (UIM), part of the Form Processor that handles messaging between the NTM and the UI. It also provides authorization checks and initiates applications.

User Interface Services: (UIS), subset of the IISS User Interface that consists of a package of routines that aid users in controlling their environment. It includes message management, change password, and application definition services.

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.
Window Manager: a facility which allows the following to be manipulated: size and location of windows, the device on which an application is running, the position of a form within a window. It is part of the Form Processor.
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 form used by the FDFE was individually tested. This testing 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 program were then made after a testing session.

2.2 Pretest Activity Results

Each testing of the forms used in the FDFE application discovered a few minor bugs which were then corrected and retesting proved successful. Testing included exceptional conditions and error conditions for data entered on the forms. The overall test results during development showed no major programming errors. Only minor bugs were discovered and corrected.
SECTION 3
SYSTEM DESCRIPTION

3.1 System Description

The FDFE interfaces directly with users as an application which uses the Form Processor (FP) - via the NTM. Physical terminals are assumed to have video display, a textual keyboard, four cursor positioning keys or key sequences, a help key or key sequence, a message key, an entry key, a quit key and four other keys to be used by the FDFE for special processing (see section 5.3). The FDFE must interface with the following software tools: the Forms Processor (FP), the Forms Compiler (FLAN), C language runtime routines and forms storage management. It is used to create or modify FDL files and to create new FD files; it can also be used to delete existing FDL and FD files as well as to rename existing FDL files (see Figure 3-1).
3.2 Testing Schedule

The execution of the FDFE is dependent upon the NTM subsystem of IISS and testing of the FDFE must be done only after the NTM has been successfully tested. Within the UI subsystem, the FDFE uses the FP, VT, AI and FLAN and must be tested only after they have been successfully tested.
3.3 **First Location Testing**

These tests of the FDFE require the following:

**Equipment:** Air Force VAX, or IBM terminal supported by the VT as listed in the UI Terminal Operator Guide.

**Support Software:** The Integrated Information Support System, the Oracle database management system, and C run-time libraries.

**Personnel:** One integrator familiar with the UIS.

**Training:** FDPE manuals have been previously provided with the past release.

**Deliverables:** The FDFE subsystem of the UI.

**Security considerations:** None.

3.3.1 **Test Materials on VAX**

**Test Materials:** This test is interactive and can be manually performed as outlined in this test plan. It also could be run as a script file if so desired (see below).

3.4 **Subsequent Location Testing on VAX**

The requirements as listed above need to be met; however, in subsequent testing it may be advantageous to create a script file of the outlined tests and run this saving the output of the test for future comparisons.
SECTION 4

TEST SPECIFICATIONS AND EVALUATIONS

4.1 Test Specification

The following functionality of the FDFE is demonstrated by the test outlined in section 5:

1) Insert a Form Language Source
   A) Insert Form into Forms Language Source
      a) Layout Edit mode
      b) Single Field Edit mode
      c) Icon Edit mode
   B) Modify Form in Forms Language Source
      a) Single Field Edit mode

2) Modify a Form Language Source
   A) Insert Form into Forms Language Source
      a) Layout Edit mode
   B) Modify Form in Forms Language Source
      a) Single Field Edit mode
      b) Icon Edit mode
   D) Drop a Form from Forms Language Source
   E) List Forms in Forms Language Source
   F) Write and Compile Forms Language Source

3) Select a Form Language Source
   A) Layout Edit mode
   B) Icon Edit mode

4) Display Compiled Form Definition

5) Copy Form Language Source to New Form Language Source

6) Rename Form Language Source to New Form Language Source

7) Drop Form Language Source

8) Drop Form Language Object

9) List Form Language Sources (FDL files)

10) List Form Language Objects (FD files)

11) Exit

Table 4-1 shows the direct correspondence between the test (the steps outlined in Section 5) and the functional requirements as listed in this section. These functions directly correspond to the detailed functional requirements of the Forms Driven Form Editor Development Specification. The '.' indicates the figures which illustrate the testing of the top level functions: insert, modify or select a forms language source file. The '*' indicates the figures which illustrate the testing of specific functions.
| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W |
| 1.A.a | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1.A.b | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1.B   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1.C   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2.A   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2.B.a |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2.B.b |   |   |   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2.C   |   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2.D   |   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2.E   |   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 3.A   |   |   |   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 3.B   |   |   |   |   |   |   |   | * |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 4     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 5     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 6     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 7     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 8     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 9     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 10    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 11    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | *  |

Table 4-1 Matrix mapping FDFE functions with test plan
Key for Table 1:
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 tests the normal mode of operation of these functions and does not completely exercise all the error combinations that a user of the FDFE might create by faulty entry of form field information. These tests have been done, however, through the normal testing done by the developer of these functions. IISSULIB and IISSSLIB should point to default directory. No additional constraints are placed on this unit test besides those listed in Section 3.2 and 3.3 of this unit test plan.

4.3 Test Progression

The progression of testing of the FDFE 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.

W = last instruction
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 below in section 5, each test of FDFE functionality provides an input screen with the required data entry specified and the resulting output for a successful test. The only differences found should be the date and time stamps on the IISS Function Screen (Figure 5-3).

4.4.1 Test Evaluation on the VAX

To speed up this testing, scripting can be used. If scripting is used, the script file FDFEUTP.SCP and its released test saved output FDFEUTP.SAV (Under IISS Configuration Management) should be copied to your test directory. To execute the scripting option type "-RFDFEUTP.SCP -SFDFETST.SAV" when you activate the User Interface. For example:

$VT100 -RFDFEUTP.SCP -SFDFETST.SAV

To compare the results with those obtained by SDRC, compare FDFETST.SAV with FDFEUTP.SAV using the command file DIFFILE.COM (Under IISS Configuration Management).
SECTION 5

TEST SPECIFICATIONS AND EVALUATIONS

5.1 Test Description

This test consists of creating an FDL source file using the FDFE and then using this file to perform all functions.

5.2 Test Control

As outlined, this unit test is a manual test which may be done by anyone. The required input data for each function being tested, the resulting successful output and the order of the testing are completely specified below. The test control information is completely described in Section 4.4. Accurate observation of the resulting successful output must be made to ensure the unit test was done properly. As noted in Section 4.4 above scripting may be used instead of the manual test described below.

5.3 Test Procedures

5.3.1 Keypad Function

The test of the FDFE application consists of individually testing each function provided by the FDFE. The following keys are generally used to move within forms (using the VT100 terminal as an example): the <ENTER> key is used to activate all commands; the <QUIT> key is used to go back to previous activity without taking current action; 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. The only application defined function keys used by the FDFE are: the function key (PF 15 on a VT100) which when in layout mode is used to transfer control to layout description mode and back again; the function key (PF 12 on a VT100) which when in layout mode is used to move fields around on a form; and two function keys which when in edit field mode or layout description mode are used to go to the previous and the next field on a form (PF 16 and PF 17 respectively on a VT100).
5.3.1.1 Keypad on VAX

The function key locations are mapped 1:1 to the generic layout shown in Figure 5-1.

5.3.1.2 Keypad on IBM

The generic keypad must be mapped to the terminal that you are using. PA2 is used to shift to the PF 13 to PF24 set. The PF0 key (ENTER) is the <RETURN> key.

5.3.2 Test Procedures on VAX

To run the unit test plan in the VAX/VMS environment as outlined below, one must be logged on to an IISS account. The NTM must be up and running and the UI logical names IISSFLIB, IISSULIB, IISSSLIB and IISSMLIB must be set properly at the

---

<table>
<thead>
<tr>
<th>PF 1</th>
<th>PF 2</th>
<th>PF 3</th>
<th>PF 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODE KEY</td>
<td>HELP KEY</td>
<td>MESSAGE QUEUE KEY</td>
<td>QUIT KEY</td>
</tr>
<tr>
<td>PF 5</td>
<td>PF 6</td>
<td>PF 7</td>
<td>PF 8</td>
</tr>
<tr>
<td>PF 9</td>
<td>PF 10</td>
<td>PF 11</td>
<td>PF 12</td>
</tr>
<tr>
<td>Move fields on form in layout mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF 13</td>
<td>PF 14</td>
<td>PF 15</td>
<td></td>
</tr>
<tr>
<td>Go to dscpt. mode \ back to layout md</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF 16</td>
<td>PF 17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While in dscpt mode or edit field mode these keys used to scroll through fields on a form PF 16 = &lt; \ PF 17 = &gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5-1 Keypad Layout
group level. IISSFLIB points to the directory containing system form definitions (FD files). IISSULIB points to the directory containing the user's form definitions (FD files). IISSSSLIB points to the directory containing the user's form definition source files (FDL files). IISSMLIB points to the directory containing error and help messages (MSG files). To perform this test IISSULIB and IISSSSLIB must be pointing to the default directory.

Assuming the NTM is up and running, an IISS user may start the test using scripting as follows:

```
$ SET DEF <to directory containing NTM environment>
$ VT100 -RFDFEUTP.SCP -SFDFETST.SAV
```

These commands start up the VT100 device driver with a source script as input and specify a save file for the results of the test.

If the User Interface system has been installed at your site with a different device driver, then this step is amended as appropriate. The results of this test are saved in the current directory in the file FDFEUTP.SAV. To execute this test manually enter only "VT100" at the second '$'. This brings up the IISS Logon Screen as shown in Figure 5-2. The inputs and outputs for the test are illustrated by examining the screens shown in section 5.3.4.

5.3.3 IBM Test Procedures

Partitioned datasets must be allocated for each of the following symbolic names: iisssslib, iissflib, iissulib, tmplis and iissmlib. Each of the datasets should be compressed before testing. Additionally, it is recommended that the following dataset characteristics and minimum space allocations be used:

- **iisssslib**: Variable blocked with LRCL 80, BLKSIZE 3120, and 10 tracks with 5 directory blocks.
- **iissflib, iissulib, tmplis**: Variable blocked with LRCL 80, BLKSIZE 3120, and 15 tracks with 15 directory blocks.
- **iissmlib**: Fixed block with LRECL 73, BLKSIZE 730, and 3 tracks with 2 directory blocks.
NOTE that the BATIIS5 JCL that is run to bring up the NTM preallocates these datasets.

Assuming the NTM is up and running, an IISS user may start this test by accessing the IISS environment. To do this, enter "LOGON APPLID(IISSi)" at the ENTER APPLICATION: prompt. The "i" following IISS must be your IISS instance id as specified in the NTM SYSGEN file. This starts up the IBM3270 device driver and brings up the IISS Logon Screen as shown in Figure 5-2. Proceed as described in the following section.

5.3.4 Testing the FDFE

On either an IBM or VAX host, when the User Interface is activated, the following form appears:

```
+---------------------------------------------------------------------------------
| USER ID: ____________ |
| PASSWORD: ____________ |
| ROLE: ____________    |
+---------------------------------------------------------------------------------
```

Figure 5-2 IISS Logon Screen

(1) USER ID is input as "MORENC".

(2) PASSWORD is input as "STANLEY".
(3) ROLE is input as "MANAGER".

When this form is correctly completed and the <ENTER> key is pressed, the IISS Function Screen is displayed.

---

**IISS TEST BED VERSION 3**

DATE: __/__/__  TIME: __:__:__  USER ID: ______  ROLE: ______

FUNCTION: ________  DEVICE TYPE: ________  DEVICE NAME: ______

Msg: 0

application

---

Figure 5-3 IISS Function Screen
To invoke the FDFE enter:

![Command Entry Table](image)

Figure 5-4 Invoking the FDFE
The result should be.

![UIWS FORMS DRIVER FORM EDITOR - VERSION 3.0 JUNE 1, 1985](image)

**WORK TASKS**
- List FDL Source (LS)
- Insert FDL Source (IS)
- Modify FDL Source (MS)
- Select FDL Source (SS)
- Copy FDL Source (CS)
- Rename FDL Source (RS)
- Drop FDL Source (DS)
- List Compiled form definitions (LC)
- View Compiled form definition (VC)
- Drop Compiled form definition (DC)
- Exit form driven form editor (EX)

**Command Entry**

**Figure 5-5 First FDFE Screen - Test Screen 1**
To test inserting a form into an FDL file enter:

```
IIS FOOFS
Df'.VtK
fOOM
EVPt?'
VERSJOt.
```

Figure 5-6 Test Screen 2
The following screen should appear.

Figure 5-7 Test Screen 3
If the following is entered on the edit task menu screen,

Figure 5-8 Test Screen 4
The following screen should appear.

![Diagram of UIMS Forms Driver Form Editor - Version 2.0 June 1, 1985]

Figure 5-9 Test Screen 5
If the following is entered,

![Form Screen Image]

Figure 5-10 Test Screen 6
The following screen should appear.

Figure 5-11 Test Screen 7
To return to edit task menu screen press the <QUIT> key. The result should be.

Figure 5-12 Test Screen 8
If the following is entered on the edit task menu screen,

![Test Screen](image)

Figure 5-13 Test Screen 9
The following screen should appear.

Figure 5-14 Test Screen 10
If the following is entered and <pfl7> is pressed,

![Diagram showing the UIMS FORMS DRIVER FORM EDITOR interface]

Figure 5-15 Test Screen 11
The following screen should appear.

![Test Screen 12](image)

Figure 5-16 Test Screen 12
If the following is entered,

Figure 5-17 Test Screen 13
The following screen should appear.

Figure 5-18 Test Screen 14
If <pfl7> is pressed the following screen should appear.

![Diagram of field edit mode]

Figure 5-19 Test Screen 15
If the following is entered,

Figure 5-20 Test Screen 16
The following screen should appear.

Figure 5-21 Test Screen 17
To return to edit task menu screen press the <QUIT> key. The result should be.

**Figure 5-22 Test Screen 18**
If the following is entered on the edit task menu screen,

![Edit Task Menu Screen](image)

Figure 5-23 Test Screen 19
The following screen should appear.

![Test Screen 20](image)

Figure 5-24 Test Screen 20
If cursor is placed where marked below and <pf16> is pressed,

Figure 5-25 Test Screen 21
The result should be.

Figure 5-26 Test Screen 22
If cursor is placed where marked below and <pf16> is pressed,

Figure 5-27 Test Screen 23
The result should be.

Figure 5-28 Test Screen 24
If cursor is placed where marked below and <ENTER> is pressed,

Figure 5-29 Test Screen 25
The result should be.

Figure 5-30 Test Screen 26
If <pf5> is pressed, the result should be.

Figure 5-31 Test Screen 27
If <pf8> is pressed, the result should be.

![Message Queue](image)

Figure 5-32 Test Screen 28
If <pf6> is pressed, the result should be.

Figure 5-33 Test Screen 29
If <pf7> is pressed, the result should be.

Figure 5-34 Test Screen 30
If the following is entered,

```
UINS PONS DRTS FOI ZDITOP - USLRSION 2.0 JUN 1
M96 VMU ?ASK Pal (FtOs, Nose To/Now Nose Vt17
List VDL Source (Li
Loss FStouc (IS)
Modify PDL Source (RS)
Select PDL Source ($5)
Copy PDL Source (CS)
Rename PDL Source (RS)
Drop PDL Source (BS)
List Compiled form definitions (LC)
View Compiled form definition (VC)
Drop Compiled form definition (DC)
Exit Form driven form editor (ER)
```

**Figure 5-35 Test Screen 31**
The result should be.

Figure 5-36 Test Screen 32
If <pf10> is pressed, the result should be.

Figure 5-37 Test Screen 33
If <pfl0> is pressed, the result should be.

Figure 5-38 Test Screen 34
If <pf10> is pressed, the result should be.

![Figure 5-39 Test Screen 35](image-url)
If <pf10> is pressed, the result should be.

Figure 5-40 Test Screen 36
If the following is entered,

Figure 5-41 Test Screen 37
The result should be.

Figure 5-42 Test Screen 38
If cursor is placed where marked below and <pfl6> is pressed,

![Diagram]

Figure 5-43 Test Screen 39
The result should be.

Figure 5-44 Test Screen 40
If cursor placed where marked below and <pfl6> is pressed,

Figure 5-45 Test Screen 41
The result should be.

---

**Figure 5-46 Test Screen 42**
If cursor is placed where marked below and <ENTER> is pressed,

- *CR_DATE* has been successfully modified
- *FAT TD* has been successfully modified
- *AMOUNT* has been successfully modified
- *DOLLARS* has been successfully modified
- *MEMO* has been successfully modified
- *SIGNATURE* has been successfully modified

![Message Queue Diagram]

Figure 5-47 Test Screen 43
The result should be.

Figure 5-48 Test Screen 44
If cursor is placed where marked below and <pf16> is pressed,

Figure 5-49 Test Screen 45
The result should be.

![Diagram](image)

**Figure 5-50 Test Screen 46**
If cursor is placed where marked below and <pf16> is pressed,

Figure 5-51 Test Screen 47
The result should be.

Figure 5-52 Test Screen 48
If cursor is placed where marked below and <ENTER> is pressed,

Figure 5-53 Test Screen 49
The result should be.

![UIX FORMS DRIVER: FORM EDITOR - VERSION 2.0 JUNE 1, 1985](image)

<table>
<thead>
<tr>
<th>Command Entry</th>
<th>Form Name</th>
<th>Edit Mode</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDIT TASKS Command</td>
<td>List: Forms(LF)</td>
<td>Write: Forms(WF)</td>
<td>Compile: Forms(CF)</td>
</tr>
<tr>
<td></td>
<td>Select a Form (SF)</td>
<td>Insert a Form (IF)</td>
<td>Modify a Form (MF)</td>
</tr>
<tr>
<td></td>
<td>Drop a Form (DF)</td>
<td>Exit Write (EW)</td>
<td>Exit Compile (EC)</td>
</tr>
<tr>
<td></td>
<td>Exit No save (ED)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Edit Task: Modify FD4 Source**

**Form Source: UTIPFORM**

Figure 5-54 Test Screen 50
If cursor is placed where marked below and <ENTER> is pressed,

Figure 5-55 Test Screen 51
The result should be.

Figure 5-56 Test Screen 52
If cursor is placed where marked below and <ENTER> is pressed,

Figure 5-57 Test Screen 53
The result should be.

Figure 5-58 Test Screen 54
If <pfl4> is pressed, the result should be.

Figure 5-59 Test Screen 55
If <pf13> is pressed, the result should be.

![Figure 5-60 Test Screen 56](image-url)
If `<pfl3>` is pressed, the result should be.

![UIP Forms Driver Form Editor - Version 2.0 June 3, 1990](image)

**Figure 5-61 Test Screen 57**
If cursor is placed where marked below and <ENTER> is pressed,

![Diagram of a computer screen with options for form editing]

Figure 5-62 Test Screen 58
The result should be.

Figure 5-63 Test Screen 59
If cursor is placed where marked below and <ENTER> is pressed,

![Figure 5-64 Test Screen 60](image-url)
The result should be.

Figure 5-65 Test Screen 61
If cursor is placed where marked below and <p16> is pressed,

Figure 5-66 Test Screen 62
The result should be.

---

**Figure 5-67 Test Screen 63**

---
If cursor is placed where marked below and <ENTER> is pressed.

Figure 5-68 Test Screen 64
The result should be.

Figure 5-69 Test Screen 65
If cursor is placed where marked below and <pfl6> is pressed,

![Diagram of a test screen]

Figure 5-70 Test Screen 66
The result should be.

Figure 5-71 Test Screen 67
If cursor is placed where marked below and <ENTER> is pressed,

Figure 5-72 Test Screen 68
The result should be.

Figure 5-73 Test Screen 69
If cursor is placed where marked below and <ENTER> is press,

Figure 5-74 Test Screen 70
The result should be.

---

**Figure 5-75 Test Screen 71**

---
If <pf13> is pressed, the result should be.

Figure 5-76 Test Screen 72
If cursor is placed where marked below and <ENTER> is press,

![Diagram of UIMS FORMS DRIVEN FORM EDITOR]

**Figure 5-77 Test Screen 73**
The result should be.

Figure 5-78 Test Screen 74
If cursor is placed where marked below and <ENTER> is pressed,

Figure 5-79 Test Screen 75
The result should be.

![Diagram of form editor interface]

Figure 5-80 Test Screen 76
If the following is entered,

![Figure 5-81 Test Screen 77](image-url)
The result should be.

Figure 5-82 Test Screen 78
If the following is entered,

![Diagram of command entry and command tasks]

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>List PDL Source</td>
<td>List PDL Source</td>
</tr>
<tr>
<td>Insert PDL Source</td>
<td>Insert PDL Source</td>
</tr>
<tr>
<td>Modify PDL Source</td>
<td>Modify PDL Source</td>
</tr>
<tr>
<td>Select PDL Source</td>
<td>Select PDL Source</td>
</tr>
<tr>
<td>Copy PDL Source</td>
<td>Copy PDL Source</td>
</tr>
<tr>
<td>Rename PDL Source</td>
<td>Rename PDL Source</td>
</tr>
<tr>
<td>Drop PDL Source</td>
<td>Drop PDL Source</td>
</tr>
<tr>
<td>List Compiled form definition</td>
<td>List Compiled form definition</td>
</tr>
<tr>
<td>View Compiled form definition</td>
<td>View Compiled form definition</td>
</tr>
<tr>
<td>Drop Compiled form definition</td>
<td>Drop Compiled form definition</td>
</tr>
<tr>
<td>Exit form driven form editor</td>
<td>Exit form driven form editor</td>
</tr>
</tbody>
</table>

Figure 5-83 Test Screen 79
The result should be.

Figure 5-84 Test Screen 80
If <pfl4> is pressed, the result should be.

**Figuer 5-85 Test Screen 81**
If the following is entered,

![Diagram of a form editor interface]

Figure 5-86 Test Screen 82
The result should be.

Figure 5-87 Test Screen 83