INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 18 - Forms Language Compiler 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

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533
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.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

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

DATE 25 July 91

FOR THE COMMANDER:

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.
REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
   Unclassified

2a. SECURITY CLASSIFICATION AUTHORITY
   RIPONT SEGURITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE
   RESTRICTIVE MARKINGS

3. DISTRIBUTION/AVAILABILITY OF REPORT
   Approved for Public Release:
   Distribution is Unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
   UTP620344401

5. MONITORING ORGANIZATION REPORT NUMBER(S)
   WRDC-TR-90-8007 Vol. VIII, Part 18

6a. NAME OF PERFORMING ORGANIZATION
   Control Data Corporation;
   Integration Technology Services

6b. OFFICE SYMBOL
    (if applicable)

7a. NAME OF MONITORING ORGANIZATION
    WRDC/MTI

7b. ADDRESS (City, State, and ZIP Code)
    WPAFB, OH 45433-6533

8a. NAME OF FUNDING/SPONSORING
    Wright Research and Development Center,
    Air Force Systems Command, USAF

8b. OFFICE SYMBOL
    (if applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM.
   F33600-87-C-0464

10. SOURCE OF FUNDING NOS.

11. TITLE
    Forms La

12. PERSONAL AUTHOR(S)
    Structural Dynamics Research Corporation: Barker, S.

13a. TYPE OF REPORT
    Final Report

13b. TIME COVERED
    4/1/87-12/31/90

14. DATE OF REPORT (Yr.,Mo.,Day)
    1990 September 30

15. PAGE COUNT
    298

16. SUPPLEMENTARY NOTES
    WRDC/MTI Project Priority 6203

17. COSATI CODES

18. SUBJECT TERMS
   (Continue on reverse if necessary and identify block no.)

19. ABSTRACT
   (Continue on reverse if necessary and identify block number)
   This unit test plan establishes the methodology and procedures to test the Forms Definition Language Compiler computer program.

   BLOCK 11:

   INTEGRATED INFORMATION SUPPORT SYSTEM
   Vol VIII - User Interface Subsystem

   Part 18 - Forms Language Compiler Unit Test Plan

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT

   UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
    Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
    David L. Judson

22b. TELEPHONE NO.
    (513) 255-7371

22c. OFFICE SYMBOL
    WRDC/MTI
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 Research Corporation</td>
<td>Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.</td>
</tr>
<tr>
<td>Arizona State University</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 Reference Documents</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>3-3</td>
</tr>
<tr>
<td>3.3 First Location Testing</td>
<td>3-3</td>
</tr>
<tr>
<td>3.3.1 VAX Environment Test Materials</td>
<td>3-4</td>
</tr>
<tr>
<td>3.4 Subsequent Location Testing</td>
<td>3-4</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 Specifications</td>
<td>4-1</td>
</tr>
<tr>
<td>4.1.1 Test Forms</td>
<td>4-1</td>
</tr>
<tr>
<td>4.1.2 Test Graphs</td>
<td>4-2</td>
</tr>
<tr>
<td>4.1.3 Test 2-D Graphics</td>
<td>4-13</td>
</tr>
<tr>
<td>4.2 Test Methods and Constraints</td>
<td>4-14</td>
</tr>
<tr>
<td>4.3 Test Progression</td>
<td>4-14</td>
</tr>
<tr>
<td>4.4 Test Evaluation</td>
<td>4-14</td>
</tr>
<tr>
<td>4.4.1 Test Evaluation Stages</td>
<td>4-14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION 5.0 TEST PROCEDURES</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Forms Test Description</td>
<td>5-1</td>
</tr>
<tr>
<td>5.1.1 Forms Test Control</td>
<td>5-1</td>
</tr>
<tr>
<td>5.1.2 Forms Test Procedures</td>
<td>5-1</td>
</tr>
<tr>
<td>5.1.2.1 VAX Test Procedures</td>
<td>5-2</td>
</tr>
<tr>
<td>5.1.2.2 Choosing the FLAN Function</td>
<td>5-3</td>
</tr>
<tr>
<td>5.1.2.3 Standalone Version of FLAN on VAX</td>
<td>5-7</td>
</tr>
<tr>
<td>5.1.3 IBM Test Procedures</td>
<td>5-8</td>
</tr>
<tr>
<td>5.1.3.1 Choosing the FLAN Function</td>
<td>5-8</td>
</tr>
<tr>
<td>5.1.3.2 Standalone version of FLAN on IBM</td>
<td>5-9</td>
</tr>
<tr>
<td>5.1.4 APPEARS IF Tests</td>
<td>5-11</td>
</tr>
<tr>
<td>5.2 Graph Test Description</td>
<td>5-123</td>
</tr>
<tr>
<td>5.2.1 Graph Test Control</td>
<td>5-123</td>
</tr>
<tr>
<td>5.2.2 Graph Test Procedures</td>
<td>5-123</td>
</tr>
<tr>
<td>5.2.2.1 Graph VAX Test Procedures</td>
<td>5-123</td>
</tr>
<tr>
<td>5.2.2.1.1 Access to GDL Test Programs</td>
<td>5-124</td>
</tr>
<tr>
<td>5.2.2.1.2 Running The GRFTST Program</td>
<td>5-126</td>
</tr>
<tr>
<td>5.2.2.1.3 Running The GRAFDE Program</td>
<td>5-126</td>
</tr>
<tr>
<td>5.3 2-D Graphics Test Description</td>
<td>5-127</td>
</tr>
<tr>
<td>5.3.1 2-D Graphics Test Control</td>
<td>5-128</td>
</tr>
<tr>
<td>5.3.2 2-D Graphics Test Procedures</td>
<td>5-128</td>
</tr>
<tr>
<td>5.3.2.1 2-D Graphics VAX Test Procedures</td>
<td>5-128</td>
</tr>
<tr>
<td>5.3.2.1.1 Access To 2-D Graphics Test Programs</td>
<td>5-129</td>
</tr>
<tr>
<td>5.3.2.1.2 Running The ICONST Program</td>
<td>5-132</td>
</tr>
<tr>
<td>Appendix</td>
<td>Title</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>A</td>
<td>FLAN1.FDL</td>
</tr>
<tr>
<td>B</td>
<td>FLAN2.FDL</td>
</tr>
<tr>
<td>C</td>
<td>SCREENS AND GDL FOR GRFTST</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 GDL Functions to Test Graphs</td>
<td>4-8</td>
</tr>
<tr>
<td>4-2</td>
<td>Matrix Mapping GDL Functions to Test Graphs</td>
<td>4-11</td>
</tr>
<tr>
<td>Figure</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3-1</td>
<td>FLAN Interfaces</td>
<td>3-2</td>
</tr>
<tr>
<td>5-1</td>
<td>IISS Logon Screen</td>
<td>5-2</td>
</tr>
<tr>
<td>5-2</td>
<td>IISS Function Screen</td>
<td>5-3</td>
</tr>
<tr>
<td>5-3</td>
<td>FLAN Screen</td>
<td>5-4</td>
</tr>
<tr>
<td>5-4</td>
<td>Starting ARTEST Application</td>
<td>5-4</td>
</tr>
<tr>
<td>5-5</td>
<td>ARTEST Screen</td>
<td>5-5</td>
</tr>
<tr>
<td>5-6</td>
<td>Add Testform to Screen</td>
<td>5-6</td>
</tr>
<tr>
<td>5-7</td>
<td>FLAN1.FDL/TESTFORM Screen</td>
<td>5-7</td>
</tr>
<tr>
<td>5-8</td>
<td>APPEARS IF Test Form</td>
<td>5-10</td>
</tr>
<tr>
<td>5-9a</td>
<td>Test Case 1</td>
<td>5-11</td>
</tr>
<tr>
<td>5-9b</td>
<td>I3 Appears</td>
<td>5-12</td>
</tr>
<tr>
<td>5-10a</td>
<td>Test Case 2</td>
<td>5-13</td>
</tr>
<tr>
<td>5-10b</td>
<td>I3 Does Not Appear</td>
<td>5-14</td>
</tr>
<tr>
<td>5-11a</td>
<td>Test Case 3</td>
<td>5-15</td>
</tr>
<tr>
<td>5-11b</td>
<td>I3 Appears</td>
<td>5-16</td>
</tr>
<tr>
<td>5-11c</td>
<td>Change Il Value</td>
<td>5-17</td>
</tr>
<tr>
<td>5-11d</td>
<td>I3 Does Not Appear</td>
<td>5-18</td>
</tr>
<tr>
<td>5-11e</td>
<td>Change Il Value</td>
<td>5-19</td>
</tr>
<tr>
<td>5-11f</td>
<td>I3 Appears</td>
<td>5-20</td>
</tr>
<tr>
<td>5-12a</td>
<td>Test Case 4</td>
<td>5-21</td>
</tr>
<tr>
<td>5-12b</td>
<td>Does Not Appear</td>
<td>5-22</td>
</tr>
<tr>
<td>5-12c</td>
<td>Change Il Value</td>
<td>5-23</td>
</tr>
<tr>
<td>5-12d</td>
<td>I3 Appears</td>
<td>5-24</td>
</tr>
<tr>
<td>5-12e</td>
<td>Change Il Value</td>
<td>5-25</td>
</tr>
<tr>
<td>5-12f</td>
<td>I3 Appears</td>
<td>5-26</td>
</tr>
<tr>
<td>5-13a</td>
<td>Test Case 5</td>
<td>5-27</td>
</tr>
<tr>
<td>5-13b</td>
<td>I3 Appears</td>
<td>5-28</td>
</tr>
<tr>
<td>5-13c</td>
<td>Change Il Value</td>
<td>5-29</td>
</tr>
<tr>
<td>5-13d</td>
<td>I3 Does Not Appear</td>
<td>5-30</td>
</tr>
<tr>
<td>5-13e</td>
<td>Change Il Value</td>
<td>5-31</td>
</tr>
<tr>
<td>5-13f</td>
<td>I3 Appears</td>
<td>5-32</td>
</tr>
<tr>
<td>5-14a</td>
<td>Test Case 6</td>
<td>5-33</td>
</tr>
<tr>
<td>5-14b</td>
<td>I3 does Not Appear</td>
<td>5-34</td>
</tr>
<tr>
<td>5-14c</td>
<td>Change Il Value</td>
<td>5-35</td>
</tr>
<tr>
<td>5-14d</td>
<td>I3 Appears</td>
<td>5-36</td>
</tr>
<tr>
<td>5-14e</td>
<td>Change Il Value</td>
<td>5-37</td>
</tr>
<tr>
<td>5-14f</td>
<td>I3 Does Not Appear</td>
<td>5-38</td>
</tr>
<tr>
<td>5-15a</td>
<td>Test Case 7</td>
<td>5-39</td>
</tr>
<tr>
<td>5-15b</td>
<td>I3 Does Not Appear</td>
<td>5-40</td>
</tr>
<tr>
<td>5-15c</td>
<td>Change Il Value</td>
<td>5-41</td>
</tr>
<tr>
<td>5-15d</td>
<td>I3 Appears</td>
<td>5-42</td>
</tr>
<tr>
<td>5-15e</td>
<td>change Il Value</td>
<td>5-43</td>
</tr>
<tr>
<td>5-15f</td>
<td>I3 Does Not Appear</td>
<td>5-44</td>
</tr>
<tr>
<td>5-16a</td>
<td>Test Case 8</td>
<td>5-45</td>
</tr>
<tr>
<td>5-16b</td>
<td>I3 Appears</td>
<td>5-46</td>
</tr>
<tr>
<td>5-16c</td>
<td>Change Il Value</td>
<td>5-47</td>
</tr>
<tr>
<td>5-16d</td>
<td>I3 Does Not Appear</td>
<td>5-48</td>
</tr>
<tr>
<td>5-16e</td>
<td>Change Il Value</td>
<td>5-49</td>
</tr>
<tr>
<td>5-16f</td>
<td>Ie Appears</td>
<td>5-50</td>
</tr>
<tr>
<td>5-17a</td>
<td>Test Case 9</td>
<td>5-51</td>
</tr>
<tr>
<td>5-17b</td>
<td>I3 Does Not Appear</td>
<td>5-52</td>
</tr>
</tbody>
</table>

vii
<table>
<thead>
<tr>
<th>Test Case</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-18a</td>
<td>Test Case 10</td>
<td>5-53</td>
</tr>
<tr>
<td>5-18b</td>
<td>I3 Appears</td>
<td>5-54</td>
</tr>
<tr>
<td>5-19a</td>
<td>Test Case 11</td>
<td>5-55</td>
</tr>
<tr>
<td>5-19b</td>
<td>I3 Appears</td>
<td>5-56</td>
</tr>
<tr>
<td>5-19c</td>
<td>Change I1 Value</td>
<td>5-57</td>
</tr>
<tr>
<td>5-19d</td>
<td>I3 Does Not Appear</td>
<td>5-58</td>
</tr>
<tr>
<td>5-19e</td>
<td>Change I1 Value</td>
<td>5-59</td>
</tr>
<tr>
<td>5-19f</td>
<td>I3 Appears</td>
<td>5-60</td>
</tr>
<tr>
<td>5-20a</td>
<td>Test Case 12</td>
<td>5-61</td>
</tr>
<tr>
<td>5-20b</td>
<td>I3 Does Not Appear</td>
<td>5-62</td>
</tr>
<tr>
<td>5-20c</td>
<td>Change I1 Value</td>
<td>5-63</td>
</tr>
<tr>
<td>5-20d</td>
<td>I3 Appears</td>
<td>5-64</td>
</tr>
<tr>
<td>5-20e</td>
<td>Change I1 Value</td>
<td>5-65</td>
</tr>
<tr>
<td>5-20f</td>
<td>I3 Does Not Appear</td>
<td>5-66</td>
</tr>
<tr>
<td>5-21a</td>
<td>Test Case 13</td>
<td>5-67</td>
</tr>
<tr>
<td>5-21b</td>
<td>I3 does Not Appear</td>
<td>5-68</td>
</tr>
<tr>
<td>5-21c</td>
<td>Create Page 2 in W3</td>
<td>5-69</td>
</tr>
<tr>
<td>5-21d</td>
<td>I3 Appears</td>
<td>5-70</td>
</tr>
<tr>
<td>5-21e</td>
<td>Remove Page 2 in W3</td>
<td>5-71</td>
</tr>
<tr>
<td>5-21f</td>
<td>I3 does Not Appear</td>
<td>5-72</td>
</tr>
<tr>
<td>5-22a</td>
<td>Test Case 14</td>
<td>5-73</td>
</tr>
<tr>
<td>5-22b</td>
<td>I3 does Not Appear</td>
<td>5-74</td>
</tr>
<tr>
<td>5-22c</td>
<td>I3 appears</td>
<td>5-75</td>
</tr>
<tr>
<td>5-22d</td>
<td>I3 does not appear</td>
<td>5-76</td>
</tr>
<tr>
<td>5-23a</td>
<td>Test Case 15</td>
<td>5-77</td>
</tr>
<tr>
<td>5-23b</td>
<td>I3 Does Not appear in W1</td>
<td>5-78</td>
</tr>
<tr>
<td>5-23c</td>
<td>Change Contents of Page 1 in W3</td>
<td>5-79</td>
</tr>
<tr>
<td>5-23d</td>
<td>I3 appears in W1</td>
<td>5-80</td>
</tr>
<tr>
<td>5-23e</td>
<td>Change Contents of Page 1 in W3</td>
<td>5-81</td>
</tr>
<tr>
<td>5-23f</td>
<td>I3 Does Not Appear in W1</td>
<td>5-82</td>
</tr>
<tr>
<td>5-24a</td>
<td>Test Case 16</td>
<td>5-83</td>
</tr>
<tr>
<td>5-24b</td>
<td>I3 does Not Appear</td>
<td>5-84</td>
</tr>
<tr>
<td>5-25a</td>
<td>Test Case 17</td>
<td>5-85</td>
</tr>
<tr>
<td>5-25b</td>
<td>I3 Appears</td>
<td>5-86</td>
</tr>
<tr>
<td>5-26a</td>
<td>Test Case 18</td>
<td>5-87</td>
</tr>
<tr>
<td>5-26b</td>
<td>I3 appears</td>
<td>5-88</td>
</tr>
<tr>
<td>5-27a</td>
<td>Test Case 19</td>
<td>5-89</td>
</tr>
<tr>
<td>5-27b</td>
<td>I3 Does Not Appear</td>
<td>5-90</td>
</tr>
<tr>
<td>5-27c</td>
<td>Change I1 Value</td>
<td>5-91</td>
</tr>
<tr>
<td>5-27d</td>
<td>I3 Appears</td>
<td>5-92</td>
</tr>
<tr>
<td>5-27e</td>
<td>Change I1 Value</td>
<td>5-93</td>
</tr>
<tr>
<td>5-27f</td>
<td>I3 does Not Appear</td>
<td>5-94</td>
</tr>
<tr>
<td>5-28a</td>
<td>Test Case 20</td>
<td>5-95</td>
</tr>
<tr>
<td>5-28b</td>
<td>I3 Does Not Appear</td>
<td>5-96</td>
</tr>
<tr>
<td>5-28c</td>
<td>Change I1 value</td>
<td>5-97</td>
</tr>
<tr>
<td>5-28d</td>
<td>I3 Does Not Appear</td>
<td>5-98</td>
</tr>
<tr>
<td>5-28e</td>
<td>Change I1 Value</td>
<td>5-99</td>
</tr>
<tr>
<td>5-28f</td>
<td>I3 Appears</td>
<td>5-100</td>
</tr>
<tr>
<td>5-29a</td>
<td>Test Case 21</td>
<td>5-101</td>
</tr>
<tr>
<td>5-29b</td>
<td>I3 Appears</td>
<td>5-102</td>
</tr>
<tr>
<td>5-29c</td>
<td>Change I1 Value</td>
<td>5-103</td>
</tr>
<tr>
<td>5-29d</td>
<td>I3 Does Not Appear</td>
<td>5-104</td>
</tr>
<tr>
<td>5-29e</td>
<td>Change I1 value</td>
<td>5-105</td>
</tr>
<tr>
<td>5-29f</td>
<td>I3 appears</td>
<td>5-106</td>
</tr>
</tbody>
</table>
5-30a Test Case 22 .................................. 5-107
5-30b 13 does Not appear ........................... 5-108
5-30c Change II Value ................................ 5-109
5-30d 13 Appears .................................... 5-110
5-30e Change II value ................................ 5-111
5-30f 13 Does Not appear ............................. 5-112
5-31a Test Case 23 ................................... 5-113
5-31b F1 Does Not Appear ............................ 5-114
5-31c Change II Value ................................ 5-115
5-31d F1 Appears ..................................... 5-116
5-31e Change II value ................................ 5-117
5-31f F1 Does Not appear ............................. 5-118
5-32a Test Case 24 ................................... 5-119
5-32b W1 No Longe Appears .......................... 5-120
5-32c Change FF6.I1 Value ............................ 5-121
5-32d W1 appears ...................................... 5-122
5-33 IISS Logon Screen ............................... 5-124
5-34 IISS Function Screen ............................. 5-125
5-35 Initial GRAFDE Screen ........................... 5-126
5-36 Test Data for GRAFDE ............................ 5-127
5-37 IISS Logon Screen ............................... 5-129
5-38 IISS Function Screen ............................. 5-130
5-39 ICNTST startup screen ........................... 5-132
5-40 FORMS Data Input screen ........................ 5-133
5-41 Molecule Graphics form .......................... 5-134
5-42 Example display of Molecule Graphics .......... 5-135
5-43 Sales Graphs form ............................... 5-136
5-44 Example display of Sales Graphs ................. 5-137
C-1 GDL Test Activity A: and corresponding GDL C-2
C-2 GDL Test Activity B: and corresponding GDL C-6
C-3 GDL Test Activity C: and corresponding GDL C-9
C-4 GDL Test Activity D: and corresponding GDL C-11
C-5 GDL Test Activity E: and corresponding GDL C-13
C-6 GDL Test Activity F: and corresponding GDL C-16
C-7 GDL Test Activity G: and corresponding GDL C-19
C-8 GDL Test Activity G: and corresponding GDL C-21
C-9 GDL Test Activity G: and corresponding GDL C-23
C-10 GDL Test Activity G: and corresponding GDL C-25
C-11 GDL Test Activity G: and corresponding GDL C-28
C-12 GDL Test Activity G: and corresponding GDL C-30
C-13 GDL Test Activity G: and corresponding GDL C-33
C-14 GDL Test Activity G: and corresponding GDL C-36
C-15 GDL Test Activity G: and corresponding GDL C-39
C-16 GDL Test Activity G: and corresponding GDL C-42
C-17 GDL Test Activity G: and corresponding GDL C-45
C-18 GDL Test Activity G: and corresponding GDL C-48
C-19 GDL Test Activity G: and corresponding GDL C-51
C-20 GDL Test Activity G: and corresponding GDL C-54
C-21 GDL Test Activity G: and corresponding GDL C-57
C-22 GDL Test Activity G: and corresponding GDL C-59
C-23 GDL Test Activity G: and corresponding GDL C-61
C-24 GDL Test Activity G: and corresponding GDL C-63
C-25 GDL Test Activity G: and corresponding GDL C-65
C-26 GDL Test Activity G: and corresponding GDL C-67
C-27 GDL Test Activity G: and corresponding GDL C-70

ix
SECTION 1
GENERAL

1.1 Purpose

This unit test plan establishes the methodology and procedures used to adequately test the capabilities of the computer programs identified as the Forms Definition Language Compiler known in this document as FLAN and MAKE Includes known as MAKINC. FLAN and MAKINC are configuration items of the Integrated Information Support System (IISS) User Interface (UI).

1.2 Reference Documents


1.3 Terms and Abbreviations

Application Generator (AG): A subset of the IISS User Interface that consists of software modules that generate IISS application code and associated form definitions based on a language input. The part of the AG that generates report programs is called the Report Writer. The part of the AG that generates interactive applications is called the Rapid Application Generator.

Application Interface (AI): A 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: A field characteristic such as blinking, highlighted, black, etc., and various other combinations. Background attributes are defined for some 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.

Closed Figure: A figure is closed if the path traced by a moving point returns to its starting position. The starting position may be arbitrarily assigned. "Fillarea" is synonymous with "closed figure".

Complex Figure: A figure is complex if the path traced by a moving point crosses itself. An arbitrary point may be determined to be contained within the traced boundary if a line drawn to infinity crosses the boundary an odd number of times. If the number of crossings is zero or even, the point is outside the traced boundary.

Dependent Data: Data correlated to a dependent variable.

Dependent Variable: A mathematical variable whose value is determined by that of one or more other variables in a function.
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: An internal Form Processor list that contains only those forms that have been added to the screen and are currently displayed on the screen, along with information on where those forms are used.

Element: A graphics line or other primitive composed of graphics lines, such as an arc.

Field: In reference to the Forms Processor, "field" refers to any object on the open or display list. These objects can be forms, items, windows, etc.

In reference to graphs, "field" refers to a collection of one or more graph figures. A graph field can be an axis, curve, pie chart, grid, etc.

Figure: A collection of elements. A figure may be closed or open.

Fillarea: A collection of elements. A fillarea must be closed. "Closed figure" is synonymous with "fillarea".

Form: A 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, windows, prompts, non-graphics lines, and graphics.

Forms Definition Language (FDL): The language in which electronic forms are defined.

Forms Driven Form Editor (FDFE): A subset of the Form Editor which consists of a forms-driven application used to create and/or modify Form Definition files interactively.

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

Form Hierarchy: A graphic representation of the way in which fields are related to their parent form.

Forms Language Compiler (FLAN): A subset of the Form Editor that consists of a batch process that accepts a series of Forms Definition Language (FDL) statements and produces form definition files as output.

Form Processor (FP): A subset of the IISS User Interface that consists of a set of callable execution-time routines available to an application program for form processing.
Graph: A picture correlated with data that alters as the data changes; by necessity, this is a dynamic (not pre-defined) picture. A graph may be imposed on windows or forms.

Graph Definition Language (GDL): An extension of the Forms Definition Language (FDL) which is used to define business graphs such as pie charts, X-Y plots, and bar charts.

Graph Figure: A collection of graphics primitives. The primitives can be circles, lines, arcs, etc.

Graphics Kernal System (GKS): A 2-dimensional graphics standard which is defined independently of any programming language.

Icon: A collection of figures and points that is pre-defined. An icon may be imposed on windows or forms. "Icon" is synonymous with "picture".

Independent Data: Data that is correlated to an independent variable.

Independent Variable: A mathematical variable whose value is specified first and determines the value of one or more other values in an expression or function. For example, in a business graph of sales versus month, month is the independent variable and sales is the dependent variable, because sales varies by month.

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 (LAN).

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

Local Area Network (LAN): A privately owned network that offers reliable, high-speed communications channels optimized for connecting information processing equipment in a limited geographic area.

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

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

Open Figure: A figure is open if the path traced by a moving point does not return to its starting position. The starting position may be arbitrarily assigned. "Polyline" is synonymous with "open figure".
Open List: An internal Form Processor list that contains all forms that the application has opened for use along with information on where the form is used.

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: An instance of a form in a window that is created whenever a form is added to a window.

Physical Device: A hardware terminal.

Picture: A collection of figures and points that is pre-defined. A picture may be imposed on a window or a form. "Picture" is synonymous with "icon".

Picture Definition Language (PDL): An extension of the Forms Definition Language (FDL) which allows the definition of any graphics picture.

Point: A marker or a symbol.

Polyline: A collection of elements. A polyline must be an open figure. "Open figure" is synonymous with "polyline".

Primitive: The smallest unit of graphic detail. A graphic primitive can be a line, point, arc, etc.

Qualified Name: The name of a field preceded by the hierarchy path so that it is uniquely identified.

Report Writer (RW): Part of the Application Generator (AG) that generates source code for report programs based on a language input.

Subform: A form that is used within another form.

Text Editor (TE): A subset of the IISS User Interface that consists of a file editor that is based on the text editing functions built into the Form Processor (FP).

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

User Interface (UI): A subsystem of IISS that controls the user's terminal and interfaces with the rest of the subsystem. 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): A collection of IISS User Interface subsystems that is used by application programmers as they develop IISS applications. The UIDS includes the Form Editor (FE) and the Application Generator (AG).
User Interface Management System (UIMS): The run-time UI. It consists of the Form Processor (FP), Virtual Terminal (VT), Application Interface (AI), the User Interface Services (UIS), and the Text Editor (TE).

User Interface Services (UIS): A 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.

Window: A dynamic area of a terminal screen on which pre-defined 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 (FP).
SECTION 2
DEVELOPMENT ACTIVITY

2.1 Statement of Pretest Activity

During system development, the computer programs will be
tested progressively. Functionality will be incrementally
tested and as bugs are discovered, the software will be
corrected.

2.2 Pretest Activity Results

This activity is not applicable until development begins.
SECTION 3

SYSTEM DESCRIPTION

3.1 System Description

FLAN is a compiler which translates Form Definition Language source files into binary Form Definition File format. The binary Form Definition Files are then used as input by the Form Processor (another configuration item of the IISS UI) for display and entry of data under the control of other application programs.

The format of the binary Form Definition Files produced by FLAN is constrained to agree with the format expected by the Form Processor configuration item.

The syntax of the Form Definition Language accepted as input is described in the Forms Language Compiler Development Specification.

The interface block diagram for FLAN is shown in Figure 3-1. The top box represents the file MYFORMS which is input to the FLAN compiler (second box). FLAN produces a Form Definition object file (FD) for each CREATE FORM statement in the source file. Each FD file is input for the Form Processor which is part of the User Interface system. The compilation of an FDL file which results in an FD file is the same as program language compilation. The FDL file is the source; the FD file is the object.
Figure 3-1  FLAN Interfaces
While FLAN is normally invoked from the IISS Function Screen another version is available which can be invoked from the host system. This second version is required so current configuration management software can be used in managing FDL files in a manner similar to other source files.

MAKINC is a program that creates program variable declarations which correspond to the structure of a form and may be used in application programs which make use of the Form Processor calls PDATA and GDATA. The following programming languages are supported: PL/I, COBOL, and C. MAKINC is invoked from the host system.

3.2 Testing Schedule

The execution of FLAN is dependent upon the NTM subsystem of IISS and testing of FLAN must be done only after the NTM has been successfully tested. Within the UI subsystem, FLAN uses the Forms Processor and must be tested only after its successful test.

3.3 First Location Testing

These tests of FLAN require the following:

Equipment: IISS Air Force Testbed VAX or IBM, terminals supported by the Virtual Terminal as listed in the IISS Terminal Operator Guide.


Personnel: One integrator familiar with the IISS FLAN.

Training: FLAN training and manuals have been previously provided with all past releases.

Deliverables: The Forms Language Compiler subsystem of the IISS UI/VTI.

Security considerations: None.

Test Materials: This test is interactive and can be manually performed as outlined in this test plan.
3.3.1 **VAX Environment Test Materials**

This test also could be run as a script file if so desired. No script file has been provided because it is believed that on first testing it should be observed and then may be run again to create a script file for later testing reruns.

3.4 **Subsequent Location Testing**

The requirements as listed above need to be met. The script file, FLANUTP.SCP and the saved output to be used for comparison, FLANUTP.SAV are under IISS Configuration Management.
SECTION 4
SPECIFICATIONS AND EVALUATIONS

4.1 Test Specification

The following requirements are demonstrated by the outlined tests:

4.1.1 Test Forms

<table>
<thead>
<tr>
<th>Functional Requirements</th>
<th>Test Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specification of forms:</td>
<td>A B C D E F G H I J K L M</td>
</tr>
<tr>
<td>background attributes</td>
<td>*</td>
</tr>
<tr>
<td>form prompts</td>
<td>*</td>
</tr>
<tr>
<td>size</td>
<td>*</td>
</tr>
<tr>
<td>fields</td>
<td>*</td>
</tr>
<tr>
<td>Specification of fields:</td>
<td>*</td>
</tr>
<tr>
<td>type of field</td>
<td>*</td>
</tr>
<tr>
<td>arrays</td>
<td>*</td>
</tr>
<tr>
<td>location</td>
<td>*</td>
</tr>
<tr>
<td>size</td>
<td>*</td>
</tr>
<tr>
<td>display attributes</td>
<td>*</td>
</tr>
<tr>
<td>field prompts</td>
<td>*</td>
</tr>
<tr>
<td>domain (item only)</td>
<td>*</td>
</tr>
<tr>
<td>help (message and form)</td>
<td>*</td>
</tr>
<tr>
<td>value (item only)</td>
<td>*</td>
</tr>
<tr>
<td>appears if</td>
<td>*</td>
</tr>
<tr>
<td>generate form</td>
<td>*</td>
</tr>
<tr>
<td>definitions</td>
<td>*</td>
</tr>
<tr>
<td>semantic error messages</td>
<td>*</td>
</tr>
</tbody>
</table>

A - input of forms fat1 and fat2.
B - input of form testform.
C - input of field types: items, windows and forms.
D - input of item field i4.
E - input of all fields.
F - input of all items and windows.
G - input of items i1, i5, i6, i7, i8, i9, i10, 10, window w1, form fat1.
H - input of items i5, i6, i7, i8.
I - input of items i9, i10.
J - input of items i11, i2, i3.
K - form definitions used by Form Processor.
L - input of file FLAN2.FDL.
M - Figures 5-9a through 5-32d
4.1.2 Test Graphs

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

List of Functions

GRAPH DEFINITION
1. bar
2. pie
3. line
4. independent axis
5. independent data

ATTRIBUTE DEFINITION
6. color
7. font
8. size
9. upvector
10. line width
11. line type
12. symbol
13. symbol frequency

DATA LOCATION
14. constant list
15. path list

CURVE DEFINITION
16. absolute display
17. additive display
18. dependent axis
19. independent data
20. shading
21. monochromatic shading
22. display
23. monochromatic display
24. legend label
LEGEND
25. enclosed
26. not enclosed
27. horizontal
28. vertical

PIE SEGMENT
29. explosion
30. shading
31. monochromatic shading
32. legend label
33. label
34. inside percent label
35. outside percent label
36. inside quantity label
37. outside quantity label

AXIS DEFINITION
38. length
39. log scale
40. linear scale
41. grid lines
42. fine grid lines
43. horizontal
44. vertical
45. location
46. label
47. maximum limit
48. minimum limit
49. minor tick marks
50. major tick marks by step
51. major tick marks by number
52. major tick mark labels

AUTOMATIC GENERATION
53. independent axis
54. dependent axis
55. tick marks
56. axis length
57. minimum axis value
58. maximum axis value
59. tick mark labels
60. legend labels
61. pie segments
62. pie segment percent label
63. automatic layout

CLIPPING
64. polyline clipping
65. fillarea clipping
66. text clipping
Tables 4-1 and 4-2 show the direct correspondence between the test graphs and the functional requirements as listed in this section. These functions directly correspond to the detailed functional requirements of the Graph Definition Language Development Specification. The 'x' indicates the tests for the functionality implemented in the current release. The '*' indicates functionality not yet implemented.
| 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 |
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 63 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 64 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 65 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 66 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

Table 4-1 Matrix Mapping GDL Functions to Test Graphs
<table>
<thead>
<tr>
<th>AA</th>
<th>BB</th>
<th>CC</th>
<th>DD</th>
<th>EE</th>
<th>FF</th>
<th>GG</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>14</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>15</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>16</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>20</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>22</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th></th>
<th>AA</th>
<th>BB</th>
<th>CC</th>
<th>DD</th>
<th>EE</th>
<th>FF</th>
<th>GG</th>
</tr>
</thead>
<tbody>
<tr>
<td>45</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>48</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>49</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>54</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>55</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>57</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>58</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>61</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>62</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>63</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>65</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>66</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4-2 Matrix Mapping GDL Functions to Test Graphs
The test activities labeled A through GG map to the figures in Appendices C and D as follows:

A - Figure C-1
B - Figure C-2
C - Figure C-3
D - Figure C-4
E - Figure C-5
F - Figure C-6
G - Figure C-7
H - Figure C-8
I - Figure C-9
J - Figure C-10
K - Figure C-11
L - Figure C-12
M - Figure C-13
N - Figure C-14
O - Figure C-15
P - Figure C-16
Q - Figure C-17
R - Figure C-18
S - Figure C-19
T - Figure C-20
U - Figure C-21
V - Figure C-22
W - Figure C-23
X - Figure C-24
Y - Figure C-25
Z - Figure C-26
AA - Figure C-27
BB - Figure D-1
CC - Figure D-2
DD - Figure D-3
EE - Figure D-4
FF - Figure D-5
GG - Figure D-6
4.1.3 Test 2-D Graphics

This section describes the functionality of the Graphics Definition Language (GDL) test outlined in Section 5:

<table>
<thead>
<tr>
<th>Functional Requirements</th>
<th>Test Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Specification of Graphic Primitives:</td>
<td></td>
</tr>
<tr>
<td>Polyline</td>
<td>*</td>
</tr>
<tr>
<td>Polymarker</td>
<td>*</td>
</tr>
<tr>
<td>Fill Area</td>
<td></td>
</tr>
<tr>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>Display Attributes:</td>
<td></td>
</tr>
<tr>
<td>Size</td>
<td>*</td>
</tr>
<tr>
<td>Location</td>
<td></td>
</tr>
<tr>
<td>Style</td>
<td></td>
</tr>
<tr>
<td>Scroll Large 2-D Graphics region</td>
<td>*</td>
</tr>
<tr>
<td>Pick ICON</td>
<td></td>
</tr>
<tr>
<td>Combine Business graphics with 2-D graphics:</td>
<td>*</td>
</tr>
<tr>
<td>Subform with array of Items</td>
<td></td>
</tr>
</tbody>
</table>

The test activities labeled A through D map to the figures as follows:

A - Figure 5-39
B - Figure 5-40
C - Figure 5-41, 5-42
D - Figure 5-43, 5-44

The steps outlined in Section 5 and the files in the appendices show the direct correspondence between the test and the functional requirements as listed in this section.
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 FLAN might create by faulty entry of field information. These tests have been done, however, through the normal testing done by the developer of these functions. No data recording is required. No additional constraints are placed on this unit test besides those listed in Section 5 of this unit test plan.

4.3 Test Progression

The progression of testing of the FLAN 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

If scripting is used on a VAX host, the test results are evaluated by using the command file DIFFILE.COM to compare the generated file, FLANTST.SAV with the file, FLANUTP.SAV in IISS Configuration Management. The only differences should be the time and date stamps on the IISS Function Screen.

No scripting ability is available for the IBM host.

4.4.1 Test Evaluation Stages

There are several stages in the testing of FLAN.

Forms Test:

Stage 1: Input the file FLAN1.FDL to FLAN. This will produce the FD files TESTFORM, I1OHELP, FAT1 and FAT2 in the NTM directory.

Stage 2: Run ARTEST from the IISS Function Screen and add the form TESTFORM to screen. This will produce a screen like the one in Figure 5-7. Then terminate ARTEST and restart it. Follow the scenario shown in Figures 5-9a through 5-32d.

Stage 3: Input the file FLAN2.FDL to FLAN. This will produce the error messages listed in Section 5.1.2.3.

Graph Test:

Stage 1: Input the file GRAFTEST.FDL to FLAN. This will produce the necessary FD files in the tester's directory.

Stage 2: Input the file GRAFDE.FDL to RAP. This will produce the necessary FD files in the tester's directory.
Stage 3: Compile, link and sysgen the applications GRAFDE., and GRAFTST.

Stage 4: The test results are evaluated by comparing the information returned on the various output screens with that specified as successful for the given test. As outlined in section 5, each test of GDL functionality provides a screen with the output for a successful test. The data necessary for input is done automatically before the output screen. The only differences found should be the date and time stamps on the IISS Function Screen (Figure 5-3) and the first test output screen (Figure C-1).

2-D Graphics Test

Stage 1: Input the file ICONTST.FDL to RAP. This will produce the necessary FD file in the tester's directory.

Stage 2: Compile, link, and sysgen the application ICONTST.

Stage 3: The test results are evaluated by executing the application ICONTST and comparing the information returned on the various output screens with the appropriate output as specified in Section 5.
SECTION 5
TEST PROCEDURES

5.0 Test Procedures

The Form Processor Unit Test Plan consist of the following three test cases:

- Forms (with APPEARS IF)
  Section 5.1 on page 5-1

- Business Graphics (Pie, Bar, Line charts)
  Section 5.2 on page 5-125

- Graphics on Forms (Icons, 2-D graphics)
  Section 5.3 on page 5-131

5.1 Forms Test Description

This test uses the test program ARTEST and two FDL source files. FLAN1.FDL defines a form with correct syntax and semantics to test all FLAN features and FLAN2.FDL defines a form that tests all semantic errors. ARTEST is used to test the APPEARS IF syntax of the language.

5.1.1 Forms Test Control

As outlined, this unit test is a manual test which may be done by anyone. The required input data are documented for each function being tested and the resulting successful output is also documented. The order of the testing is also completely documented. The test control information is completely described in Section 5.1.2. Verification of the test is by a manual comparison of the test output with the expected results as they are documented here.

5.1.2 Forms Test Procedures

To run the unit test, you must be logged on to an IISS account. The NTM must be up and running and the UI symbolic names IISSFLIB, IISSULIB and IISSMLIB must be defined as described in the host specific sections.
5.1.2.1 VAX Test Procedures

The IISSFLIB, IISSULIB and IISSMLIB symbolic names must be defined as logicals at the group level. IISSFLIB and IISSULIB should point to the directory containing the production form definitions (FD files). IISSMLIB should point to the directory containing the error messages (MSG files).

Assuming the NTM is up and running, an IISS user may start the test by accessing the IISS environment with scripting as follows:

```
$ SET DEF <to directory containing NTM environment>
$ VT100 -RFLANUTP.SCP -SFLANTST.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 test begins executing on the terminal. The results of this test are saved in the current directory in the file FLANTST.SAV. To execute the test manually, enter only VT100 at the second '$' and enter the data as shown in the following sections.

This brings up the IISS Logon Screen which must be filled in:

```
User ID: [Blank]
Password: [Blank]
Role: [Blank]
```

Figure 5-1 IISS Logon Screen

(1) USER ID is the identification name of the user, and is 1 to 10 alpha-numeric characters. USER ID is input as "MORENC".

(2) PASSWORD must be the password associated with the USER ID, and is 1 to 10 alpha-numeric characters. PASSWORD is input as "STANLEY".

(3) ROLE is any of the identifiers which are associated
with the USER ID, and is 1 to 10 alpha-numeric characters. It will be checked against functions and applications which are selected by the user. ROLE is input as "MANAGER".

When this form is correctly completed and the <ENTER> key is pressed, the form in Figure 5-2 is displayed.

5.1.2.2 Choosing the FLAN Function

Specific IISS functions are accessed through the form displayed in Figure 5-2.

![IISS Function Screen](image)

When the form appears, the cursor is located in the Function field. The items in the form are summarized below:

1. DATE contains the current date. This may not be changed by the user.
2. TIME contains the current time. This may not be changed by the user.
3. USER ID is the user's identification that was entered in the previous form. This may not be changed by the user.
4. ROLE is the currently active role and was entered in the previous form. This may be changed at any time.
5. FUNCTION is the function the user desires to activate.
In the Function field enter FLAN. The screen in Figure 5-3 is displayed.

![FLAN screen](image)

Figure 5-3 FLAN screen

In the input field type "FLAN1.FDL" (a copy of FLAN1.FDL must be in the NTM directory) and press the <ENTER> key. Wait for the IISS Function Screen to return. Next type in ARTEST in the Function field as shown in Figure 5-4.

![ARTEST screen](image)

Figure 5-4 Starting ARTEST Application
Figure 5-5 shows the initial ARTEST screen.
In the command line type "ADDFRM SCREEN TESTFORM" as shown in Figure 5-6 and press the <ENTER> key.

Figure 5-6 Add Testform to Screen
Compare the appearance of the screen with Figure 5-7.

![Figure 5-7 FLAN1.FDL/TESTFORM Screen](image)

When finished viewing, press the <QUIT> key to terminate ARTEST. When the IISS Function Screen is displayed, enter ARTEST in the Function field to redisplay the initial ARTEST screen shown in Figure 5-5. The screens shown in section 5.1.4 describe the remainder of this unit test.

5.1.2.3 Standalone Version of FLAN on VAX

The standalone version of FLAN is invoked by typing "run [flandir]/flansa.exe" where [flandir] is the IISS production directory containing the FLAN executable. When the "args:" prompt appears type "[formdir]/flan2.fdl" where [formdir] is the directory containing the FLAN2.FDL file. The messages printed should be identical to the following list.

- **6:** ERROR - must specify relative field name
- **13:** ERROR - size not specified or invalid
- **18:** ERROR - value too big for field
- **22:** ERROR - no display attribute specified
- **64:** ERROR - unterminated string
- **66:** ERROR - value too big for field
- **70:** WARNING - string too long
- **72:** ERROR - duplicate field name: J
- **81:** ERROR - duplicate display attribute specified
- **86:** ERROR - unknown display attribute: UGLY
- **88:** ERROR - no display attribute specified
- **92:** ERROR - domain only legal for items
- **98:** ERROR - duplicate justification specified
- **104:** ERROR - duplicate case specified
- **110:** ERROR - duplicate minimum specified
- **116:** ERROR - duplicate maximum specified
- **122:** ERROR - help only legal for items

5-7
Partitioned datasets must be allocated for each of the following symbolic names: iisssslib, iissflib, iissmlib and iissulib. 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 Variable blocked with LRCL 80, BLKSIZE 3120, and 15 tracks with 15 directory blocks.
  iissulib Fixed block with LRECL 73, BLKSIZE 730, and 3 TRACKS with 2 directory blocks.
```

Assuming the NTM is up and running, an IISS user may start this test by accessing the IISS environment. To do this, enter "IISSI" at the ENTER APPLICATION: prompt. The "i" following IISS must be your IISS instance id as entered in the NTM SYSGEN file. This starts up the IBM3270 device driver and brings up the IISS Logon Screen as described in section 5.1.2.1.

When this screen is filled in correctly and the <RETURN> key is pressed, the screen in Figure 5-2 is displayed.
5.1.3.1 Choosing the FLAN Function

In the Function field type "FLAN". The screen shown in Figure 5-3 is displayed. In the input field type "flan1" (flan1 must be a member of the partitioned dataset referenced by the ddname IISSSLIB) and press the <RETURN> key. Wait for the IISS Function Screen to return. Next type "ARTEST" in the Function field as shown in Figure 5-4.

The initial ARTEST screen is shown in Figure 5-5. In the command line, type "ADDFRM SCREEN TESTFORM" as shown in Figure 5-6 and press the <RETURN> key. Compare the appearance of the resulting screen with Figure 5-7.

5.1.3.2 Standalone Version of Flan on IBM

The dataset associated with the ddnames iissulib, iiisslib, iiissmlib, iiissflib must be allocated previous to the call to FLANSA. The following clist describes how to invoke the standalone version of FLAN. The name of the file that is to be flanned should be passed in the clist.

```
PROC 0 FILE(.) LOAD(TIISS.R22.LOADLIB)
/*
/* This clist opens the required datasets for standalone FLAN.
/* The variable LOAD references the partitioned dataset containing the executable member FLAN. The variable file references the file to be flanned. It should be noted that even though FLAN will open the dataset as either a ddname or a member of a ddname, the letter method is not reached since we allocate the file member as a ddname.
/*
IF .&FILE EQ . THEN EXIT
CONTROL NOFLACH NOMSG
ALLOCATE DDN(SYSPRINT) DSN(*) SHR
ALLOCATE DDN(SYSTEM) DSN(*) SHR
ALLOCATE DDN(TIISSMLIB) DSN('TIISS.R22.MSG') SHR REUSE
ALLOCATE DDN(TIISSFLIB) DSN('TIISS.R22.FORMS.FD') SHR REUSE
ALLOCATE DDN(IISSULIB) DSN('SDREJ.FORMS.FD') SHR REUSE
ALLOCATE DDN(&FILE) DSN('SDREJ.FDRMS.FDL(&FILE)') SHR REUSE
CALL '&LOAD(FLANSA)' '&FILE'
```
5.1.3.1 Choosing the FLAN Function

In the Function field type FLAN. The screen shown in Figure 5-3 is displayed. In the input field type "flanl" (flanl must be a member of the partitioned dataset referenced by the ddname IISSSLIB) and press the <RETURN> key. Wait for the IISS Function Screen to return. Next type "ARTEST" in the Function field as shown in Figure 5-4.

The initial ARTEST screen is shown in Figure 5-5. In the command line, type "ADDFRM SCREEN TESTFORM" as shown in Figure 5-6 and press the <RETURN> key. Compare the appearance of the resulting screen with Figure 5-7.

5.1.3.2 Standalone Version of FLAN on IBM

The dataset associated with the ddnames iissulib, iisssslib, iissmlib, iissflib must be allocated previous to the call to FLANSA. The following clist describes how to invoke the standalone version of FLAN. The name of the file that is to be flanned should be passed in the clist.

PROC 0 FILE(.) LOAD(TIIS.S.R22.LOADLIB)
/*
/* This clist opens the required datasets for standalone FLAN. /*
/* The variable LOAD references the partitioned dataset /*
/* containing the executable member FLAN. The variable file /*
/* references the file to be flanned. It should be noted that /*
/* even though FLAN will open the dataset as either a ddname or /*
/* a member of a ddname, the letter method is not reached since /*
/* we allocate the file member as a ddname. /*
/*
IF .&FILE EQ . THEN EXIT
CONTROL NOFLACH NOMSG
ALLOCATE DDN(SYSPRINT) DSN(*) SHR
ALLOCATE DDN(SYSTEM) DSN(*) SHR
ALLOCATE DDN(IISSMLIB) DSN('TISS.R22.MSG') SHR REUSE
ALLOCATE DDN(IISSFLIB) DSN('TISS.R22.FORMS.FD') SHR REUSE
ALLOCATE DDN(IISSULIB) DSN('SDREJ.FORMS.FD') SHR REUSE
ALLOCATE DDN(&FILE) DSN('SDREJ.FDRMS.FDL(&FILE)') SHR REUSE
CALL '&LOAD(FLANSA)' '&FILE'
Figure 5-9a Test Case 1

CRITERION: Item I3 APPEARS IF $2 > 1$
Figure 5-9b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-10a  Test Case 2

CRITERION:  Item I3 APPEARS IF 2 < 1
Figure 5-10b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-11a  Test Case 3

CRITERION: Item I3 APPEARS IF I1 != 1
Figure 5-11b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Set the value of I1 to "1".

Figure 5-11c Change I1 Value
Figure 5-11d  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Set the value of \( I_1 \) to "3".

Figure 5-11e  Change \( I_1 \) Value
Item I3 appears because the criterion evaluates to true.
CRITERION: Item I3 APPEARS IF BETWEEN('il', 1, 10)
Il is defined to be numeric for this test.
Figure 5-12b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-12c Change I1 Value

Set the value of I1 to " 5".
Item I3 appears because the criterion evaluates to true.
Figure 5-12e Change I1 Value

Set the value of I1 to "a".
Figure 5-12f  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-13a  Test Case 5

CRITERION: Item I3 APPEARS IF IN('i1', 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
I3 is defined as numeric for this test.
Figure 5-13b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Set the value of I1 to "a".

Figure 5-13c  Change I1 Value
Figure 5-13d  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-13e  Change I1 Value

Set the value of I1 to "5".
Figure 5-13f  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-14a Test Case 6

CRITERION: Item I3 APPEARS IF 'Il' > 10 ? 1 : 0
Figure 5-14b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Set the value of I1 to "15".
Figure 5-14d  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-14e Change I1 Value

Set the value of I1 to "-1".
Figure 5-14f I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-15a  Test Case 7

CRITERION:  Item I3 APPEARS IF 'II' > "CCC" ? 1 : 0
Figure 5-15b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-15c Change Il Value

Set the value of Il to "CCD".
Figure 5-15d  I3 Appears

Item I3 appears because the criterion evaluates to true.
Set the value of I1 to "CCB".

Figure 5-15e Change I1 Value
Figure 5-15f  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-16a Test Case 8

CRITERION: Item I3 APPEARS IF NOT 'I1'
Figure 5-16b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Set the value of $I_1$ to "yes".

Figure 5-16c  Change $I_1$ Value
Item I3 does not appear because the criterion evaluates to false.
Figure 5-16e Change I1 Value

Set the value of I1 to "0".
Figure 5-16f  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-17a Test Case 9

CRITERION: Item I3 APPEARS IF NOT APPEARS('Il')
Figure 5-17b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
CRITERION: Item I3 APPEARS IF APPEARS('ff28.il')
Figure 5-18b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-19a Test Case 11

CRITERION: Item I3 APPEARS IF 'I1' <= 0 OR 'I1' >= 10
Item I3 appears because the criterion evaluates to true.
Figure 5-19c Change Il Value

Set the value of Il to "5".
Figure 5-19d  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-19e Change I1 Value

Set the value of I1 to "I1".
Item I3 appears because the criterion evaluates to true.
Figure 5-20a Test Case 12

CRITERION: Item I3 APPEARS IF 'II' != 11 AND 'II' >= 10
Figure 5-20b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-20c Change Il Value

Set the value of Il to "12".
Figure 5-20d  I3 Appears

Item I3 appears because the criterion evaluates to true.
Set the value of I1 to "11".
Figure 5-20f  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-21a Test Case 13

CRITERION: Item I3 APPEARS IF GWINDO('w3') > 1
Figure 5-21b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-21c  Create Page 2 in W3
Figure 5-21d  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-21e Remove Page 2 in W3
Figure 5-21f  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
CRITERION: Item I3 APPEARS IF CURSOR('i2')
Figure 5-22b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.

Move the cursor to Item I2 and press the <ENTER> key.
Figure 5-22c  I3 Appears

Item I3 appears because the criterion evaluates to true.
Move the cursor out of Item I2 and press the <ENTER> key.
Figure 5-22d  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-23a Test Case 15

CRITERION: Item I3 APPEARS IF GPAGE('w3', 1) = "ff3"
Figure 5-23b  I3 Does Not Appear in W1

Item I3 does not appear because the criterion evaluates to false.
Replace form ff35 on Page 1 of W3 with ff3.
Figure 5-23d  I3 Appears in W1

Item I3 appears because the criterion evaluates to true.
Figure 5-23e  Change Contents of Page 1 in W3

Replace ff3 in Page 1 of W3 with ff17.
Figure 5-23f  I3 Does Not Appear in W1

Item I3 does not appear because the criterion evaluates to false.
Figure 5-24a Test Case 16

CRITERION: Item I3 APPEARS IF GETATT ('Il', 0) != "INPUT"
Figure 5-24b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
CRITERION: Item I3 APPEARS IF NOT GETATT('I1', 0) != "INPUT"
Figure 5-25b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-26a Test Case 18

CRITERION: Item I3 APPEARS IF NOT ROLE("manager")
Figure 5-26b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-27a Test Case 19

CRITERION: Item I3 APPEARS IF
IN(BETWEEN('I1', 1, 10), 1, 2, 3, 4)
I3 is defined as numeric for this test.
Figure 5-27b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Set the value of I1 to "5".
Item I3 appears because the criterion evaluates to true.
Figure 5-27e Change I1 Value

Set the value of I1 to "11".
Figure 5-27f  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
CRITERION: Item I3 APPEARS IF
"CCC")], 1, 2, 3, 4)
Figure 5-28b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Set the value of I1 to "CCD".
Figure 5-28d  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Set the value of II to "BBB".
Figure 5-28f  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-29a Test Case 21

CRITERION: Item I3 APPEARS IF ('Il' >= 1) ?
((('Il' = 5) ? 1 : 0) : 1)
Figure 5-29b  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-29c  Change I1 Value

Set the value of I1 to "2".
Figure 5-29d  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-29e  Change I1 Value

Set the value of I1 to "5".
Figure 5-29f  I3 Appears

Item I3 appears because the criterion evaluates to true.
CRITERION: Item I3 APPEARS IF NOT IN((('I1' >= 1) ? (('I1' = 5) ? 1 : 0) : 1), 1, 2, 3)
Figure 5-30b  I3 Does Not Appear

Item I3 does not appear because the criterion evaluates to false.
Figure 5-30c Change II Value

Set the value of II to "2".
Figure 5-30d  I3 Appears

Item I3 appears because the criterion evaluates to true.
Figure 5-30e Change I1 Value

Set the value of I1 to "5".
Item I3 does not appear because the criterion evaluates to false.
The following test case tests the APPEARS IF criterion for the form field F1.

CRITERION: Form F1 APPEARS IF 'I1' > 10 ? 1 : 0
Figure 5-31b F1 Does Not Appear

Form F1 does not appear because the criterion evaluates to false.
Set the value of I1 to "15".
Form F1 appears because the criterion evaluates to true.
Figure 5-31e Change Il Value

Set the value of Il to "-1".
Form F1 does not appear because the criterion evaluates to false.

Figure 5-31f  F1 Does Not Appear
The following test case tests the APPEARS IF criterion for the window field W1.

CRITERION: Window W1 APPEARS IF 'ff1.ff6.i1' < "1111"
Set the value of FF6.I1 to "xxxx".
Figure 5-32c Change FF6.I1 Value

Set the value of FF6.I1 to blank.
Window W1 appears because the criterion evaluates to true.

This concludes the tests for the APPEARS IF syntax. Press the <QUIT> key twice to terminate ARTEST and return to the system prompt.
5.2 Graph Test Description

Two test programs are used to test the Graph definition Language (GDL). The test program GRFTST uses explicit FP calls to place the form within a window, place data within the form fields where the graph data is to be located, display the graph form, and remove the graph form from the window. Since the program issues the pdata using data from internal arrays, no data entry is required by the tester.

The test program GRAPDE is an interactive application that is generated using the Rapid Application Generator. GRAPDE represents user entered data as a pie, bar, or line graph.

5.2.1 Graph 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. Accurate observation of the resulting successful output must be made to ensure the unit test was done properly.

5.2.2 Graph Test Procedures

To run the unit test, you must be logged on to an IISS account. The NTM must be up and running and the UI symbolic names IISSFLIB, IISSULIB and IISSMLIB must be defined as described in the host specific sections. For the IBM version, please reference Section 5.1.3.

5.2.2.1 Graph VAX Test Procedures

To run the unit test plan in the VAX/VMS environment as outlined below, one must be logged onto an IISS account. The NTM must be up and running and the UI logical names IISSFLIB, IISSULIB, and IISSMLIB must be set properly at the 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). IISSMLIB points to the directory containing the user's form definition source files (FDL files). IISSMLIB points to the directory containing the UI error and help messages (MSG files). To perform this test IISSULIB and IISSMLIB must be pointing to the default directory.

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

$ SET DEF <to directory containing NTM environment>
$ TEK4100

These commands start up the TEK4100 device driver.
5.2.2.1.1 *Access to GDL Test Programs*

Following entry of the system command "TEK4100" which activates the User Interface the following form appears:

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

Figure 5-33 IISS Logon Screen

(1) USER ID is the identification name of the user, and is 1 to 10 alpha-numeric characters. USER ID is input as "MORENC".

(2) PASSWORD must be the password associated with the USER ID, and is 1 to 10 alpha-numeric characters. PASSWORD was input as "STANLEY".

(3) ROLE is any of the identifiers which are associated with the USER ID, and is 1 to 10 alpha-numeric characters. It will be checked against functions and applications which are selected by the user. ROLE is input as "MANAGER".

When this form is correctly completed and the <ENTER> key is pressed, the IISS Function Screen is displayed.
Figure 5-34 IISS Function Screen

When this form appears, the cursor is located in the input field labeled FUNCTION. The items in the form are summarized below:

1. DATE contains the current date. This may not be changed by the user.
2. TIME contains the current time. This may not be changed by the user.
3. USER ID is the user's identification that was entered in the previous form. This may not be changed by the user.
4. ROLE is the currently active role and was entered in the previous form. This may be changed at any time.
5. FUNCTION is the function the user desires to activate.

To run the GDL test programs, proceed as described in the following sections.
5.2.2.1.2 Running the GRFTST Program

To run the GRFTST program, enter "GRFTST" in the FUNCTION field on the IISS Function Screen and press the <ENTER> key. This program produces the 27 graphs shown in Appendix C. Test Graph A is displayed when the program begins. Each succeeding graph is displayed by repeatedly pressing the <ENTER> key. Before proceeding to the next graph, the graph displayed on the terminal screen should be compared with the corresponding graph in Appendix C. When all 27 graphs have been displayed and compared, a final press of the <ENTER> key terminates the program and redisplay the IISS Function Screen.

5.2.2.1.3 Running the GRAFDE Program

To run the GRAFDE program, enter "GRAFDE" in the FUNCTION field on the IISS Function Screen and press the <ENTER> key. The following screen is displayed.

```
ENGINEERING CHANGE ANALYSIS CENTER
CHANGE COST GRAPH

DATE: 6/14/88
TIME: 10:54:28


Enter data and press <PF5> - Pie, <PF6> - Bar, <PF7> - Line

Figure 5-35 Initial GRAFDE Screen
```
Enter the data as shown in Figure 5-36 and press the appropriate function key to produce the desired graph as described in Table 5-1.

<table>
<thead>
<tr>
<th>PFKEY</th>
<th>APPENDIX FIGURE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>B-1</td>
<td>Pie chart with percentages outside</td>
</tr>
<tr>
<td>6</td>
<td>B-2</td>
<td>Horizontal bar chart</td>
</tr>
<tr>
<td>7</td>
<td>B-3</td>
<td>Line graph</td>
</tr>
<tr>
<td>9</td>
<td>B-4</td>
<td>Pie chart with percentages inside and labels</td>
</tr>
<tr>
<td>10</td>
<td>B-5</td>
<td>Vertical bar chart</td>
</tr>
<tr>
<td>11</td>
<td>B-6</td>
<td>Line chart with area under curves shaded</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Quit application</td>
</tr>
</tbody>
</table>

Table 5-1 PFKEY and Figure Correlation

The screens displayed should be compared with the indicated graph figures in Appendix D. Only the time/date stamps should differ.

5.3 2-D Graphics Test Description

A test program is used to test the 2-D graphics on forms (icons). The test program ICONST, written in FDL, is an application that is generated using the Rapid Application Generator. ICONST will display an icon, a graphics form with all of the supported 2-D graphics primitives, and allow the scrolling of graphics on a form within a window.
5.3.1 2-D Graphics 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. Accurate observation of the resulting successful output must be made to ensure the unit test was done properly.

5.3.2 2-D Graphics Test Procedures

To run the unit test, you must be logged on to an IISS account. The NTM must be up and running and the UI symbolic names IISSFLIB, IISSULIB and IISSMLIB must be defined as described in the host specific sections. For the IBM version, please reference Section 5.1.3.

5.3.2.1 2-D Graphics VAX Test Procedures

To run the unit test plan in the VAX/VMS environment as outlined below, one must be logged onto an IISS account. The NTM must be up and running and the UI logical names IISSFLIB, IISSULIB, IISSLINLIB, and IISSMLIB must be set properly at the 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). IISSLINLIB points to the directory containing the user's form definition source files (FDL files). IISSMLIB points to the directory containing the UI error and help messages (MSG files). To perform this test IISSULIB and IISSLINLIB must be pointing to the default directory.

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

\$ SET DEF <to directory containing NTM environment>
\$ TEK4100

These commands start up the TEK4100 device driver.
5.3.2.1.1 Access to 2-D Graphics Test Programs

Following entry of the system command "TEK4100" which activates the User Interface the following form appears:

<table>
<thead>
<tr>
<th>USER ID:</th>
<th>PASSWORD:</th>
<th>ROLE:</th>
</tr>
</thead>
</table>

Figure 5-37 IISS Logon Screen

(1) USER ID is the identification name of the user, and is 1 to 10 alpha-numeric characters. USER ID is input as "MORENC".

(2) PASSWORD must be the password associated with the USER ID, and is 1 to 10 alpha-numeric characters. PASSWORD was input as "STANLEY".

(3) ROLE is any of the identifiers which are associated with the USER ID, and is 1 to 10 alpha-numeric characters. It will be checked against functions and applications which are selected by the user. 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 2.3 --------

DATE: ___/__/___  TIME: ___:___  USER ID: ______  ROLE: ______
FUNCTION: _________  DEVICE TYPE: _________  DEVICE NAME: _______

Msg: 0

Figure 5-38 IISS Function Screen
When this form appears, the cursor is located in the input field labeled FUNCTION. The items in the form are summarized below:

(1) DATE contains the current date. This may not be changed by the user.

(2) TIME contains the current time. This may not be changed by the user.

(3) USER ID is the user's identification that was entered in the previous form. This may not be changed by the user.

(4) ROLE is the currently active role and was entered in the previous form. This may be changed at any time.

(5) FUNCTION is the function the user desires to activate.

To run the 2-D graphics test programs, proceed as described in the following sections.
5.3.2.1.2 Running the INCONTST Program

ACTIVITY A:

To run the INCONTST program, enter "ICONTST" in the FUNCTION field on the IISS Function Screen and press the <ENTER> key. Figure 5-39 is displayed when the program begins.

Figure 5-39  ICONTST startup screen
ACTIVITY B:

Input the data shown in figure 5-40 into the appropriate fields as shown.

![Figure 5-40 Forms Data Input screen](image)

**Figure 5-40** Forms Data Input screen
ACTIVITY C:

Move the cursor to the ICON labeled "Mol." (Molecule) and pick the ICON by pressing the <ENTER> key. The form in Figure 5-41, will be partially displayed in the Graphics Window (Figure 5-42). With the mode key, select the window manager function. Place the cursor into the Graphics Window and press the Select key. The Graphics Window may now be scrolled to show the remainder of Figure 5-41 using the scroll window keys.

Figure 5-41  Molecule Graphics form
Figure 5-42 Example display of Molecule Graphics
ACTIVITY D:

Move the cursor to the ICON labeled "Sales" and pick the ICON by pressing the <ENTER> key. The form Figure 5-43, will be partially displayed in the Graphics Window (Figure 5-44). Using the Window manager function as in Activity C, scroll the Graphics Window to show the remainder of Figure 5-43.

Figure 5-43 Sales Graphs from 5-136
Figure 5-44 Example display of Sales Graphs

This completes the ICONS program to exit, place the cursor into the "exit" ICON, and press the <ENTER> key.
APPENDIX A

FLAN1.FDL

FLAN input file with correct syntax and semantics to test all features.

/* test forms for the flan compiler */

create form testform
    prompt center at 2 below i3 and column 6 "array" /* form prompt */

item i1 /* item field */
    size 1 at 2 3 /* field size */
    display as input /* field display attribute */
    value "1" /* item value */
    prompt at 1 2 "size/display" /* field prompt */

item i2
    size 2 by 2
    at below i1
    display as output
    value "2222"

item i3
    size 3 by 2
    at below i2
    display as text
    value "333333"

item i4 (2 v, 3 h 2, 2 v 2) /* array */
    at 3 below i3 and column 2
    size 1
    display as input

form fat1 /* forms */
    at 3 below i3 and column 12
    size 6
    prompt at above "forms"

form fat2
    at below fat1
    size 6
window w1 /* windows */
  at 3 right of fat1
  size 5
  background white
  prompt at above "windows"

window w2
  at below w1
  size 5
  background black

item i5 /* domains */
  at 2 15
  size 2
  display as input
  domain (must enter must fill numeric)
  prompt at right "(must enter must fill numeric)"
  prompt at 1 above and col 15 "domains"

item i6
  at below i5
  size 4
  display as input
  domain (left lower)
  prompt at right "(left lower)"

item i7
  at below i6
  size 4
  display as input
  domain (right upper)
  prompt at right "(right upper)"

item i8
  at below i7
  size 4
  display as input
  domain (max 10 min 0)
  prompt at right "(max 10 min 0)"
item i9
   at 2 60
   /* help */
   display as input
   help "help message for i9"
   prompt at right "message"
   prompt at 1 above "help"

item i10
   at below i9
   display as input
   help i10help
   prompt at right "form"

item 10
   at 13 40
   /* location tests */
   display as input
   prompt at 4 above "location"

item 11
   size 2 by 2
   display as input
   bottom right at 1 above 10 and 2 left of 10
   value "1111"

item 12
   size 2 by 2
   display as input
   bottom left at 2 above top left of 10
   value "1212"

item 13
   size 2 by 2
   display as input
   bottom right at above top right of 10
   value "1313"

item 14
   size 2 by 2
   display as input
   bottom left at 1 above 10 and 2 right of 10
   value "1414"
item 15
  size 2 by 2
display as input
top right at left of top left of 10
  value"1515"

item 16
  size 2 by 2
display as input
  bottom right at 2 left of bottom left of 10
  value"1616"

item 17
  size 2 by 2
display as input
top right at 1 below 10 and 2 left of 10
  value"1717"

item 18
  size 2 by 2
display as input
top left at below bottom left of 10
  value"1818"

item 19
  size 2 by 2
display as input
top right at 2 below bottom right of 10
  value"1919"

item la
  size 2 by 2
display as input
top left at 1 below 10 and 2 right of 10
  value"lala"

item lb
  size 2 by 2
display as input
top left at 2 right of top right of 10
  value"lb1b"
item lcl
  size 2 by 2
display as input
  bottom left at right of bottom right of 10
  value"lclc"

create form i10help
  size 80 by 23
  prompt center at 10 40 "help form form item i10"

create form fat1     /* form background and size */
  background white
  size 5
  prompt at 1 2 "fat1"

create form fat2
  background black
  size 5
  prompt at 1 2 "fat2"
APPENDIX B

FLAN2.FDL

FLAN input file to test all semantic error messages.

/* flan forms to force all semantic error messages */

create form testerr
size 1 /*("form %s too narrow: fields extend to column %d",
("form %s too short: fields extend to row %d",
prompt at left "testerr" /*("must specify relative field
name");*/
prompt at 25 2 "off bottom"

item a /*("size not specified or invalid");*/
at 1 2
display as input

item b /*("value too big for field");*/
size 1
value "22"
display as input

item cc /*("no display attribute specified");*/
size 1
at 1 4

item bb /*("field %s referenced in %s %s% not defined",*/
at below nothing
size 1
display as input

item cd /*("circular reference in location of %s %s% not defined",*/
at below dc
size 1
display as input

item dc
at above cd
size 1
display as input

item c /*("overlap between %s %s% and %s %s%",*/
at 2 2
size 1
display as input
item f
at 2 2
size 1
display as input

item g /*("%s %s %s off top of screen","*/
at -1 10
size 1
display as input

item hh /*("%s %s %s off left of screen","*/
at 1 -1
size 1
display as input

item i /*("unterminated string");*/
at 1 6
size 1
display as input
value "hello"

item j /*("string too long");*/
at 1 8
size 150
display as input
value "12345678911234567892123456789312345678941234567895123456789612/34567897123/4567898123456789912345678901234567891123456789212345678931234567894"

item j /*("duplicate field name: %s","*/
at 3 2
size 1
display as input

item k /*("duplicate display attribute specified");*/
at 3 4
size 1
display as input
display as input

item l /*("unknown display attribute: %s","*/
at 3 6
size 1
display as ugly
window m at 3 8
size 1
display as black
domain (upper) /*("domain only legal for items");*/

item n /*("duplicate justification specified");*/
at 3 10
size 1
display as input
domain (left right)

item o /*("duplicate case specified");*/
at 3 12
size 1
display as input
domain (left right)

item p /*("duplicate minimum specified");*/
at 3 14
size 1
display as input
domain (min 10 min 2)

item q /*("duplicate maximum specified");*/
at 3 16
size 1
display as input
domain (max 10 max 2)

window rr /*("help only legal for items"); */
at 3 18
size 1
display as black
help "hello"

create form testform
item s /*("duplicate help specified");*/
at 3 20
size 1
display as input
help "hello"
help "hello"
item t /*("help message too long, truncated")*/
at 3 22
size 1
display as input
help
"12345678911234567892123456789312345678941234567895123456789612
34567897"

window u /*("value only legal for items")*/
at 3 24
size 1
display as black
value "hello"

create form testform
item vv /*("duplicate value specified")*/
at 3 26
size 10
display as input
value "hello"
value "hello"

item w /*("unknown function %s", */
at 3 38
size 20
display as input
value func('hello')

item x /*("invalid argument for INDEX")*/
at 3 60
size 20
display as input
value index(1)

item y /*("duplicate size specified")*/
at 4 2
size 1
size 1
display as input

create form endless
/* else if (c == EOF) (fatal("unterminated comment"); return
  c;)

B-4
APPENDIX C
SCREENS AND GDL FOR GRFTST

This appendix contains all the screens for the first test of the Graph Definition Language. The necessary FDL follows the screens.
Figure C-1 GDL Test Activity A: and corresponding GDL
create form grftstl
size 80 by 30
prompt at 2 20 "ENGINEERING CHANGE ANALYSIS SYSTEM -"
prompt at 3 30 "CHANGE COST GRAPH"
attribute out (background white, display blue, guarded, nowrite)
attribute out2 (background blue, display yellow, guarded)
attribute out3 (background red, display black, guarded)
attribute nosee (guarded, hidden)

item curdat
at 2 68
size 8
prompt at 2 62 "DATE:"
value '.date'
display as out

item curtim
at 3 68
size 8
prompt at 3 62 "TIME:"
value '.time'
display as out

item anlid
at 5 10
size 3
prompt at 5 2 "ANL ID:"
display as out2

item chgid
at 6 10
size 6
prompt at 6 2 "CHG ID:"
display as out2

item chgdsc
at 6 28
size 50
prompt at 6 18 "CHG DESC:"
display as out3

item eleid
at 7 10
size 3
prompt at 7 2 "ELE ID:"
display as out2

item eleid
at 7 28
size 50
prompt at 7 18 "ELE DESC:"
display as out3
graph cstgrf
    at 9 2
    display as blue
    size 60 by 21

form csttab
    at 9 65
    display as black
    size 16 by 21

create bar graph cstgrf
    using ('csttab.ids' axis ax1)
    attribute a line (display yellow)
    attribute b prompt (display white)
    attribute c prompt (display green)
    attribute d prompt (display red)
    legend at 2 2
    label display as d, at 20 2 "DISPOSITION = SCRAP + REWORK"

curve rework
    'csttab.rewcst' using axis ax2
    legend c "rework"
    absolute

curve misc
    'csttab.msccst'
    additive using curve scrap
    legend c "misc."

curve scrap
    'csttab.scrcst'
    additive using curve rework
    legend c "scrap"
axis ax1
  horizontal
display as a
  at 16 25
  min 0
  size 30
  label b " result id"
tick every 1 d " " "1" "2" "3" "4" "5" "6" "7" "8" "9" "10"

axis ax2
  at 16 25
  size 15
  label b " cost $ "
vertical
  min 0
  display as a

create form csttab
  prompt at 1 2 "RES."
prompt at 2 2 "ID"
prompt at 1 8 "EFF."
prompt at 2 8 "DATE"
  attribute hid (hidden, guarded)

item dates (10 v 0)
  size 8
  at 3 6
  display as magenta

item ids (10 v 0)
  size 3
  at 3 2
  domain (numeric)
  display as cyan

item mscst (10 v 0)
  size 6
  at 3 15
  display as hid
domain (numeric)

item scrcst (9 v 0)
  at 3 35
domain (numeric)
  display as hid
  size 6

item rewcst (5 v 0)
  at 3 45
  size 6
domain (numeric)
  display as hid
Figure C-2 GDL Test Activity b: and corresponding GDL
create form grftst2
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grfl
  at 1 15
  size 60 by 20
  display as blue

create line graph grfl
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  background blue

curve aaa
  'grftst2.i1' using axis ax2
  absolute

curve two
  'grftst2.i2'
  additive using curve aaa

axis ax1
  horizontal
  display as x
  at 15 30
  size 30
axis ax2
at 15 30
size 15
vertical
display as x
Figure C-3 GDL Test Activity C: and corresponding GDL

STS: 0
create form grftst3
  size 80 by 30

form fgrf (3 v 0)
  at 1 1
  size 5 by 1

graph grf2
  at 1 1_
  size 60 by 20
  display as blue

create pie graph grf2
  at 10 30
  size 20 by 8
  using ('grftst3.fgrf(*).il')

pie 1
  shade color red

pie 2
  shade color magenta

pie 3
  shade color white

create form fgrf

item il
  display as red
  at 1 2
  size 3
  domain (numeric)
Figure C-4 GDL Test Activity D: and corresponding GDL

STS: 0

appl.
create form grftst4
    size 80 by 30

item il (3 v 0)
    display as red
    at 1 2
    size 3
    domain (numeric)

graph grf3
    at 1 15
    size 60 by 20
    display as blue

create pie graph grf3
    at 10 30
    size 20 by 8
    using ('grftst4.il')

    pie 1
        shade color red

    pie 2
        shade color magenta
        explode 2

    pie 3
        shade color white
Figure C-5 GDL Test Activity E: and corresponding GDL
create form grftst5
size 80 by 30

item i1 (3 v 0)
display as red
at 1 2
size 3
domain (numeric)

item i2 (5 v 0)
display as yellow
at 1 6
size 3
domain (numeric)

graph grf4
at 1 15
size 60 by 20
display as blue

create line graph grf4
using (1, 2, 3, 4, 5 axis ax1)
attribute a fill (display cyan)
attribute b line (display magenta)
attribute xy prompt (display yellow)
attribute x line (display yellow)
attribute c prompt (display white)
attribute d line (display green)
background blue

curve aaa
'grftst5.i1' using axis ax2
absolute

curve two
'grftst5.i2'
additive using curve aaa

axis ax1
horizontal
display as x
at 18 2
size 30
label c "this is a label"
axis ax2
at 18 2
size 15
label c "this is a label"
vertical
display as x
Figure C-6 GDL Test Activity F: and corresponding GDL
create form grftst6  
    size 80 by 30  

item i1 (3 v 0)  
    display as red  
    at 1 2  
    size 3  
    domain (numeric)  

item i2 (5 v 0)  
    display as yellow  
    at 1 6  
    size 3  
    domain (numeric)  

graph grf5  
    at 1 15  
    size 60 by 20  
    display as blue  

create line graph grf5  
    using (1, 2, 3, 4, 5 axis ax1)  
    attribute a fill (display cyan)  
    attribute b line (display magenta)  
    attribute xy prompt (display yellow)  
    attribute x line (display yellow)  
    attribute c prompt (display white)  
    attribute d line (display green)  
    background blue  

curve aaa  
    'grftst6.i1' using axis ax2  
    absolute  

curve two  
    'grftst6.i2'  
    additive using curve aaa  

axis ax1  
    horizontal  
    display as x  
    label c "this is axis ax1"  
    at 15 30  
    size 30
axis ax2
  at 15 30
  size 15
  vertical
  display as x
  label xy "this is axis ax2"
Figure C-7 GDL Test Activity G: and corresponding GDL
create form grftst7
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf6
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf6
  at 10 2
  size 40 by 16
  using ('grftst7.il')

pie 1
  shade color red

pie 2
  shade color magenta

pie 3
  shade color white
  explode 20
Figure C-8 GDL Test Activity H: and corresponding GDL
create form grftst8
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf7
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf7
  at 2 2
  size 40 by 16
  using ('grftst8.il')

pie 1
  shade color yellow

pie 2
  shade color white

pie 3
  shade color red
  explode 20
Figure C-9 GDL Test Activity I: and corresponding GDL
create form grftst9
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf8
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf8
  at 10 30
  size 20 by 8
  using ('grftst9.il')

pie 1
  shade color red

pie 2
  shade color magenta
  explode 35

pie 3
  shade color white
Figure C-10 GDL Test Activity J: and corresponding GDL

STS: 0
create form grftst10
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf9
  at 1 15
  size 60 by 20
  display as blue

create bar graph grf9
  using (1, 2, 3, 4, 5) axis ax1
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  background blue

curve aaa
  'grftst10.i1' using axis ax2
  absolute

curve two
  'grftst10.i2'
  additive using curve aaa

axis ax1
  horizontal
  display as x
  at 15 30
  tick 5 1 c "A" "B" "C"
  size 30
axis ax2
   at 15 30
   size 15
   vertical
display as x
Figure C-11 GDL Test Activity K: and corresponding GDL

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>110</td>
<td></td>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0</td>
<td>1.5</td>
<td>2.0</td>
<td>2.5</td>
<td>3.0</td>
<td>3.5</td>
<td>4.0</td>
<td>4.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0</td>
<td>1.5</td>
<td>2.0</td>
<td>2.5</td>
<td>3.0</td>
<td>3.5</td>
<td>4.0</td>
<td>4.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

STS: 0

appl.
create form grftstll
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf10
  at 1 15
  size 60 by 20
  display as blue

create line graph grf10
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  legend at 2 2
  background blue

curve aaa
  'grftstll.i1' using axis ax2
  absolute
  legend xy "curve aaa"

curve two
  'grftstll.i2'
  additive using curve aaa
  legend xy "curve two"

axis ax1
  horizontal
  display as x
  at 15 30
  size 30

axis ax2
  at 15 30
  size 15
  vertical
  display as x
Figure C-12 GDL Test Activity L: and corresponding GDL

curve two
curve aaa

axis ax1
axis ax2

STS: 0
appl.

C-30
create form grftst12
    size 80 by 30

item i1 (3 v 0)
    display as red
    at 1 2
    size 3
    domain (numeric)

item i2 (5 v 0)
    display as yellow
    at 1 6
    size 3
    domain (numeric)

graph grf11
    at 1 15
    size 60 by 20
    display as blue

create line graph grf11
    using (1, 2, 3, 4, 5 axis ax1)
    attribute a fill (display cyan)
    attribute b line (display magenta)
    attribute xy prompt (display yellow)
    attribute x line (display yellow)
    attribute c prompt (display white)
    attribute d line (display green)
    legend at 2 2
    label display as c, at 15 2 "STUPID GRAPH"
    background blue

curve aaa
    'grftst12.i1' using axis ax2
    absolute
    legend xy "curve aaa"

curve two
    'grftst12.i2'
    additive using curve aaa
    legend xy "curve two"
axis ax1
  horizontal
display as x
at 15 30
size 30
label c "axis ax1"

axis ax2
  at 15 30
  size 15
  label c "axis ax2"
  vertical
display as x
Figure C-13 GDL Test Activity M: and corresponding GDL
create form grftstl3
    size 80 by 30

item i1 (3 v 0)
    display as red
    at 1 2
    size 3
    domain (numeric)

item i2 (5 v 0)
    display as yellow
    at 1 6
    size 3
    domain (numeric)

graph grf12
    at 1 15
    size 60 by 20
    display as blue

create line graph grfl2
    using (1, 2, 3, 4, 5 axis ax1)
    attribute a fill (display cyan)
    attribute b line (display magenta)
    attribute xy prompt (display yellow)
    attribute x line (display yellow)
    attribute c prompt (display white)
    attribute d line (display green)
    legend h at 2 2
    label display as c, at 15 2 "STUPID GRAPH"
    background blue

curve aaa
    'grftstl3.i1' using axis ax2
    absolute
    legend xy "curve aaa"

curve two
    'grftstl3.i2'
    additive using curve aaa
    legend xy "curve two"
axis ax1
  horizontal
  display as x
  at 15 30
  size 30
  label c "axis ax1"

axis ax2
  at 15 30
  size 15
  label c "axis ax2"
  vertical
  display as x
Figure C-14 GDL Test Activity N: and corresponding GDL
create form grftst14
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grfl3
  at 1 15
  size 60 by 20
  display as blue

create bar graph grf13
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  legend h at 2 2
  label display as c, at 15 2 "STUPID GRAPH"
  background blue

curve aaa
  'grftst14.i1' using axis ax2
  absolute
  legend xy "curve aaa"

curve two
  'grftst14.i2'
  additive using curve aaa
  legend xy "curve two"
axis ax1
  horizontal
display as x
at 15 30
size 30
label c "axis ax1"

axis ax2
  at 15 30
size 15
label c "axis ax2"
vertical
display as x
Figure C-15 GDL Test Activity 0: and corresponding GDL
create form grftst15
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf14
  at 1 15
  size 60 by 20
  display as blue

create bar graph grf14
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  legend at 2 2
  label display as c, at 15 2 "STUPID GRAPH"
  background blue

curve aaa
  'grftst15.i1' using axis ax2
  absolute
  legend xy "curve aaa"

curve two
  'grftst15.i2'
  additive using curve aaa
  legend xy "curve two"
axis ax1
   horizontal
display as x
at 15 30
size 30
label c "axis ax1"

axis ax2
   at 15 30
   size 15
   label c "axis ax2"
   vertical
display as x
Figure C-16 GDL Test Activity P: and corresponding GDL
create form grftst16
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf15
  at 1 15
  size 60 by 20
  display as blue

create line graph grf15
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  legend at 2 2 box
  label display as c, at 15 2 "STUPID GRAPH"
  background blue

curve aaa
  'grftst16.il' using axis ax2
  absolute
  legend xy "curve aaa"

curve two
  'grftst16.i2'
  additive using curve aaa
  legend xy "curve two"
axis ax1
    horizontal
display as x
at 15 30
size 30
label c "axis ax1"

axis ax2
    at 15 30
    size 15
    label c "axis ax2"
    vertical
    display as x
Figure C-17 GDL Test Activity Q: and corresponding GDL
create form grftstl7
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf16
  at 1 15
  size 60 by 20
  display as blue

create line graph grf16
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  legend h at 2 2 box
  label display as c, at 15 2 "STUPID GRAPH"
  background blue

curve aaa
  'grftstl7.i1' using axis ax2
  absolute
  legend xy "curve aaa"

curve two
  'grftstl7.i2'
  additive using curve aaa
  legend xy "curve two"
axis ax1
  horizontal
display as x
  at 15 30
  size 30
  label c "axis ax1"

axis ax2
  at 15 30
  size 15
  label c "axis ax2"
  vertical
display as x
Figure C-18 GDL Test Activity R: and corresponding GDL

STUPID GRAPH

axis ax1

axis ax2

10 20
30 40
50 60
70 80

curve two

curve aaa

STS: 0

appl.
create form grftstl8
size 80 by 30

item i1 (3 v 0)
display as red
at 1 2
size 3
domain (numeric)

item i2 (5 v 0)
display as yellow
at 1 6
size 3
domain (numeric)

graph grf17
at 1 15
size 60 by 20
display as blue

create bar graph grf17
using (1, 2, 3, 4, 5 axis ax1)
attribute a fill (display cyan)
attribute b line (display magenta)
attribute xy prompt (display yellow)
attribute x line (display yellow)
attribute c prompt (display white)
attribute d line (display green)
legend at 2 2 box
label display as c, at 15 2 "STUPID GRAPH"
background blue

curve aaa
'grftstl8.i1' using axis ax2
absolute
legend xy "curve aaa"

curve two
'grftstl8.i2'
additive using curve aaa
legend xy "curve two"
axis ax1
  horizontal
display as x
at 15 30
size 30
label c "axis ax1"

axis ax2
at 15 30
size 15
label c "axis ax2"
vertical
display as x
Figure C-19 GDL Test Activity S: and corresponding GDL
axis ax1
  horizontal
  display as x
  at 15 30
  size 30
  label c "axis ax1"

axis ax2
  at 15 30
  size 15
  label c "axis ax2"
  vertical
  display as x
create form grftst20
  size 80 by 30

item i1 (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf19
  at 1 15
  size 60 by 20
  display as blue

create bar graph grf19
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  legend h at 6 2 box
  label display as c, at 15 2 "STUPID GRAPH"
  background blue

curve aaa
  'grftst20.i1' using axis ax2
  absolute
  legend xy "curve aaa"

curve two
  'grftst20.i2'
  additive using curve aaa
  legend xy "curve two"
axis ax1
  horizontal
display as x
at 15 30
size 30
label c "axis ax1"

axis ax2
  at 15 30
  size 15
  label c "axis ax2"
  vertical
display as x
Figure C-21 GDL Test Activity U: and corresponding GDL
create form grftst21
  size 80 by 30

item il (3 v 0)
display as red
at 1 2
size 3
domain (numeric)

graph grf20
at 1 15
size 60 by 20
display as blue

create pie graph grf20
at 2 2
size 40 by 16
using ('grftst21.il')
attribute c prompt (display magenta)

pie 1
  quantity c outside
  shade color yellow

pie 2
  percent c inside
  quantity c outside
  shade color white

pie 3
  shade color red
  explode 20
Figure C-22 GDL Test Activity V: and corresponding GDL
create form grftst22
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf21
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf21
  at 2 2
  size 40 by 16
  using ('grftst22.il')
  attribute c prompt (display magenta)

pie 1
  quantity c outside
  shade color yellow

pie 2
  shade color red

pie 3
  percent c outside
  quantity c outside
  label c "this is a white pie slice"
  shade color white
  explode 20
Figure C-23 GDL Test Activity W: and corresponding GDL
create form grftst23
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf22
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf22
  at 2 2
  size 40 by 16
  using ('grftst23.il')
  attribute c prompt (display magenta)
  legend at 2 48 box

pie 1
  quantity c outside
  shade color yellow
  legend c "segment 1"

pie 2
  shade color red
  legend c "segment 2"

pie 3
  percent c outside
  quantity c outside
  shade color white
  legend c "segment 3"
  explode 20
Figure C-24 GDL Test Activity X: and corresponding GDL
create form grftst24
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf23
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf23
  at 5 2
  size 40 by 10
  using ('grftst24.il')
  attribute c prompt (display magenta)
  legend at 5 45 box

  pie 1
    quantity c outside
    shade color yellow
    legend c "segment 1"

  pie 2
    shade color red
    legend c "segment 2"

  pie 3
    percent c outside
    quantity c outside
    shade color white
    legend c "segment 3"
    explode 20
Figure C-25 GDL Test Activity Y: and corresponding GDL
create form grftst25
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

graph grf24
  at 1 15
  size 60 by 20
  display as blue

create pie graph grf24
  at 5 2
  size 40 by 10
  using ('grftst25.il')
  attribute a prompt (display red)
  attribute b prompt (display blue)
  attribute c prompt (display magenta)
  legend at 5 45 box

pie 1
  quantity c outside
  shade color yellow
  percent b inside
  legend c "segment 1"

pie 2
  shade color red
  label a "The Shadow knows!!!"
  legend c "segment 2"

pie 3
  percent c outside
  quantity c outside
  shade color white
  label c "Who knows what evil lurks"
  label c "in the hearts of men?"
  legend c "segment 3"
  explode 20
Figure C-26 GDL Test Activity Z: and corresponding GDL
create form grftst26
  size 80 by 30

item il (3 v 0)
  display as red
  at 1 2
  size 3
  domain (numeric)

item i2 (5 v 0)
  display as yellow
  at 1 6
  size 3
  domain (numeric)

graph grf25
  at 1 15
  size 60 by 20
  display as blue

create bar graph grf25
  using (1, 2, 3, 4, 5 axis ax1)
  attribute a fill (display cyan)
  attribute b line (display magenta)
  attribute xy prompt (display yellow)
  attribute x line (display yellow)
  attribute c prompt (display white)
  attribute d line (display green)
  background blue

curve aaa
  'grftst26.il' using axis ax2
  absolute

curve two
  'grftst26.i2'
  additive using curve aaa

axis ax1
  horizontal
  display as x
  at 15 30
  size 30
axis ax2
at 15 30
size 15
vertical
display as x
Figure C-27 GDL Test Activity AA: and corresponding GDL
create form grftst27
size 80 by 30

item i1 (3 v 0)
display as red
at 1 2
size 3
domain (numeric)

item i2 (5 v 0)
display as yellow
at 1 6
size 3
domain (numeric)

graph grf26
at 1 15
size 60 by 20
display as blue

create bar graph grf26
using (1, 2, 3, 4, 5 axis ax1)
attribute a line (display cyan)
attribute e line (display red)
attribute b line (display magenta)
attribute xy prompt (display yellow)
attribute x line (display yellow)
attribute c prompt (display white)
attribute d line (display green)
legend at 2 2
background blue

curve aaa
'grftst10.i1' using axis ax2
absolute
shade color yellow
display as a

curve two
'grftst10.i2'
additive using curve aaa
shade color green
display as e
axis ax1
  horizontal
display as x
at 15 30
tick 5 1 c "A" "B" "C"
size 30
fine grid

axis ax2
at 15 30
size 15
vertical
display as x
grid
maximum 110