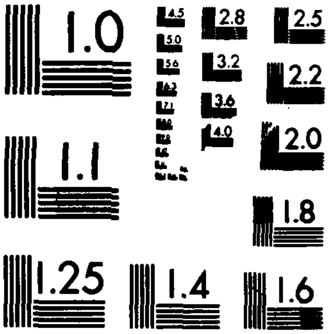
AD-A183 063 INTEGRATED INFORMATION SUPPORT SYSTEM (IISS) VOLUME 8 1/1
USER INTERFACE SUBS..(U) GENERAL ELECTRIC CO
SCHENECTADY NY PRODUCTION RESOURCES CONSU..
UNCLASSIFIED C MORENC ET AL. 01 NOV 85 DS-620144700 F/G 12/5 NL



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

2

AFWAL-TR-86-4006
Volume VIII
Part 32

INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VIII - User Interface Subsystem
Part 32 - Application Interface Development Specification

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

Final Report for Period 22 September 1980 - 31 July 1985

November 1985

Approved for public release; distribution is unlimited.

MATERIALS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AFB, OH 45433-6533

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; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

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

David Judson

DAVID L. JUDSON, PROJECT MANAGER
AFWAL/MLTC
WRIGHT PATTERSON AFB OH 45433

5 Aug 1986
DATE

FOR THE COMMANDER:

Gerald C. Shumaker

GERALD C. SHUMAKER, BRANCH CHIEF
AFWAL/MLTC
WRIGHT PATTERSON AFB OH 45433

7 Aug 86
DATE

"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 AFWAL/MLTC, W-PAFB, OH 45433 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.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

A183 063

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release;
distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)
AFWAL-TR-86-4006 Vol VIII, Part 32

6a. NAME OF PERFORMING ORGANIZATION
General Electric Company
Production Resources Consulting

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
AFWAL/MLTC

6c. ADDRESS *(City, State and ZIP Code)*
1 River Road
Schenectady, NY 12345

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

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
Materials Laboratory
Air Force Systems Command, USAF

8b. OFFICE SYMBOL *(If applicable)*
AFWAL/MLTC

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
F33615-80-C-5155

8c. ADDRESS *(City, State and ZIP Code)*
Wright-Patterson AFB, Ohio 45433

10. SOURCE OF FUNDING NOS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
78011F	7500	62	01

11. TITLE *(Include Security Classification)*
(See Reverse)

12. PERSONAL AUTHOR(S) Glandorf, Frank and Morenc, Carol and Robie, Penny

13a. TYPE OF REPORT
Final Technical Report.

13b. TIME COVERED
22 Sept 1980 - 31 July 1985

14. DATE OF REPORT *(Yr., Mo., Day)*
1985 November

15. PAGE COUNT
28

16. SUPPLEMENTARY NOTATION
ICAM Project Priority 6201
The computer software contained herein are theoretical and/or references that in no way reflect Air Force-owned or -developed computer software.

17. COSATI CODES

FIELD	GROUP	SUB GR.
1308	0905	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*

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

→ This specification establishes the development, test and qualification requirements of a computer program identified as the Application Interface (AI). The AI is used by application programs to create the messages which correspond to FP calls and are sent to the User Interface Monitor of the Form processor by way of the Network Transaction Manager. The UIM then decodes these messages and calls the appropriate FP routine. This enables applications to run under IISS in a distributed environment.

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

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
David L. Judson

22b. TELEPHONE NUMBER *(Include Area Code)*
513-255-6976

22c. OFFICE SYMBOL
AFWAL/MLTC

EDITION OF 1 JAN 73 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

11. Title

Integrated Information Support System (IISS)
Vol VIII - User Interface Subsystem
Part 32 - Application Interface Development Specification

A S D 86 0045
9 Jan 1986

PREFACE

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

Certain work aimed at improving Test Bed Technology has been performed by other contracts with Project 6201 performing integrating functions. This work consisted of enhancements to Test Bed software and establishment and operation of Test Bed hardware and communications for developers and other users. Documentation relating to the Test Bed from all of these contractors and projects have been integrated under Project 6201 for publication and treatment as an integrated set of documents. The particular contributors to each document are noted on the Report Documentation Page (DD1473). A listing and description of the entire project documentation system and how they are related is contained in document FTR620100001, Project Overview.

The subcontractors and their contributing activities were as follows:

TASK 4.2

Subcontractors	Role
Boeing Military Aircraft Company (BMAC)	Reviewer.
D. Appleton Company (DACOM)	Responsible for IDEF support, state-of-the-art literature search.
General Dynamics/ Ft. Worth	Responsible for factory view function and information models.

Subcontractors	Role
Illinois Institute of Technology	Responsible for factory view funotion research (IITRI) and information models of small and medium-size business.
North American Rockwell	Reviewer.
Northrop Corporation	Responsible for factory view function and information models.
Pritsker and Associates	Responsible for IDEF2 support.
SofTech	Responsible for IDEF0 support.

TASKS 4.3 - 4.9 (TEST BED)

Subcontractors	Role
Boeing Military Aircraft Company (BMAC)	Responsible for consultation on applications of the technology and on IBM computer technology.
Computer Technology Associates (CTA)	Assisted in the areas of communications systems, system design and integration methodology, and design of the Network Transaction Manager.
Control Data Corporation (CDC)	Responsible for the Common Data Model (CDM) implementation and part of the CDM design (shared with DACOM).
D. Appleton Company (DACOM)	Responsible for the overall CDM Subsystem design integration and test plan, as well as part of the design of the CDM (shared with CDC). DACOM also developed the Integration Methodology and did the schema mappings for the Application Subsystems.

Subcontractors	Role
Digital Equipment Corporation (DEC)	Consulting and support of the performance testing and on DEC software and computer systems operation.
McDonnell Douglas Automation Company (McAuto)	Responsible for the support and enhancements to the Network Transaction Manager Subsystem during 1984/1985 period.
On-Line Software International (OSI)	Responsible for programming the Communications Subsystem on the IBM and for consulting on the IBM.
Rath and Strong Systems Products (RSSP) (In 1985 became McCormack & Dodge)	Responsible for assistance in the implementation and use of the MRP II package (PIOS) that they supplied.
SofTech, Inc.	Responsible for the design and implementation of the Network Transaction Manager (NTM) in 1981/1984 period.
Software Performance Engineering (SPE)	Responsible for directing the work on performance evaluation and analysis.
Structural Dynamics Research Corporation (SDRC)	Responsible for the User Interface and Virtual Terminal Interface Subsystems.

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

Contractors	ICAM Project	Contributing Activities
Boeing Military Aircraft Company (BMAC)	1701, 2201, 2202	Enhancements for IBM node use. Technology Transfer to Integrated Sheet Metal Center (ISMC).

Contractors	ICAM Project	Contributing Activities
Control Data Corporation (CDC)	1502, 1701	IISS enhancements to Common Data Model Processor (CDMP).
D. Appleton Company (DACOM)	1502	IISS enhancements to Integration Methodology.
General Electric	1502	Operation of the Test Bed and communications equipment.
Hughes Aircraft Company (HAC)	1701	Test Bed enhancements.
Structural Dynamics Research Corporation (SDRC)	1502, 1701, 1703	IISS enhancements to User Interface/Virtual Terminal Interface (UI/VTI).
Systran	1502	Test Bed enhancements. Operation of Test Bed.

TABLE OF CONTENTS

APPENDICES

FIGURES

SECTION 1

SCOPE

1.1 Identification

This specification establishes the development, test and qualification requirements of a computer program identified as the Application Interface, referred to as the AI. The AI is one configuration item of the Integrated Information Support System (IISS) User Interface (UI).

1.2 Functional Summary

The AI is a collection of procedures that may be linked with an application to enable it to use the Form Processor (FP) and run in the distributed IISS environment. The AI does this by sending/receiving FP requests through the NTM (Network Transaction Manager) to/from the User Interface Monitor (UIM) of the Form Processor.

```
 Computer 1          |           Computer 2
                     |
+---------+          |    +-----+           +----+------+
|   NTM   |<---------+--->| NTM |<--------->| AI |  AP  |
+---------+          |    +-----+           +----+------+
     ^               |
     |               |
     V               |
+---------+          |
|         |          |
|   FP    |          |
|         |          |
+---------+          |
```

Figure 1-1 Distributed IISS Environment

SECTION 2

DOCUMENTS

2.1 Reference Documents

[1] General Electric Co., System Design Specification, 7 February 1983.

[2] Structural Dynamics Research Corporation, Form Processor Development Specification, DS 620144200B, 1 November 1985.

[3] Structural Dynmics Research Corporation, Forms Language Compiler Development Specification, DS 620144401B, 1 November 1985.

[4] Structural Dynamics Research Corporation, Forms Driven Form Editor Development Specification, DS 620144402B, 1 November 1985.

[5] Structural Dynmics Research Corporation, Report Writer Development Specification, DS 620144501 , 1 November 1985.

[6] Structural Dynamics Research Corporation, Rapid Application Generator Devleopment Specification, DS 620144502 , 1 November 1985.

[7] Structural Dynamics Research Corporation, Text Editor Development Specification, DS 620144600B, 1 November 1985.

[8] Structural Dynamics Research Corporation, User Interface Services Development Specification, DS 620144100B, 1 November 1985.

[9] Structural Dynamics Research Corporation, Virtual Terminal Interface Development Specification, DS 620144300B, 1 November 1985.

[10] Systran, ICAM Documentation Standards, 15 September 1983.

[11] Structural Dynamics Research Corporation, Form Processor User Manual, UM 620144200B, 1 November 1985.

2.2 Terms and Abbreviations

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

Application Definition Language: an extension of the Forms Definition Language that includes retrieval of database information and conditional actions. It is used to define interactive application programs.

Application Generator: (AG), 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.

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

Buffer Name: the default file in which the buffer will be saved if no file is given on a save command.

Common Data Model: (CDM), IISS subsystem that describes common data application process formats, form definitions, etc. of the IISS and includes conceptual schema, external schemas, internal schemas, and schema transformation operators.

Conceptual Schema: (CS), the standard definition used for all data in the CDM. It is based on IDEF1 information modelling.

Current Cursor Position: the position of the cursor before an edit command or function is issued in the text editor.

Cursor Position: the position of the cursor after any command is issued.

Cut and Paste Buffer: where deleted lines go and the paste and fill edit commands get their data.

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: is similar to the open list, except that it contains only those forms that have been added to the screen and are currently displayed on the screen.

Display Start Line: the first line in the buffer to be displayed.

Display Size: the number of lines used in the edit area.

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

External Schema: (ES), an application's view of the CDM's conceptual schema.

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

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

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

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

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

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

Logical Device: a conceptual device which to an application is indistinguishable from a physical device and is then mapped to part or all of a physical device.

Neutral Data Manipulation Language: (NDML), the command language by which the CDM is accessed for the purpose of extracting, deleting, adding, or modifying data.

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

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

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

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

Physical Device: a hardware terminal.

Presentation Schema: (PS), may be equivalent to a form. It is the view presented to the user of the application.

Previous Cursor Position: the position of the cursor when the previous edit command was issued.

Previous Edit Command: the function key pressed before the current one.

Rapid Application Generator: (RAP), part of the Application Generator that generates source code for interactive programs based on a language input.

Report Definition Language: an extension of the Forms Definition Language that includes retrieval and calculation of database information and is used to define reports.

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

Select Line: one terminus of the select range.

Select Mode: when on, certain commands will be executed over the lines in the selected range. The commands are ‹DELETELINE› and replace.

Subform: a form that is used within another form.

Text Editor: (TE), 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.

Top of file: the first line of the buffer.

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

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

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

Virtual Terminal Interface: (VTI), the callable interface to the VT.

Virtual Terminal Interface Field Map: defines the complete terminal screen by breaking it into pieces of the various forms and items that are displayed. Each area of the terminal screen must be defined as belonging to a particular field in the display list.

Window Manager: a facility which allows the following to be manipulated: size and location of windows, the device on which an application is running, the position of a form within a window. It is part of the Form Processor.

SECTION 3

REQUIREMENTS

This section includes functional and performance requirements for the AI. In addition, the AI interfaces to other IISS Computer Program Configuration Item's (CPCI's) are defined.

3.1 Computer Program Definition

Application programs use the FP Application Interface (AI) to create the messages which correspond to FP calls and are sent to the User Interface Monitor (UIM) of the Form Processor by way of the Network Transaction Manager (NTM). The UIM then decodes these messages and calls the appropriate Form Processor routine. This enables applications to run under IISS in a distributed environment.

3.1.1 System Capacities

The AI procedures are written in COBOL and are linked with the application. The application runs on a DEC VAX minicomputer running the VMS operating system.

3.1.2 Interface Requirements

The AI interfaces with application programs and the UIM. The UIM receives the message formatted by the AI and translates it into a call to the appropriate FP routine. The Form Processor routines then interface with the VT by translating an application request into the appropriate VT command when input/output is necessary.

The following describes a typical application/AI transaction:

1) Application calls the AI procedure.

2) The AI procedure creates a message using the FP procedure's id number and input parameters and sends the message.

3) The UIM receives the message and calls the corresponding FP procedure.

4) The UIM creates a message using the FP procedure's output parameters and sends the message.

5) AI procedure receives message and returns the contents to the application via the argument list.

3.1.2.1 Interface Block Diagram

The structure of the AI interfaces is shown in Figure 3-1.

```
+-------------------+
|    application    |
|                   |
+-------------------+
|                   |
|        AI         |
+---------+---------+
          ^
          | messages
          |
          v
+---------+---------+
|                   |
|        NTM        |
+---------+---------+
          ^
          | messages
          |
          v
+---------+---------+
|                   |
|      UIM/FP       |
+-------------------+
```

Figure 3-1 AI Interfaces

3.1.2.2 Detailed Interface Definition

3.1.2.2.1 Application

The interface to the application is identical to that of the Form Processor procedures and is documented in the FP User Manual.

3.1.2.2.2 NTM Messages

The AI sends and receives messages to and from the UIM via the NTM. Each FP procedure is identified by a unique number in this message. The remainder of the message consists of FP input parameters for an AI send and FP output parameters for an AI receive. These message formats are documented in Appendix A.

3.1.2.2.3 UIM

The UIM receives messages from the NTM, calls the appropriate FP procedure and sends the results back to the AI via the NTM.

3.2 Detailed Functional Requirements

The AI enables an application to run under IISS in a distributed environment. It provides an application with a set of procedures whose names, number and types of arguments and functionality are identical to that of the Form Processor. These procedures are documented in the Form Processor Development Specification.

3.3 Special Requirements

3.3.1 Programming Methods

The AI is programmed using structured design and care has been taken to insure portability of the the AI code with minimum effort. Basic programming standards for readability and ease of debugging shall be followed.

3.3.2 Expandibility

Since the AI has been designed as a set of interface routines, new functionality may be added by simply adding new interface routines.

3.4 Human Performance

The AI is a set of callable procedures which an application uses to interface with a terminal. These procedures are standard for all terminals under IISS. The programming of applications is made simpler since these procedures are standard. The input and output parameters for each procedure are clearly defined by the Form Processor User Manual.

SECTION 4

QUALITY ASSURANCE PROVISIONS

4.1 Introduction and Definitions

The AI is tested using the following tests:

Computer programming test and evaluation. This testing primarily involves testing all the AI interface routines and internal functions for correct processing and output.

System test. This testing involves testing all the AI interface routines and internal functions within the integrated system.

4.2 Computer Programming Test and Evaluation

The test developed for the AI consists of another computer program which simulates the use of the AI interfaces routines by an application program. Every AI interface routine is exercised directly by the computer test program and every internal function is indirectly exercised by the computer test program. Variations on the computer test program can be provided by user interaction with the computer test program.

The same computer test program is run to test the AI during the system test process with the other components of the IISS.

SECTION 5

PREPARATION FOR DELIVERY

The implementation site for the constructed software is the Integrated Support System (IISS) Test Bed site located at the General Electric Company in Schenectady, NY. The software associated with each AI CPCI release is delivered on a media which is compatible with the IISS Test Bed. The release is clearly identified and will include instructions on procedures to be followed for installation of the release.

APPENDIX A

FP/AI MESSAGE FORMATS

The following is a COBOL declaration of the message formats used between the AI and the FP.

FP to AI message formats, FP output parameters.

NOTE: All 01 line records correspond to the FP routines being called. For example, ADDELM-FP-RECORD is used in the routine ADDELM.

```
01  ADDELM-FP-RECORD.
    05  ADDELM-ELEMENT-NUMBER   PIC 9(4).
    05  ADDELM-RCODE            PIC X(5).
01  ADDFRM-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  ADDFRM-PAGE-NUMBER      PIC 9(4).
    05  ADDFRM-RCODE            PIC X(5).
01  CHGLDV-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  CHGLDV-RCODE            PIC X(5).
01  CLSFRM-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  CLSFRM-RCODE            PIC X(5).
01  CLSLDV-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  CLSLDV-RCODE            PIC X(5).
01  GDATA-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GDATA-BUFFER-LENGTH     PIC 9(4).
    05  GDATA-RCODE             PIC X(5).
    05  GDATA-BUFFER            PIC X(4096).
01  GDATLN-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GDATLN-BUFFER-LENGTH    PIC 9(4).
    05  GDATLN-RCODE            PIC X(5).
01  GETATT-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GETATT-ATTRIBUTE        PIC X(10).
    05  GETATT-RCODE            PIC X(5).
01  GETBAK-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GETBAK-ATTRIBUTE        PIC X(10).
    05  GETBAK-RCODE            PIC X(5).
01  GETCUR-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GETCUR-FIELD-NAME       PIC X(120).
    05  GETCUR-FIELD-TYPE       PIC X.
    05  GETCUR-ROW              PIC 9(4).
    05  GETCUR-COL              PIC 9(4).
    05  GETCUR-RCODE            PIC X(5).
01  GPAGE-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GPAGE-FORM-NAME         PIC X(10).
    05  GPAGE-RCODE             PIC X(5).
```

```
01  GWINDO-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  GWINDO-TOTAL-PAGES      PIC 9(4).
    05  GWINDO-RCODE            PIC X(5).
01  INQLDV-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  INQLDV-LOG-DEV-ID       PIC 9(5).
    05  INQLDV-RCODE            PIC X(5).
01  OISCR-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  OISCR-FUNCTION          PIC 9(4).
    05  OISCR-RCODE             PIC X(5).
01  OPNFRM-FP-RECORD REDEFINES ADDELM-FP-RECORD.
    05  OPNFRM-RCODE            PIC X(5).
    01  OPNLDV-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  OPNLDV-LOG-DEV-ID       PIC 9(5).
        05  OPNLDV-RCODE            PIC X(5).
    01  OUTSCR-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  OUTSCR-RCODE            PIC X(5).
    01  PARFQN-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  PARFQN-PAR-NAME         PIC X(120).
        05  PARFQN-PAR-TYPE         PIC X.
        05  PARFQN-RCODE            PIC X(5).
    01  PDATA-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  PDATA-RCODE             PIC X(5).
    01  PUTATT-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  PUTATT-RCODE            PIC X(5).
    01  PUTBAK-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  PUTBAK-RCODE            PIC X(5).
    01  PUTCUR-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  PUTCUR-RCODE            PIC X(5).
    01  PUTLOC-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  PUTLOC-RCODE            PIC X(5).
    01  RMVPAG-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  RMVPAG-RCODE            PIC X(5).
    01  RPLFRM-FP-RECORD REDEFINES ADDELM-FP-RECORD.
        05  RPLFRM-RCODE            PIC X(5).
```

AI to FP message format, input parameters.

```
    01  INPUT-RECORD.
        05  ROUTINE-ID              PIC 99.
    01  ADDELM-FPAI-RECORD REDEFINES INPUT-RECORD.
        05  ROUTINE-ID              PIC 99.
        05  ADDELM-ELEMENT-NAME     PIC X(120).
```

```
01  ADDFRM-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  ADDFRM-WINDOW-NAME        PIC X(120).
    05  ADDFRM-FORM-NAME          PIC X(10).
01  CHGLDV-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  CHGLDV-LOG-DEV-ID         PIC 9(5).
01  CLSFRM-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  CLSFRM-FORM-NAME          PIC X(10).
01  CLSLDV-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  CLSLDV-LOG-DEV-ID         PIC 9(5).
01  GDATA-FPAI-RECORD REDEFINES  INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  GDATA-INSTANCE-ID         PIC 9(4).
    05  GDATA-FIELD-NAME          PIC X(120).
01  GDATLN-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  GDATLN-FIELD-NAME         PIC X(120).
01  GETATT-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  GETATT-FIELD-NAME         PIC X(120).
    05  GETATT-DURATION           PIC 9(4).
01  GETBAK-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  GETBAK-FIELD-NAME         PIC X(120).
    05  GETBAK-DURATION           PIC 9(4).
01  GETCUR-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
01  GPAGE-FPAI-RECORD REDEFINES  INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  GPAGE-WINDOW-NAME         PIC X(120).
    05  GPAGE-PAGE-NUMBER         PIC 9(4).
01  GWINDO-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  GWINDO-WINDOW-NAME        PIC X(120).
01  INQLDV-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
01  OISCR-FPAI-RECORD REDEFINES  INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  OISCR-WINDOW-NAME         PIC X(120).
01  OPNFRM-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  OPNFRM-FORM-NAME          PIC X(10).
01  OPNLDV-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
```

```
01  OUTSCR-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  OUTSCR-WINDOW-NAME        PIC X(120).
01  PARFQN-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PARFQN-FIELD-NAME         PIC X(120).
    05  PARFQN-FIELD-TYPE         PIC X.
    05  PARFQN-LEVEL              PIC 9(4).
01  PDATA-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PDATA-FIELD-NAME          PIC X(120).
    05  PDATA-BUFFER              PIC X(4096).
01  PMSGLC-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PMSGLC-MSG-CODE           PIC X(5).
01  PMSGLS-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PMSGLS-MSG-STRING         PIC X(60).
01  PUTATT-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PUTATT-FIELD-NAME         PIC X(120).
    05  PUTATT-DURATION           PIC 9(4).
    05  PUTATT-ATTRIBUTE          PIC X(10).
01  PUTBAK-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PUTBAK-FIELD-NAME         PIC X(120).
    05  PUTBAK-DURATION           PIC 9(4).
    05  PUTBAK-ATTRIBUTE          PIC X(10).
01  PUTCUR-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PUTCUR-FIELD-NAME         PIC X(120).
01  PUTLOC-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  PUTLOC-FIELD-NAME         PIC X(120).
    05  PUTLOC-ROW                PIC 9(4).
    05  PUTLOC-COL                PIC 9(4).
01  RMVPAG-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  RMVPAG-WINDOW-NAME        PIC X(120).
    05  RMVPAG-PAGE-NUMBER        PIC 9(4).
01  RPLFRM-FPAI-RECORD REDEFINES INPUT-RECORD.
    05  ROUTINE-ID                PIC 99.
    05  RPLFRM-WINDOW-NAME        PIC X(120).
    05  RPLFRM-PAGE-NUMBER        PIC 9(4).
    05  RPLFRM-FORM-NAME          PIC X(10).
```

*U.S.Government Printing Office: 1987 — 748-061/60978

END
9-87
DTIC