

Laser-Scan Ltd.

IMP

Reference Manual

Volume I

Issue 5.9 - 3-December-1997

Copyright (C) 1997 Laser-Scan Ltd
Science Park, Milton Road, Cambridge, England CB4 4FY tel: (01223) 420414

Document "IMP REFERENCE", Category "REFERENCE"
Document Issue 5.9 Matt Wenham (modified 3-Dec-1997)

Laser-Scan Ltd.

IMP

Reference Manual

Volume II

Issue 5.9 - 3-December-1997

Copyright (C) 1997 Laser-Scan Ltd
Science Park, Milton Road, Cambridge, England CB4 4FY tel: (01223) 420414

Document "IMP REFERENCE", Category "REFERENCE"
Document Issue 5.9 Matt Wenham (modified 3-Dec-1997)

Laser-Scan Ltd.

IMP

Reference Manual

Volume III

Issue 5.9 - 3-December-1997

Copyright (C) 1997 Laser-Scan Ltd
Science Park, Milton Road, Cambridge, England CB4 4FY tel: (01223) 420414

Document "IMP REFERENCE", Category "REFERENCE"
Document Issue 5.9 Matt Wenham (modified 3-Dec-1997)

CONTENTS

IMP reference documentation change record	i
PREFACE	1
Intended audience	1
Structure of this document	1
Associated documents	2
Conventions used in this document	2
Command line data types	3

CHAPTER 1 INTRODUCTION

INTRODUCTION	1-1
IMP - FEATURES	1-1
IMP and the DAMP package.	1-1
IMP and IFF	1-3
Introduction of new IFF entries for IMP	1-3
IMP and the IFF HI entry	1-4
IMP and the concept of origin offset	1-5
IMP and DCL symbol \$STATUS	1-5
IMP and three dimensional strings.	1-5
Getting started with IMP	1-6
IMP modules and /IN_SITU mode	1-6
Opening IFF file with IMP	1-7
How to specify IMP command qualifier arguments .	1-9
Integer value ranges	1-10
Floating point value ranges	1-10

CHAPTER 2 MODULE ICHECK

FUNCTION	2-1
FORMAT	2-1
PROMPT	2-1
PARAMETER	2-1
COMMAND QUALIFIERS	2-2
RESTRICTIONS	2-5
DESCRIPTION	2-6
EXAMPLES	2-11
MESSAGES (INFORMATIONAL)	2-16
MESSAGES (WARNING)	2-17
MESSAGES (ERROR)	2-19
MESSAGES (OTHER)	2-22

CHAPTER 3 MODULE IDIFFERENCE

FUNCTION	3-1
FORMAT	3-1
PROMPTS	3-1
PARAMETERS	3-1
COMMAND QUALIFIERS	3-2
DESCRIPTION	3-5
General	3-5

What is compared	3-6
Comparing real values	3-6
The RA (RAnge) entries	3-7
The HI (HIStory) entries	3-7
The MH (Map Header) entries	3-7
The MD (Map Descriptor) entries	3-7
The NS (New Section) entries	3-7
The CC (Cubic Coefficients) entries	3-8
The CP (Control Point) entries	3-8
The Layers	3-8
The Features	3-8
Summary of differences	3-12
EXAMPLES	3-14
MESSAGES (GENERAL)	3-24
MESSAGES (INFORMATIONAL)	3-25
MESSAGES (ERROR)	3-29
MESSAGES (FATAL)	3-31
MESSAGES (OTHER)	3-33

CHAPTER 4

MODULE IFILTER

FUNCTION	4-1
FORMAT	4-1
PROMPT	4-2
PARAMETERS	4-2
COMMAND QUALIFIERS	4-2
RESTRICTIONS	4-6
DESCRIPTION	4-6
General	4-6
BUNCH Filter	4-6
Douglas-Peucker Filter	4-7
Curve Interpolation	4-8
IFF Files With Junction Structure	4-9
EXAMPLES	4-10
MESSAGES (INFORMATIONAL)	4-24
MESSAGES (WARNING)	4-26
MESSAGES (ERROR)	4-28
MESSAGE (FATAL)	4-31
MESSAGES (OTHER)	4-32

CHAPTER 5

MODULE IFIXAREA

FUNCTION	5-1
FORMAT	5-1
PROMPT	5-2
PARAMETERS	5-2
COMMAND QUALIFIERS	5-2
RESTRICTIONS	5-4
DESCRIPTION	5-5
EXAMPLES	5-6
MESSAGES (INFORMATIONAL)	5-10
MESSAGES (WARNING)	5-12
MESSAGES (ERROR)	5-13
MESSAGES (OTHER)	5-15

CHAPTER 6 MODULE IFROMTEXT

FUNCTION	6-1
FORMAT	6-1
PROMPTS	6-1
PARAMETERS	6-2
COMMAND QUALIFIERS	6-2
DESCRIPTION	6-3
General	6-3
IFROMTEXT and Map Headers	6-5
IFROMTEXT and Map Descriptors	6-7
IFROMTEXT and the /ORIGIN qualifier	6-8
IFROMTEXT and the CB entry	6-8
IFROMTEXT and Default Procedures	6-9
EXAMPLES	6-11
MESSAGES (INFORMATIONAL)	6-13
MESSAGES (WARNING)	6-22
MESSAGES (ERROR)	6-32
MESSAGES (FATAL)	6-33
MESSAGES (OTHER)	6-34

CHAPTER 7 MODULE IINFO

FUNCTION	7-1
FORMAT	7-1
PROMPT	7-2
PARAMETER	7-2
COMMAND QUALIFIERS	7-2
DESCRIPTION	7-5
EXAMPLES	7-9
MESSAGES (INFORMATIONAL)	7-18
MESSAGES (WARNING)	7-20
MESSAGES (ERROR)	7-21
MESSAGES (OTHER)	7-22

CHAPTER 8 MODULE ILAYFC

FUNCTION	8-1
FORMAT	8-1
PROMPTS	8-1
PARAMETERS	8-1
QUALIFIERS	8-2
DESCRIPTION	8-2
EXAMPLES	8-3
MESSAGES (INFORMATIONAL)	8-5
MESSAGES (WARNING)	8-6
MESSAGES (ERROR)	8-8
MESSAGES (OTHER)	8-9

CHAPTER 9 MODULE IMEND

FUNCTION	9-1
FORMAT	9-1

PARAMETER	9-1
COMMAND QUALIFIERS	9-1
DESCRIPTION	9-2
EXAMPLES	9-4
MESSAGES (INFORMATIONAL)	9-7
MESSAGES (ERROR)	9-9
MESSAGES (OTHER)	9-9

CHAPTER 10

MODULE IMERGE

FUNCTION	10-1
FORMAT	10-1
PROMPT	10-3
PARAMETERS	10-3
COMMAND QUALIFIERS	10-3
DESCRIPTION	10-5
General	10-5
1. IFF File mergeing	10-5
2. IFF file NS (New Section) concatenation	10-7
3. Layer selection	10-8
4. File bifurcation	10-8
IMERGE and the IFF MD (Map Descriptor) entry	10-8
IMERGE and the IFF HI (HIstory) entry	10-10
EXAMPLES	10-10
MESSAGES (INFORMATIONAL)	10-14
MESSAGES (WARNING)	10-17
MESSAGES (ERROR)	10-21
MESSAGES (FATAL)	10-23
MESSAGES (OTHER)	10-24

CHAPTER 11

MODULE IPATCH

FUNCTION	11-1
FORMAT	11-1
PROMPTS	11-1
PARAMETER	11-1
QUALIFIERS	11-2
DESCRIPTION	11-3
General	11-3
How the PREVIOUS command works	11-4
Wildcarding	11-5
Comments	11-5
Searching	11-6
IPATCH COMMANDS	11-7
IFF ENTRIES	11-41
EXAMPLE EDITING SESSION	11-92
MESSAGES (INFORMATIONAL)	11-97
MESSAGES (WARNING)	11-100
MESSAGES (FATAL)	11-118
MESSAGES (OTHER)	11-119

CHAPTER 12	MODULE IRECODE	
	FUNCTION	12-1
	FORMAT	12-1
	PROMPTS	12-2
	PARAMETERS	12-2
	COMMAND QUALIFIERS	12-2
	DESCRIPTION	12-5
	Implications of changing IFF entry values . .	12-6
	EXAMPLES	12-8
	MESSAGES (INFORMATIONAL)	12-11
	MESSAGES (WARNING)	12-12
	MESSAGES (ERROR)	12-14
	MESSAGES (OTHER)	12-15
CHAPTER 13	MODULE IRENUMBER	
	FUNCTION	13-1
	FORMAT	13-1
	PROMPTS	13-1
	PARAMETERS	13-1
	COMMAND QUALIFIERS	13-2
	DESCRIPTION	13-3
	EXAMPLES	13-4
	MESSAGES (INFORMATIONAL)	13-6
	MESSAGES (WARNING)	13-7
	MESSAGES (ERROR)	13-8
	MESSAGES (OTHER)	13-9
CHAPTER 14	MODULE IREPORT	
	FUNCTION	14-1
	FORMAT	14-1
	PROMPT	14-2
	PARAMETER	14-2
	COMMAND QUALIFIERS	14-2
	RESTRICTIONS	14-10
	DESCRIPTION	14-11
	EXAMPLES	14-12
	MESSAGES (INFORMATIONAL)	14-14
	MESSAGES (ERROR)	14-15
	MESSAGES (OTHER)	14-18
CHAPTER 15	MODULE ISELAC	
	FUNCTION	15-1
	FORMAT	15-1
	COMMAND QUALIFIERS	15-1
	DESCRIPTION	15-1
	General	15-1
	IFF AC (Ancillary Code) entries	15-1
	Skeleton files	15-2
	Selecting AC names or attribute names . . .	15-6

Abbreviations	15-8
ISELAC COMMANDS	15-9
EXAMPLES	15-25
MESSAGES (INFORMATIONAL)	15-29
MESSAGES (WARNING)	15-30
MESSAGES (ERROR)	15-34
MESSAGES (OTHER)	15-38

CHAPTER 16

MODULE ISELECT

FUNCTION	16-1
FORMAT	16-1
PROMPTS	16-2
PARAMETERS	16-2
COMMAND QUALIFIERS	16-2
RESTRICTIONS	16-4
DESCRIPTION	16-4
EXAMPLES	16-5
MESSAGES (INFORMATIONAL)	16-8
MESSAGES (ERROR)	16-9
MESSAGES (OTHER)	16-12

CHAPTER 17

MODULE ISORT

FUNCTION	17-1
FORMAT	17-1
PROMPTS	17-1
PARAMETERS	17-1
COMMAND QUALIFIERS	17-2
DESCRIPTION	17-3
EXAMPLES	17-3
MESSAGES (INFORMATIONAL)	17-7
MESSAGES (WARNING)	17-8
MESSAGES (ERROR)	17-9
MESSAGES (OTHER)	17-10

CHAPTER 18

MODULE ISTART

FUNCTION	18-1
FORMAT	18-1
PARAMETER	18-2
COMMAND QUALIFIERS	18-2
RESTRICTIONS	18-7
DESCRIPTION	18-8
General	18-8
Template mode	18-9
Ordnance Survey mode	18-9
Digitising table mode	18-10
Logical names required for ISTART	
digitising table mode.	18-11
Functions available in all ISTART modes.	18-12
ISTART and LASERAID	18-12
ISTART and digitising table characteristics	18-13

EXAMPLES	18-13
MESSAGES (INFORMATIONAL)	18-18
MESSAGES (WARNING)	18-19
MESSAGES (ERROR)	18-22
MESSAGES (OTHER)	18-26

CHAPTER 19

MODULE ITOTEXT

FUNCTION	19-1
FORMAT	19-1
PROMPTS	19-2
PARAMETERS	19-2
COMMAND QUALIFIERS	19-2
RESTRICTIONS	19-4
DESCRIPTION	19-5
ITOTEXT and the HI entry.	19-5
ITOTEXT and the MH entry.	19-5
ITOTEXT and the MD entry.	19-6
Modification of ITOTEXT output by the /CUSTOMER and /NOMH qualifiers.	19-7
ITOTEXT and the /BRIEF qualifier	19-7
ITOTEXT and the /ABSOLUTE qualifier	19-8
ITOTEXT and IFF junction structure	19-8
ITOTEXT and text file editing	19-9
ITOTEXT example output	19-9
EXAMPLES	19-12
MESSAGES (SUCCESS)	19-13
MESSAGES (INFORMATIONAL)	19-14
MESSAGES (WARNING)	19-15
MESSAGES (ERROR)	19-16
MESSAGES (OTHER)	19-18

CHAPTER 20

MODULE ITRANS

FUNCTION	20-1
FORMAT	20-1
PROMPTS	20-1
COMMAND PARAMETERS	20-2
RESTRICTIONS	20-2
COMMAND QUALIFIERS	20-3
DESCRIPTION	20-8
EXAMPLES	20-22
MESSAGES (INFORMATIONAL)	20-35
MESSAGES (WARNING)	20-36
MESSAGES (FATAL)	20-42
MESSAGES (OTHER)	20-45

CHAPTER 21

MODULE IWINDOW

FUNCTION	21-1
FORMAT	21-1
RESTRICTIONS	21-2
PARAMETERS	21-2

COMMAND QUALIFIERS	21-3
ORDNANCE SURVEY MODE	21-6
DESCRIPTION	21-6
PROGRAM OPERATION	21-8
EXAMPLES	21-10
MESSAGES (SUCCESS)	21-14
MESSAGES (INFORMATIONAL)	21-15
MESSAGES (WARNING)	21-17
MESSAGES (ERROR)	21-21
MESSAGES (OTHER)	21-28

CHAPTER 22 MODULE ICUT

FUNCTION	22-1
FORMAT	22-2
PROMPTS	22-3
PARAMETERS	22-3
COMMAND QUALIFIERS	22-4
RESTRICTIONS	22-6
DESCRIPTION	22-7
EXAMPLES	22-8
EXAMPLES - OF BADLY FORMED COMMAND LINE SPECIFICATION	22-10
MESSAGES (INFORMATIONAL)	22-12
MESSAGES (WARNING)	22-14
MESSAGES (ERROR)	22-15
MESSAGES (OTHER)	22-18

APPENDIX A ITRANS - SPHEROIDS OFFERED

APPENDIX B ITRANS - PROJECTIONS OFFERED

APPENDIX C ITRANS - NUMBER FORMATS FOR PROMPTS

IMERGE chapter to reflect alteration to the checks performed on type 2 MD (Map Descriptor) entries. (All projection parameters now compared, not just the projection zone).

Version 1.4 Jon Billing

13-November-1987

ISORT chapter to document addition of /FC qualifier. (Sort by feature code).

IMEND chapter updated to reflect new /RANGE qualifier.

Version 2.0 Various

14-January-1988

ISTART now sets the projection, spheroid and units fields of the MD (Map Descriptor) entry

All IMP utilities now deal with CB (Coordinate Block) entries.

Version 2.1 Various

5-May-1988

New ISTART /NOTICKS and /NOSYMBOLS qualifiers documented.

All chapters - convert to new chapter titling conventions

New IINFO /LAYER qualifier documented.

IREPORT has the addition of %TEST \$CURSINWIN and %TEST %ZOOM 1 to LITES2 output file (as in ICHECK).

IREPORT has /NF entries FSN, ISN now displayed as unsigned integers.

IFILTER /BEZIER qualifier restrictions documented.

Version 2.2 R.W. Russell

1-Jun-1988

New ITRANS /DATUM_SHIFT, /HEIGHT_ADJUST and /PROJECTIVE qualifiers documented.

ITRANS now does not prompt for map descriptor, if none available and doing plane transformations

ITRANS documentation tidied.

Version 2.3 A.L. Bennetton

25-Jul-1988

IFILTER chapter to include Figures and to fix some ambiguities in text.

Version 2.4 D.R. Catlow

2-Sep-1988

IFILTER chapter to change reference to maximum number of points handled from 10000 to 15000, and to add an additional information message.

Version 2.5 A.C. Morris

3-Nov-1988

New IFIXAREA. /CLOSE qualifier has had a real argument added to permit closing of all open features within a specified tolerance only.

Version 2.6 D R Catlow

23-Nov-1988

IFILTER chapter modified. Information on use of BEZIER qualifier on junction structure IFF files added.

Version 2.7 R W Russell

28-Feb-1989

ITRANS chapter modified. Information on use of DATUM_SHIFT modified.

Version 2.8 A.T. Timms, R.W. Russell

17-Mar-1989

IWINDOW chapter modified to include the 5 new qualifiers, /ABSOLUTE, /AREA, /FRT, /FSN_BY_LAYER, and /LITES2.

ITRANS chapter modified. Typographical corrections; data on projection 22 included; some illegal combination of switches clarified.

ITRANS_SPHEROIDS appendix modified. Numbers refined slightly, and presented with more figures

Version 2.9 R.W. Russell, John Cadogan

30-Oct-1989

ITRANS chapter modified. New action when scaling up control points documented.

ICHECK chapter modified. ICHECK now copes with ZS entries.

Version 3.0 John Cadogan

9-Feb-1990

IINFO chapter modified. /REVISION_LEVEL and /BRIEF qualifiers added.

Version 3.1 P Pan

12-Feb-1990

IMERGE chapter modified to include a new qualifier, /UNIQUEFSN.

Version 3.2 P Pan

20-Feb-1990

IMERGE chapter modified. Warning message UNSORTFSN modified to display LAYER and FSN information.

Version 3.3 R.W. Russell

1-Mar-1990

ITRANS chapter modified. New qualifier /RETAIN_CP; more information about /POINT and /DESCRIPTOR qualifiers.

Version 3.4 T.J.Hartnall

10-Apr-1990

ISELAC chapter modified. New NOSELFND message documented and general description reflects changed behaviour when no features are selected.

Version 3.5 P.Pan

14-May-1990

IDIFFERENCE chapter modified. New qualifier /IGNORE documented.

Version 3.6 J.Barber

07-Jun-1990

IWINDOW chapter modified. New logical name LSL\$EMPTY_LITES2_GUIDANCE documented. Some qualifier restrictions changed.

Version 3.7 C.Biggs

07-Jun-1990

IINFO chapter modified. New qualifier /ABSOLUTE documented.

Version 3.8 A.Verrill

18-Jun-1990

IREPORT chapter modified. Insist on use of brackets around /WITHFC=, /FC=, /INLAYER= and /AC= lists.

Version 3.9 Jon Barber

04-Jul-1990

IFROMTEXT chapter modified. New message UNKNACDNAM for ACD name not found in FRT supplied, or defaults.

Version 4.0 Catherine Biggs, R.W. Russell

06-Sep-1990

Added section on new qualifier /CONTROL, to allow a file containing the control points to be read.

New qualifier /SIX, to allow second order transformation. Change references to "plane transformations" to "empirical transformations". More explanation of these transformations.

Version 4.1 R.W. Russell

17-Oct-1990

/CONTROL file takes absolute coordinates.

Format of some examples changed slightly

Version 4.2 T.J. Hartnall

22-Oct-1990

New ICHECK /IGNORE qualifier documented.

Version 4.3 S Townrow

26-Oct-1990

IFROMTEXT chapter modified. Explanation of how type 3 and 4 OS map headers are dealt with.

IINFO chapter modified. New qualifier /MH_TYPE documented.

IPATCH chapter modified. Type 3 and 4 headers are output as ASCII.

ISTART chapter modified. New qualifier /MH_TYPE documented along with a description of how type 3 and 4 headers are handled.

ITOTEXT chapter modified. Conversion of type 3 and 4 headers is now possible.

IWINDOW chapter modified. Explanation of how type 3 and 4 OS map headers are dealt with. New warning message: MHMDDIFF - conflicting Map Header and Map Descriptor values.

Version 4.4 J Barber

15-Jan-1991

IINFO chapter modified. New qualifiers /NORTH, /SOUTH, /EAST and /WEST to define a window of interest within the IFF file.

New qualifier /OUTPUT to specify a file for the IINFO output if required.

New qualifier /SYMBOLS to output the program statistics to DCL symbols.

New error message MDNOTFOUND to signal the absence of any Map Descriptor entry if /ABSOLUTE qualifier given.

Version 4.5 J Cadogan

15-Jan-1991

ISTART OS type 2 map headers now set to spaces instead of nulls.

ISTART For OS type 3 map headers OSTF -1 AND -20 records are now inserted at positions specified in the translation table, LSL\$OS_MH_TABLE

ISTART chapter modified. The following paragraphs were changed:

Warning

Type 3 and 4 map headers are not intended to be used outside OS, and Laser-Scan recommend that other users of OS data use type 2 map headers which can be edited using the CONVERT utility OSMHED.

ISTART produces only a limited number of fields that can be inserted into the MH entry. It is intended that the header is manually edited with a suitable OS developed header editor to insert other fields such as entry flags and the job number before using CONVERT utilities such as OSPIF.

If the /MH_TYPE qualifier was used to create a type 3 or 4 header, the start sheet flag (-1), header flag (-20), Eastings, Northings, scale, capture scale and basic grid interval are inserted into the header using the positions and sizes specified in the translation table, LSL\$OS_MH_TABLE.

Version 4.6 S Townrow

26-Feb-1991

IWINDOW chapter modified. Type 4 OS map headers cannot be processed due to the lack of a Basic Grid Interval (BGI) in a CITF MH entry. If this is attempted, a new warning message is given: NOBGIINMH4, no BGI in type 4 Map Header (MH) entry.

Version 4.7 S Townrow

25-Mar-1991

IWINDOW chapter modified. Composite text entries which lie outside the clipping area are now removed unless /KEEP_TEXT was specified.

Version 4.8 S Gupta

7-May-1991

ISELECT modified. New qualifier /EXCLUDE_FC de-selects all feature codes which match specified numbers.

/EXCLUDE_FC and /FC qualifiers can not be used in combination.

/NOFC qualifier no longer valid.

Version 4.9 S Gupta

28-May-1991

New IMP utility ICUT added. New chapter for ICUT documentation added. Release notes and software product specification updated.

Version 5.0 S Gupta

12-Jun-1991

IRECODE modified. New qualifier /ALPHABETIC changes the feature code of the current feature using a list of associations read from the file specified as the argument to /ALPHABETIC.

Chapter on IRECODE, release notes and help library modified to accomodate change.

Version 5.1 S Townrow

12-Jun-1991

IFROMTEXT documentation clarified concerning the types of map header entries supported and who they are intended for.

Version 5.2 S Townrow

26-Nov-1992

IFROMTEXT chapter has had a revised explanation of how type 3 and 4 OS map headers are dealt with.

ISTART chapter has been modified to show the default header now created for a type 4 map header.

IWINDOW chapter has been changed to explain that the basic grid interval for a type 4 map header is now calculated from the scale.

Version 5.3 R Russell

4-Mar-1993

ITRANS chapter has had a revised list of the new projections that the program can now deal with.

Appendix B (ITRANS PROJECTIONS) has been enlarged to cover the new projections that have been added.

Version 5.4 J Barber

29-Jul-1994

ITOTEXT chapter has had new qualifiers /ABSOLUTE, /WIDTH and /DECIMAL_PLACES added to enable the output of absolute coordinates, with a user specified field width and number of decimal places if required.

IFROMTEXT chapter has had a new qualifier /ORIGIN = (x,y) to supply an origin offset to convert an absolute coordinate text file to a local coordinate IFF file.

Version 5.5 J Barber

02-Nov-1994

ITOTEXT chapter now describes the qualifiers /ABSOLUTE, /WIDTH and /DECIMAL_PLACES that are now usable independently of each other.

Version 5.6 J Barber

16-May-1995

ITOTEXT chapter now describes the qualifier /QUOTED_TEXT to output quotes around all text in the text entries.

Version 5.7 J Barber

13-Jul-1995

ITRANS chapter now describes the new Modified-Stereographic projection available via a new LSLMAINT package GCTPLIB library, and a new LSL\$CMNIFF common block projection codes and options. See the IMP Release Notes for descriptions on this new projection and its options on the sphere.

Version 5.8 T Mulcahy

27-Sep-1995

ITRANS chapter now incorporates the new limit of 800 control points in the file specified by the qualifier /CONTROLPOINTS.

Version 5.9 M Wenham

3-Dec-1997

ICHECK chapter now details the /PARALLEL qualifier.

PREFACE

Intended audience

This manual is intended for all users of the Laser-Scan IMP (IFF Map Processing) package running under the VAX/VMS operating system.

Structure of this document

This document is composed of 2 major sections.

The Introduction is an overview of IMP and is intended as a quick reference guide to the salient features of the IMP package and the associated changes made to the Laser-Scan IFF (Internal Feature Format) library.

There then follow the User Reference Guides for the individual modules which comprise IMP. Each individual module contains the same basic categories of information. These are:

MODULE	- the name of the IMP module.
REPLACES	- which older Laser-Scan programs it replaces.
FUNCTION	- a synopsis of what the modules does
FORMAT	- a summary of the module command format and command qualifiers. Default qualifier settings are indicated.
PROMPT	- how it prompts the user.
PARAMETERS	- description of expected command parameters.
COMMAND QUALIFIERS	- description of all command qualifiers. Qualifiers are ordered alphabetically and default argument values are indicated.
DESCRIPTION	- the definitive description of the module action.
EXAMPLES	- annotated examples of module useage.
MESSAGES	- all classes of message are listed and described and suggested user action given. The messages are divided into sections according to message severity within which the messages are ordered alphabetically by message mnemonic.

Where applicable, additional categories are available for some modules. Some modules, for example, have a "RESTRICTIONS" category.

Associated documents

For summary information about a specific IMP module see the IMP User Guide.

Conventions used in this document

Convention	Meaning
<CR>	The user should press the carriage return key on the terminal
<CTRL/x>	The phrase <CTRL/x> indicates that the user must press the key labelled CTRL while simultaneously pressing another key, for example, <CTRL/Z>.
\$ IINFO JIM<CR>	Command examples show all user entered commands in bold type.
\$ IFIXAREA JIM<CR> . . .	Vertical series of periods, or ellipsis, mean either that not all the data that IMP would display in response to the particular command is shown or that not all the data that the user would enter is shown.
file-spec...	Horizontal elipsis indicates that additional parameters, values or information can be entered.
[logical-name]	Square brackets indicate that the enclosed item is optional. (Square brackets are not, however, optional in the syntax of a directory name in a file-specification, or in the syntax of a substring specification in a VMS assignment statement).
'integer'	An integer number is expected in the specified input or output field. (See "Command line data types" below).
'real'	A real number is expected in the specified input or output field. (See "Command line data types" below).

Convention	Meaning
FSN 'integer' ('integer')	FSN followed by two integer arguments indicates an IFF feature serial number. The integer number enclosed in round brackets is the feature internal sequence number.
00003DE7	A hexadecimal address of a location within an IFF file. IMP modules express all IFF addresses using hexadecimal radix. The address is always padded with leading zeros to a standard field width of 8 characters.

Command line data types

IMP utilities use the VMS Command Line Interpreter (CLI) to get and parse the program command line. IMP utilities thus offer a VMS emulating user interface. Unfortunately the VMS Digital Command Language (DCL) does not support the real (or "floating point") data type. Many IMP utilities require real value arguments for the specification of tolerances and distances etc. To meet this requirement, Laser-Scan have developed an enhanced CLI based command line decoding mechanism. This enables the interpretation of numbers as either "real" or "integer". Throughout this document the number types are differentiated by the words 'integer' for integer numbers and 'real' for real (or "floating point") numbers.

IMP command line decoding operates in decimal radix.

CHAPTER 1

INTRODUCTION

INTRODUCTION

IMP is the Laser-Scan **IFF Map Processing** package. IFF stands for **I**nternal **F**eature **F**ormat and is the Laser-Scan vector file format generated by LASERAID and other Laser-Scan mapping systems and used as the data structure throughout the Laser-Scan LAMPS system. IFF files are binary and cannot be manipulated directly using a text editor. The IMP package enables the user to perform a wide range of file manipulation tasks related to the requirements of the automated mapping industry.

IMP - FEATURES

The IMP package consists of independent modules which together form a powerful data manipulation system within an automated mapping environment. The modules which form the IMP package offer:

- o common command syntax. Module command lines are decoded using the Command Line Interpreter as used by the VAX/VMS utilities.
- o VMS format messages referenced using 32 bit condition code symbols.
- o VMS DCL symbol \$STATUS on image exit.
- o handling of IFF HI (HISTORY), TS (Text Status), ZS (three dimensional strings) and type 2 MD (Map Descriptor) entries. (For a detailed description of these and the other IFF entries see the IFF Users Guide). These entries could not be handled by the DAMP (**D**ATA **M**anipulation **P**ackage) utilities which IMP has replaced.
- o reference to an IFF origin offset information.
- o comprehensive documentation in this reference manual using a style consistent with that used by Digital Equipment Corporation in their VMS utility manuals. The IMP User Reference documentation includes an explanation of the messages output by the modules together with suggested user action.
- o on-line help which is available via the VAX/VMS HELP utility.

IMP and the DAMP package.

IMP is designed to replace the existing Laser-Scan DAMP (**D**ATA **M**anipulation **P**ackage). The following summary listing of the DAMP package can be made to indicate the functional correspondence between the two packages.

REPLACED DAMP MODULE	DAMP MODULE FUNCTION	NEW IMP MODULE
CHKLP	Check for loops in IFF data	ICHECK
CMPIFF	Compare two IFF files	IDIFFERENCE
I2TEXT	IFF to text file conversion	ITOTEXT
TEXT2I	Text file to IFF conversion	IFROMTEXT
ICE	IFF combine and extract (for edge matching)	IMERGE
ICLIP	Map clipping	IWINDOW
IED	IFF patch editor	IPATCH
IFINFO	Summarise the contents of an IFF file	IINFO
IFIX	Correct specific faults in IFF data	IFIXAREA
IFSN	Reallocate feature serial numbers	IRENUMBER
IFT	Tidy up an incomplete IFF file	IMEND
IPR	Process (transform or clip) IFF data	ITRANS
IRN	Reallocate internal sequence numbers	IRENUMBER
LIT	Change or extract feature codes by layer	ILAYFC
MER	Merge IFF files or extract IFF layers	IMERGE
MIF	Mend an improperly closed IFF file	IMEND
SCI	Scan an IFF file, giving statistics	IINFO
SELAC	Select features on the basis of ACs or FSNS	ISELAC
SIF	Sort by feature or internal sequence number	ISORT

All major DAMP package functionality is retained in IMP, although the module which provides that functionality may have changed. In addition there are IMP modules which offer functions not available within the DAMP package:

- o IFILTER - offers interpolation and/or filtering of IFF strings.
- o IRECODE - enables global change of the value of specified IFF entry fields.
- o IREPORT - report on the presence or absence of IFF entries.
- o ISELECT - select features on the basis of IFF entry components.
- o ISTART - is a universal IFF template file generator.

Other IMP modules exceed the functionality of the DAMP module equivalent. IMERGE, for example, offers division of data into 2 separate IFF output files on the basis of IFF layer, the DAMP equivalent, MER, does not. IPATCH the IMP non-graphic IFF editor offers backward movement through the IFF file. IED the DAMP non-graphic editor does not.

IMP and IFF

Within the VAX/VMS system IFF files can be treated as any other file type for file management purposes. To enable the user to distinguish an IFF file from a file of another type IFF files have by default the file extension '.IFF'. To provide great flexibility in the production environment IFF files are referenced by all the IMP modules using logical name LSL\$IF:. (For an explanation of logical names see the VAX/VMS document set). Logical name LSL\$IF: is assigned to a device and directory specification either using the VMS DEFINE command or the Laser-Scan SI utility, (see Appendix D).

Introduction of new IFF entries for IMP

Four new IFF entry types are available for use with IMP modules:

- (i) HI (HISTORY) entry. This is a mechanism for automatically recording statistics in an IFF file each time it is updated, so that it may be determined which users and programs contributed to the final state of the file.
- (ii) MD (type 2 Map Descriptor) entry, upwards compatible with the existing type 1 MD. This entry contains projection, scale and spheroid information etc. It also contains an origin offset. This is an (X,Y) coordinate pair, each held as a REAL*8 (double precision) number, that is added to any pair of coordinates in the IFF file to give the true projection coordinates of the points. This allows the absolute size of the coordinates in the IFF file, which are only held as REAL*4 (single precision) numbers, to be reduced. This avoids problems of truncation and reduced accuracy.
- (iii) TS (Text Status) entry. Traditionally, IFF text features could contain only one text string, with associated location and descriptive data. The new TS entry allows IFF text features to be **composite** - that is composed of several sub-texts or text components, which may be manipulated independently or as a single entity. Each text component starts with a TS entry, and ends with the next TS entry, or the final EF of the feature. The first TS entry occurs immediately after the FS entry and any AC entries. Text components may not include FS or AC entries, but may contain any other entries that are legal within a normal text feature.
- (iv) ZS (3 dimensional string) entry. The ZS 3 dimensional strings are held as x,y,z coordinates in a similar fashion to the existing IFF 2 dimensional ST (STring) entries.

Many Laser-Scan customers have large archives of IFF files which do not contain the new IFF entries. IMP modules are designed to treat IFF files of an historic nature in the same manner as the DAMP utilities would. No new type IFF entries are placed into an IFF file if the IFF file specified as input to an IMP utility does not already have those entries. Compatibility between IMP and existing databases is thus achieved. "New type" IFF files which contain HI, type 2 MD,

TS, and ZS entries may be generated using ISTART (or IFROMTEXT in conjunction with a suitable textual representation of an IFF file).

A problem arises for those IMP modules which take input from more than one IFF file when the input files are a mixture of "old type" and "new type" IFF files. Under these circumstances the input file which is specified first is considered to set the standard for the output file.

IMP and the IFF HI entry

The modules which form the IMP package provide a mechanism for automatically recording statistics in an IFF file each time it is updated, so that it may be determined which users and programs contributed to the final state of the file. The statistics are stored in an HI (HISTORY) entry within the IFF file. This entry is of fixed length (4001 words). The first word contains a count of the number of filled 'history records', and is followed by space for 100 80-byte ASCII records each with the following format:

Date	Time	Username	Program	Function	Elapsed	CPU	STATUS
23-JUL-1985	12:22	CLARKE	TWOTVES	Output	01:31:34	00:09:05	00000001

In order that the mechanism can work, a blank history entry must be inserted in files created from scratch. Clearly it would cause compatibility problems with customer databases which contain historic IFF files if HI entries were created in output files by default. To overcome this problem an HI entry is only written to the output file if (one of) the input files contained an HI entry. The IMP module ISTART creates a template IFF file containing an HI entry.

In addition to writing timing statistics into the HI entry, the IMP modules also set the 'function' field of the HI and also set the final status field to indicate the success (or failure!) of the run.

IMP provides a module (IINFO) which may be used to analyse the contents of an IFF file's HI entry and optionally produce timing statistics to indicate total elapsed and CPU time spent during the processing of the file.

When an IFF file is opened for write, a 'prototype' history record, with blank elapsed and CPU fields, and a status of 0, is written to the HI entry and also to the forepart of the file. In the event that the file is never properly closed, this record can be examined (possibly using the VMS DUMP utility) to determine which operation had failed. It will not be possible to open such a file with any IFF library (IFFLIB) based utility (such as those of IMP) until the IMP module IMEND has been used.

IMP and the concept of origin offset

The origin offset is held in a type 2 MD (Map Descriptor) entry. The origin offset is an (X,Y) coordinate pair, each held as a REAL*8 (double precision) number, that is added to any pair of coordinates in the IFF file to give the true projection coordinates of the points. This allows the absolute size of the coordinates in the IFF file, which are only held as REAL*4 (single precision) numbers, to be reduced, thus avoiding problems of truncation and impaired accuracy. The origin offset entry may be set up using IFROMTEXT, ISTART or ITRANS/DESCRIPTOR. Generally the use of an origin offset results in the south west (bottom left) control point of the map being reduced to the coordinate (0.0,0.0). IMP modules are designed to utilise the origin offset only when reference to adjoining map sheets (as IFF files) or IFF file projection transformation is required. The user may manipulate the within-file data relative to the local (0.0,0.0) origin without reference to the origin offset. It is most important that the origin offset is correctly set. The IMERGE (file merging) and ITRANS (file transformation) utilities rely heavily upon the origin offset. Errors in its application may result in irrevocably scrambled data.

IMP and DCL symbol \$STATUS

Like VMS utilities, all IMP modules generate VMS format messages and set VMS DCL symbol \$STATUS on image exit. This is a valuable feature as a non-interactive process can test the success of a preceding IMP module before proceeding. \$STATUS will always be set to a VMS 32 bit condition code. Successful program execution will result in \$STATUS being set to SS\$_NORMAL. If an error occurred during IMP processing, SS\$_ABORT of varying severities, or a VMS System or CLI (Command Line Interpreter) condition code will be used. The user may simply test \$STATUS for TRUE or FALSE within a DCL command procedure. If \$STATUS is TRUE then processing was successful. If it is FALSE, an error occurred during processing. For a detailed description of the uses of \$STATUS see the VAX VMS document set.

IMP and three dimensional strings.

All IMP modules handle 3 dimensional strings held in IFF ZS entries. Where appropriate the three dimensional strings are processed in the same manner as 2 dimensional strings. The exceptions to this rule are IFILTER and IWINDOW.

IFILTER does not interpolate or filter 3 dimensional strings at all but instead passes them through to the output file in an unmodified state.

IWINDOW will clip a three dimensional string using the same rules as applied to 2 dimensional strings held in ZS entries. The Z-value of the intersection between string and clipping boundary is estimated using simple linear interpolation based upon the known heights of the two points defining the segment which contains the intersection point.

Getting started with IMP

Once logged in the user must give two commands to initialise the IMP package. Before the IMP package can be used DCL symbols and logical names must be assigned to enable the user to invoke the modules. This is done using a command procedure IMPINI.COM which is supplied as part of the IMP package. IMPINI itself will be defined as a DCL symbol at your site and should be invoked thus: (see PREFACE for explanation of presentation conventions)

```
$ IMPINI<CR>
```

The IMPINI command invokes a command procedure which defines a DCL symbol (the module name) for each of the IMP modules. After using IMPINI the user need only type the symbol name to activate the module of his choice.

As an alternative to explicitly typing the IMPINI command each time the user wishes to use the IMP package, the IMPINI command may be placed in the users login file, or in the site dependent default login file.

The second command which must be given before using the IMP package is the SI command. The SI command assigns the logical name LSL\$IF: (or IF: for short) to the device-directory specification which contains the IFF file(s) that are to be manipulated. For example:

```
$ SI DUA3:[BUREAU.TRIALS.DIGITISING]
```

This will assign logical name LSL\$IF: to the device and directory specification DUA3:[BUREAU.TRIALS.DIGITISING]

For a discussion of the use of the SI command see Appendix D.

IMP modules and /IN_SITU mode

Some IMP modules offer an option to make the specified changes in an in-situ mode on the input file. If the user is confident that the changes specified are correct then there may be advantages in processing speed and certainly disk space advantages in adopting this approach. It cannot be overstressed, however, that the in-situ mode of operation does involve over-write of the source data which cannot be restored in the event of user, system or processing error. The default mode of operation of all IMP modules is to open the input file as read-only. Processing will result in the creation of a new output file. In-situ IFF file processing should be used with great care.

NOTE

The VMS CLI (Command Line Interpreter) allows the user to redefine the syntax of a command line on the basis of the presence of chosen qualifiers. This mechanism is used by some IMP utilities. A side effect of this

functionality is that command line elements **to the left of the /IN_SITU qualifier will be ignored**. If the user wants to process an IFF file using a utility with in-situ mode of operation, the /IN_SITU qualifier should appear immediately after the command, for example:

```
$ IRENUMBER/IN_SITU DUA0:[BUREAU.TRIALS]FILE4.IFF<CR>
```

IMP modules which have the /IN_SITU command qualifier and are thus subject to this restriction are:

- o ILAYFC
- o IRECODE
- o IRENUMBER

Opening IFF file with IMP

All IMP modules open IFF files using the Laser-Scan LSLLIB library of service routines. This results in a standard set of four messages for successful IFF file opening and three primary messages for IFF opening errors:

Messages to indicate that the IFF file was successfully opened:

IFFOPENED, 'file-spec' opened for read

Explanation: The indicated file was successfully opened, readonly, for input.

User action: None, other than to check that the correct file has been opened.

IFFOPENED, 'file-spec' opened for write

Explanation: The indicated file was successfully created and opened for output.

User action: None, other than to check that the correct file has been opened.

IFFOPENED, 'file-spec' opened for update

Explanation: The indicated file was successfully opened for in-situ modification or extension. Beware that the original status of the file will now be irrevocably changed.

User action: None, other than to check that the correct file has been opened.

Messages to indicate that the IFF file could not be opened:

All the messages in this category will be supported by supplementary Laser-Scan and VMS System or RMS (Record Management Services) messages. These supplementary messages will normally convey all the information required to determine **why** the file could not be opened. It is recommended that the user becomes familiar with the location of the VMS error message explanations within the VMS documentation set.

IFFOPEN, IFF error opening file 'file-spec' for read

Explanation: An error occurred while attempting to open the specified file for readonly input.

User action: The supplementary Laser-Scan, VMS system or RMS messages which are output in support of this message will facilitate diagnosis. One common mistake is to forget to assign the logical name LSL\$IF: to the directory containing the desired file. If the RMS message "file not found" is issued check the assignment of LSL\$IF:.

Possible causes for the error are:

- o the file-spec was mis-typed,
- o logical name LSL\$IF: was incorrectly assigned,
- o the file does not exist.

IFFCREATE, IFF error creating file 'file-spec'

Explanation: An error occurred while attempting to create and open the specified file for output.

User action: The supplementary Laser-Scan, VMS system or RMS messages which are output in support of this message will facilitate diagnosis.

Possible causes for the error are:

- o the file-spec was invalid
- o logical name LSL\$IF: was incorrectly assigned to a non-existent device or directory
- o the device, directory or file is write protected
- o the device is full

IFFMODIFY, IFF error opening file 'file-spec' for update

Explanation: An error occurred while attempting to open the specified file for modification.

User action: The supplementary Laser-Scan, VMS system or RMS messages which are output in support of this message will facilitate diagnosis.

Possible causes for the error are:

- o the file-spec was mis-typed
- o logical name LSL\$IF: was incorrectly assigned
- o the file does not exist
- o the device, directory or file is write protected
- o the device is full

How to specify IMP command qualifier arguments

IMP utilities use the VMS Command Line Interpreter (CLI) to get and parse the program command line. IMP utilities thus offer a VMS emulating user interface. As many IMP utilities require floating point arguments to command, qualifiers Laser-Scan have developed an enhanced CLI based command line decoding mechanism. This enables the interpretation of numbers as either "real" or "integer".

The CLI allows the user to specify single and lists of integer qualifier arguments. If a list of arguments is specified, each argument must be separated by a comma and the whole list enclosed within parentheses, for example:

Single argument:

\$ **EXAMPLE/QUALIFIER=7<CR>**

Where "EXAMPLE" is the command and /QUALIFIER is a qualifier to that command. There is one qualifier argument - 7

Argument list:

\$ **EXAMPLE/QUALIFIER=(2,5,8,9,10,11,12,13,14)<CR>**

Where "EXAMPLE" is the command and /QUALIFIER is a qualifier to that command. There are 9 qualifier arguments within the argument list.

Integer value ranges

While developing the floating point command line data type (see Preface) it was recognised that there is a need for numeric range decoding within a VMS emulating command line. Argument ranges are specified with the syntax:

n:m

Where n is the lower limit of the range and m is the upper limit of the range (inclusive).

Such ranges are expanded in full. A maximum of 1024 arguments can be specified to any one command qualifier.

If we take our example argument list used above, i.e:

\$ EXAMPLE/QUALIFIER=(2,5,8,9,10,11,12,13,14)<CR>

and now use the Laser-Scan argument range decoding mechanism:

\$ EXAMPLE/QUALIFIER=(2,5,8:14)<CR>

we see that a more compact command line results but yields the same arguments. This is clearly an advantage in an IFF map processing environment where a single file could contain hundreds of attributes which the user may wish to reference via command line arguments.

Other examples are:

\$ EXAMPLE/QUALIFIER=2:9<CR>

This yields 8 integer arguments: 2, 3, 4, 5, 6, 7, 8, and 9

\$ EXAMPLE/QUALIFIER=:8<CR>

This yields 9 integer arguments: 0, 1, 2, 3, 4, 5, 6, 7, and 8

If when ranges are decoded, a qualifier has more than 1024 arguments the Laser-Scan LSLIB library issues the error message:

%LSLLIB-E-RESPARSOVF, result of parse overflowed buffer

and program execution is terminated.

Floating point value ranges

Floating point value ranges are decoded in a different manner to integer value ranges. Instead of expanding the range to yield all its component integer values the command decoder merely leaves the range as a lower limit and an upper limit. Processing then takes account of any possible value lying between these limits (inclusive).

For example:

```
$ ISELECT/HEIGHT=(23.5:110.2)<CR>
```

Select all features having a height which lies within the
range 23.5 to 110.2 inclusive.

CHAPTER 2

MODULE ICHECK

MODULE ICHECK

REPLACES DAMP module CHKLP

FUNCTION

ICHECK is an IFF geometry checking utility. It scans the coordinates of an IFF file and reports on potential errors **within features**.

Options are provided to send output to terminal, text file, IFF plot file or to a LITES2 command file.

FORMAT

\$ ICHECK file-spec

Command qualifiers

/BASE_FC=integer
/BORDER
/[NO]CROSS
/[NO]DUPLICATES=EXACT or NINT
/FC=integer
/FRT=file-spec

/IGNORE=INVISIBLE or POLYGONS
/LAYER=integer
/[NO]LITES2[=file-spec]
/[NO]LOG
/[NO]MARKERS=file-spec
/[NO]NEATLINE=('keyword-options'...)
/[NO]OUTPUT[=file-spec]
/[NO]PARALLEL
/[NO]PRINTER
/[NO]TOUCH[=real]

Defaults

See text.
No border.
/CROSS
No duplicate point checks.
See text
No checks against
graphical types.
Process invisible moves.
See text
No LITES2 file output.
/NOLOG
No marker file output.
see text
/OUTPUT=SYS\$OUTPUT:
/NOPARALLEL
/NOPRINTER
No touch check.

PROMPT

_IFF-file: IFF-file-spec

PARAMETER

IFF-file-spec

- specifies the IFF file which is to be checked. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

COMMAND QUALIFIERS

/BASE_FC=integer

- for use with the /MARKERS qualifier. /BASE_FC=integer causes ICHECK to offset all marker box IFF feature codes by the specified value. (See description for details of marker box feature codes). This enables the user to plot (or read into LITES2) both the /MARKERS IFF file and the source IFF file simultaneously without conflict of feature codes 1,2,3,4 and 5. The default feature code base value is 0 (zero).

/BORDER[=real]

- used in conjunction with /NEATLINE, /BORDER specifies the width of a zone inside the sheet neatline, within which open feature termination is considered an error. The /BORDER argument should be specified in IFF units. The default /BORDER value is 2% of the maximum sheet edge length.

When used with the /NEATLINE check, /BORDER enables the user to determine that all digitising reaches the sheet neatlines and that none falls short (i.e. that no feature ends lie within the defined zone inset from the neatlines). The ends of closed features (i.e. features having coincident first and last points) are not reported even if they lie within the checking zone defined by /BORDER.

/CROSS

/NOCROSS

- check for all possible crossing segment combinations within the current feature. This is the default ICHECK mode of operation. Checks for crossing segments may be deselected by specifying the /NOCROSS qualifier. Deselection of crossing segment checking is only possible if at least one other check option has been selected.

/DUPLICATES=EXACT

/DUPLICATES=NINT

/NODUPLICATES

- check for consecutive duplicate points within a string. The /DUPLICATE qualifier offers two keyword arguments EXACT and NINT. If keyword EXACT is specified then both the X and both the Y coordinates of both points must match exactly for ICHECK to generate an error message. If the keyword NINT is specified then the both the X and both the Y coordinates are rounded to the nearest integer before comparison. The NINT option is the default mode of operation for the /DUPLICATES qualifier.

/FC=integer[,...]

- check only features which have the specified feature code(s). The /FC qualifier will accept single, multiple or ranges of feature code arguments. Ranges of feature codes may be specified. The maximum number of feature codes which may be specified is 1024.

/FRT=file-spec

- specifies that the graphical type of features is to be checked for correspondence with the information in the specified FRT (Feature Representation Table). This provides a check on feature coding accuracy. For example a linear feature which has a feature code associated with a symbol graphical type will be considered to be erroneous. (See description for full details of the errors reported by /FRT qualifier).

/IGNORE=INVISIBLE

/IGNORE=POLYGONS

- By default ICHECK will report touching or crossing line segments even if they represent an invisible (i.e. pen-up) move. If /IGNORE=INVISIBLE is specified, ICHECK will not report touching or crossing line segments if one, or both, are invisible moves. A single feature can contain a maximum of 512 invisible moves, if this number is exceeded then the /IGNORE=INVISIBLE functionality is not applied for that feature.

If /IGNORE=POLYGONS is specified, then ICHECK performs /IGNORE=INVISIBLE functionality and in addition:

- o does not report the closure point of nested polygons which are connected to the outer polygon ring by a retraced invisible "stalk".
- o does not report the point where the invisible line "stalk" leaves and then returns to the outer polygon ring. It is assumed that there is only one "stalk" connecting nested polygons to the outer ring, as this is the structure created by the POLYGONS package.

If more than one stalk is used, e.g. where the outer ring is defined by an archipelago of separate visible polygons joined together by invisible moves, or where polygon nesting has been created manually, the points where the stalks connect to the outer ring will be reported.

/LAYER=integer[,...]

- check only features which lie within the specified layer(s). /LAYER will accept single, multiple or ranges of layer arguments. The maximum number of layers which may be specified is 1024.

/LITES2[=file-spec]

/NOLITES2

- creates a LITES2 command file to take the user to potential errors. Messages generated to report a potential error are incorporated in the file. By default the LITES2 command file specification is parsed against that of the IFF file being checked but with the substitution of logical name LSL\$LITES2CMD: and the extension '.LCM'. Thus if the input IFF file is called LSL\$IF:TST.IFF then the default LITES2 command file is LSL\$LITES2CMD:TST.LCM. Entries in the LITES2 command file are typically of the form:

```
%MESSAGE End in border zone in Feature FSN 1812 (1834)
%POSITION      765.3      456.89
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
%ABANDON
```

/LOG

/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened, and a reassuring message output each time a new IFF layer is encountered in the IFF file.

/MARKERS=IFF-file-spec

/NOMARKERS

- causes ICHECK to create the specified IFF file containing IFF features defining boxes which represent the positions of potential errors. This IFF file may then be plotted and overlaid with the original digitising source material. (See description for full details of the IFF feature codes applied to the marker boxes output by the /MARKER qualifier).

/NEATLINE=(NORTH=real,SOUTH=real,EAST=real,WEST=real)

/NONEATLINE

- causes ICHECK to test for open features which start, or end, within a border zone inset from the neatline, and to test for all features that start, or end outside the specified neatline. For a complete description of the operation of the /NEATLINE check see the Description section below.

Not all of the keywords are required, and abbreviations are allowed, but at least one keyword must be given with a real value (in IFF units) to define the relevant neatline. Thus if the user specified /NEATLINE=(SOUTH:0.0,EAST:200.0) checks will only be performed on the WEST and NORTH sheet neatlines.

/BORDER is designed for use with the /NEATLINE qualifier. If the width of the border zone is not defined by the /BORDER qualifier then a default of 2% of the maximum (X,Y) file range is taken.

/OUTPUT[=file-spec]

- by default output will be to SYS\$OUTPUT. This option allows the user to redirect output to the specified text file. /OUTPUT is the default mode of operation for ICHECK.

/PARALLEL

/NOPARALLEL

- by default, the /TOUCH qualifier does not check the proximity of parallel segments within a feature. Using the /PARALLEL qualifier with /TOUCH means that parallel segments are checked.

/PRINTER

/NOPRINTER

- queues the ICHECK text output for printing on SYS\$PRINT: under the name given by the /OUTPUT qualifier. If you specify /PRINTER without the /OUTPUT qualifier, the output is directed to a file named SYS\$DISK:ICHECK.LIS which is queued automatically for printing and then deleted.

/TOUCH[=real]

/NOTOUCH

- causes ICHECK to look for touching segments within a feature. These are segments which are closer than the specified distance apart. The default /TOUCH distance is 0.5 IFF units.

RESTRICTIONS

- o At least one check must be specified
- o At least one output must be specified
- o /BORDER requires /NEATLINE
- o /NEATLINE requires at least one keyword

DESCRIPTION

ICHECK is an IFF geometry checking utility. It scans the coordinates of an IFF file and reports on potential errors within features. Throughout the following description a segment is defined as a line joining two consecutive coordinates within an IFF feature.

The following checks are available:

- o crossing segments, for example where the digitiser has looped back upon itself,
- o touching segments, for example where the digitiser has looped back upon itself without crossing,
- o duplicate points within a feature,
- o open features which do not intersect with a user defined sheet neatline and which start, or end, within a border zone inset from the neatline, and all features which start, or end, outside of the sheet neatlines,
- o feature characteristics at variance with the graphical type of that feature defined in an FRT (Feature Representation Table) file.

ICHECK may be used to check a subset of the features in an IFF file by use of the /LAYER and /FC qualifiers. Only features which fall within specified layers or which have the specified feature code will be checked. The default ICHECK action is to check **ALL** features in the IFF file except those which lie in IFF layer 0. Layer 0 is reserved for registration features and grids.

The default mode of ICHECK operation is to check for crossing segments within features. These errors are usually the result of poor digitising.

Error messages are sent to SYS\$OUTPUT by default. The /OUTPUT qualifier may be used to specify a file in which the error messages will be duplicated.

All the checks may be used singly or in combination. However, execution time may increase if all checks are applied and specification of a large touch tolerance may dramatically slow execution speed. In addition, it can be easier for an operator to work from several logs or LITES2 command files containing errors of the same type, rather than a single large file containing many different types of error.

The checks for crossing and touching segments within a feature do not report potential errors that lie within the touch tolerance of each other in the case of touching segments and 0.1% of the maximum range of the file in the case of crossing segments. Exactly duplicate points are ignored by both checks. Care should be taken when checking

for touching segments. If too great a touch tolerance is specified then all segments within the tolerance of each other will be reported as touching. When two almost parallel segments lie within the touch tolerance of each other then the reported point of intersection may be far off the sheet boundary ! The proximity of exactly parallel segments may be enabled by using the /PARALLEL qualifier with /TOUCH.

The check for open features which terminate within a user defined border zone should be used with care. Any open feature (i.e. a feature which has end points described by different coordinates) will be considered erroneous if one of its ends falls within the border zone. ICHECK cannot make any distinction between the symbolic functions of features, so it will not distinguish between contours and communications features. While it is incorrect for a contour to terminate in space, it is perfectly alright for a road to do so (in a cul-de-sac for example). The user should make full use of the /LAYER and /FC qualifiers to avoid confusion using this check.

If dealing with contour data it is important not to make the border zone for checking too large. The default border zone is 2% of the maximum sheet length. If made too wide on a sheet containing contours which have been feathered for clarity or labelling, a large number of erroneous warnings will be issued.

The check for consecutive duplicate points within a string offers two options:

- (i) exact match of coordinates, and,
- (ii) coordinate match after rounding to the nearest integer.

These options are selected by keyword arguments to the /DUPLICATES qualifier. The keywords are 'EXACT' for exact comparison and 'NINT' for nearest integer comparison. The NINT option is the default mode of operation for the /DUPLICATES qualifier.

The check for conflicting feature content against FRT (Feature Representation Table) information is very powerful. The feature code carried by each feature is looked up in the FRT file and the graphical type of that feature code is noted. If the graphical type is found to be at variance with the characteristics of the feature an error is reported. Errors that will be reported with the /FRT option are:

- o feature codes not found in FRT,
- o features having graphical type 1 (linear) but only a single point in the feature ST (SString entry),
- o features having graphical type 2 (clockwise arc) but having more or less than 3 points,
- o features having graphical type 3 (anticlockwise arc) but having more or less than 3 points,

- o features having graphical type 4 (circumcircle arc) but having more or less than 3 points,
- o features having graphical type 5 (full circumcircle) but having more or less than 3 points,
- o features having graphical type 6 (interpolated curve) but less than 2 points,
- o features having graphical type 7 (unoriented symbol) but more than 1 point,
- o features having graphical type 8 (oriented symbol) but having more than 2 points, or only 1 point and no accompanying RO (ROtation) entry,
- o features having graphical type 9 (scaled symbol) but having more than 2 points,
- o features having graphical type 10 (text) but no TX entry, or more than 2 points,
- o features having graphical type 11 (symbol string) but only 1 point,
- o features having graphical type 12 (fill area) but less than 3 points, or features which do not have coincident first and last points.
- o Symbol or text features with incorrectly set symbol and text bits in the FS (Feature Status) entry.

The FRT file specification supplied as the obligatory argument to the /FRT qualifier will be parsed against the default file specification LSL\$FRT:.FRT;0.

When the /FRT qualifier is specified errors are only output to the screen or to listing file as FRT errors will cause both LITES2 and FPP plotting software to produce errors.

ICHECK offers the option to output marker box features to an IFF plot file specified with the /MARKERS qualifier. This IFF file is given the RA (Range) MH (Map Header), MD (Map Descriptor), CC (Cubic Coefficients) and CP (Control Point) entries from the file which is being checked. The marker box features are placed in IFF layer 1 and are given one of the following feature codes, (possibly offset with the value specified by the /BASE_FC qualifier):

ERROR	BOX CENTRE POSITION	FEATURE CODE
Crossing segments	Over intersection	base FC +1
Touching segments	Over 1st touching point	base FC +2
End in border zone	Over offending end	base FC +3
Points outside neatline	Over an offending point	base FC +4
Duplicate point in string	Over offending point	base FC +5

If no relevant errors are detected then the empty marker file is deleted.

The marker boxes are defined by 5 point strings and therefore the FRT (Feature Representation Table) used to plot the MARKER IFF plot file should have feature codes 1 to 5 set to be graphical type 1 (linear).

The IFF plot file specification supplied as the obligatory argument to the /MARKERS qualifier will be parsed against the default file specification LSL\$IF:ICHECK.IFF;0. If no features are found to be in error no IFF marker file is produced.

If the /LITES2 qualifier (q.v.) is specified ICHECK creates a LITES2 command file which can be used to direct the user to the position of the suspected error within the Laser-Scan LITES2 graphical editor. The positions supplied in the LITES2 command file to locate the suspected error are the same as used for the location of the marker box in the optional IFF plot file. Note that exactly duplicate points are ignored by LITES2.

The following are examples of the types of entries written to the LITES2 command file:

Crossing segments:

```
%MESSAGE Crossing segments in FSN 302 (302)
%POSITION      765.3      529.89
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
```

Touching segments:

```
%MESSAGE Touching segments in FSN 467 (467)
%POSITION      263.3      456.89
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
```

Open feature end found in border zone:

```
%MESSAGE End in border zone in FSN 670 (55)
%POSITION      591001.3      91456.00
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
```

Feature with points outside neatline:

```
%MESSAGE End outside neatline in FSN 670 (55)
%POSITION      591001.3      91456.00
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
```

Duplicate point:

```
%MESSAGE Duplicate point in FSN 517 (517)
%POSITION      60424.38      639304.67
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
```

If no relevant errors are detected then the empty LITES2 command file is deleted.

EXAMPLES

```
$ ICHECK HYDRO.IFF/LOG/FRT=HYDRO/MARK=HYDROMRK<CR>
%LSLLIB-I-IFFOPENED, DUA0:[IMP.TEST]HYDRO.IFF;3 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[IMP.TEST]HYDROMRK.IFF;2 opened for write
%ICHECK-I-LSTOPNOUT, list file SYS$OUTPUT opened for output

===== ICHECK =====

Input file:   LSL$IF:HYDRO.IFF;

ICHECK invoked by: TOM, Process: Mr Timms, on: 29-JAN-87 17:56:54
Command line:
ICHECK HYDRO.IFF/LOG/FRT=HYDRO/MARK=HYDROMRK

%ICHECK-I-FRTOPN, FRT file LSL$FRT:HYDRO.FRT opened for read
%ICHECK-I-LAYER, layer 1 will be processed
Feature FSN 75 (79) - contains crossing segments at      61.732      61.773
Feature FSN 99 (103) - contains crossing segments at      62.200      61.836
Feature FSN 100 (104) - contains crossing segments at      62.296      62.104
Feature FSN 102 (106) - contains crossing segments at      61.503      61.633
Feature FSN 104 (108) - contains crossing segments at      61.465      61.730
Feature FSN 63 (67) - number of coordinates conflicts with GT of FC 1
Feature FSN 64 (68) - number of coordinates conflicts with GT of FC 1
%ICHECK-I-LAYER, layer 2 will be processed
%ICHECK-I-LAYER, layer 3 will be processed
%ICHECK-I-LAYER, layer 4 will be processed

A total of 5 crossing segments found
A total of 2 FRT errors found
  ELAPSED:    0 00:00:10.40  CPU: 0:00:05.45  BUFIO: 35  DIRIO: 118  FAULTS: 194
$
```

This example demonstrates the use of the /CROSS and /FRT checking qualifiers with the /LOG, /OUTPUT, and /MARKER output qualifiers. The /CROSS and /OUTPUT qualifiers were not typed on the command line but taken as defaults. The user specified HYDRO.IFF as the input file which is expanded to LSL\$IF:HYDRO.IFF by default. As the /LOG qualifier is specified supplementary messages are output when files are successfully opened. In this case the messages confirm that the input file, the MARKER file, specified as HYDROMRK.IFF and the default /OUTPUT (SYS\$OUTPUT) have been opened. The /OUTPUT qualifier causes information about the input file, the users process and the command line to be output and another message confirms that the FRT file specified by the /FRT qualifier (HYDRO and expanded by default to LSL\$FRT:HYDRO.FRT) is opened successfully. The /LOG qualifier also causes messages to be output as each layer is encountered.

In this example a total of 5 crossing segments were found in the specified features at the specified positions and in 2 cases the graphical type indicated by the feature code in the FRT file conflicted with the number of coordinates for the feature. A box feature is created around the errors relating to crossing segments in the MARKER

IFF file but the FRT errors are not shown in the MARKER file. All the errors occurred in layer 1, layers 2 to 4 being free from FRT errors and crossing segments. A summary of the errors is output when the end of file is detected. \$STATUS is set to SS\$_NORMAL.

\$ ICHECK TEST/NOOUT/LITES=IF:TESTLCM/LOG/DUPPLICATES<CR>

%LSLLIB-I-IFFOPENED, DUA0:[IMP.TEST]TEST.IFF;2 opened for read

%ICHECK-I-LITOPNOUT, LITES2 command file IF:TESTLCM.LCM opened for output

%ICHECK-I-LAYER, layer 1 will be processed

%ICHECK-I-LAYER, layer 2 will be processed

%ICHECK-I-LAYER, layer 4 will be processed

Feature FSN 10234 (59) - contains duplicate points

Feature FSN 11 (217) - contains crossing segments at 12412.974 1662.149

Feature FSN 61 (240) - contains crossing segments at 6328.548 4076.191

%ICHECK-I-LAYER, layer 8 will be processed

Feature FSN 25 (81) - contains duplicate points

Feature FSN 10242 (141) - contains duplicate points

Feature FSN 225 (209) - contains crossing segments at 12394.151 3039.299

Feature FSN 10251 (215) - contains crossing segments at 12173.457 1953.396

Feature FSN 10251 (215) - contains crossing segments at 11805.022 2037.562

Feature FSN 97 (221) - contains crossing segments at 8457.673 3727.006

Feature FSN 107 (223) - contains crossing segments at 7668.132 3932.890

Feature FSN 94 (235) - contains crossing segments at 5600.643 5189.009

Feature FSN 10264 (236) - contains crossing segments at 5544.627 5183.751

Feature FSN 100 (237) - contains crossing segments at 5680.049 6547.367

Feature FSN 100 (237) - contains crossing segments at 5695.957 6498.397

Feature FSN 110 (241) - contains crossing segments at 11048.236 312.625

%ICHECK-I-LAYER, layer 9 will be processed

%ICHECK-I-LAYER, layer 10 will be processed

Feature FSN 10161 (220) - contains crossing segments at 12738.729 1219.430

A total of 13 crossing segments found

A total of 3 duplicate points found

ELAPSED: 0 00:00:37.49 CPU: 0:00:33.88 BUFIO: 32 DIRIO: 215 FAULTS: 144

\$

This example demonstrates the use of the /CROSS and /DUPLICATES check qualifiers and the /LOG and /LITES2 output qualifiers. The user specified TEST as the input file which is expanded to LSL\$IF:TEST.IFF by default. The default /OUTPUT qualifier is negated with the /NOOUTPUT qualifier but the user specified /LITES2=IF:TESTLCM to create a LITES2 command file. The LITES2 filename is created in the directory pointed to by the logical name IF and the file extension .LCM is added by default. As the /LOG qualifier is also specified messages confirming the opening of the input IFF file and the LITES2 file are output. The user specified the /DUPLICATE qualifier without either the 'NINT' or 'EXACT' keyword. By default the the /CROSS qualifier is specified and the 'NINT' keyword taken for the /DUPLICATE qualifier.

The /LOG qualifier causes messages to be output as each layer is encountered. In this case a total of 13 crossing segments were found and 3 duplicate points. These errors are noted on the screen and in the LITES2 command file. A summary of the errors is output when the end of

file is detected and \$STATUS is set to SS\$_NORMAL.

```
$ ICHECK/LOG/NEAT=(E=20000,S=0)/NOCROSS TEST<CR>
%LSLLIB-I-IFFOPENED, DUA0:[IMP.TEST]TEST.IFF;2 opened for read
%ICHECK-I-LSTOPNOUT, list file SYS$OUTPUT opened for output

===== ICHECK =====

Input file:   LSL$IF:TEST.IFF;

ICHECK invoked by: TOM, Process: Mr Timms, on: 29-JAN-87 17:59:00
Command line:
ICHECK/LOG/NEAT=(E=20000,S=0)/NOCROSS TEST

%ICHECK-I-LAYER, layer 1 will be processed
%ICHECK-I-LAYER, layer 2 will be processed
%ICHECK-I-LAYER, layer 4 will be processed
%ICHECK-I-LAYER, layer 8 will be processed
Feature FSN 10256 (205) - ends outside neatline at 20184.422  5610.272
%ICHECK-I-LAYER, layer 9 will be processed
%ICHECK-I-LAYER, layer 10 will be processed

A total of 0 features end in border zone
A total of 1 feature ends outside neatline
  ELAPSED:    0 00:00:07.87  CPU: 0:00:05.47  BUFIO: 23  DIRIO: 208  FAULTS: 112
$
```

This example demonstrates the use of the neatline check specified by the /NEATLINE qualifier. The default crossing check is negated with the /NOCROSS qualifier, but the /OUTPUT = SYS\$OUTPUT default is not negated. The /LOG qualifier is selected to send supplementary messages to the screen and the /NEATLINE qualifier has 2 (abbreviated) keywords and values specified: E(ast)=20000 and S(outh)=0. As no border tolerance is specified with the /BORDER qualifier the default of 2% of the maximum range is taken. Checks will therefore be made on open features that start or end within the default border distance of the 2 specified neatlines, or outside either. Messages are output as each file is successfully opened and when each layer is encountered. Information is also output about the user process and the command line typed. One feature (FSN 10256 (205)) is found to start or end outside the neatline and none were found to start or end in the border zone. A summary of the error totals is output and \$STATUS is set to SS\$_NORMAL.


```
$ ICHECK/LOG/FRT=OS/NOCROSS CHELSEA<CR>
%LSLLIB-E-IFFOPEN, IFF error opening file "LSL$IF:CHELSEA.IFF;" for read
%IFF-E-OPEN, error opening IFF file
-RMS-E-FNF, file not found
  ELAPSED:    0 00:00:00.81  CPU: 0:00:00.33  BUFIO: 4  DIRIO: 0  FAULTS: 110
$
```

In this example the user attempted to check the IFF file specified as CHELSEA and expanded to LSL\$IF:CHELSEA.IFF for FRT errors against the FRT file specified as OS and expanded to LSL\$FRT:OS.FRT. The default crossing segment check is negated with the /NOCROSS qualifier and the /LOG qualifier is specified to send supplementary messages to the screen. However, the input file CHELSEA.IFF is not found in LSL\$IF:, either the filename is incorrect or the logical name LSL\$IF: is not pointing to the correct directory or device. \$STATUS is set to SS\$_ABORT.

```
$ ICHECK/LOG/FRT=OS/NOCROSS/DUPL TQ3079A/MARK=LSL$DISK:[TESTDIR]TQTEST<CR>
%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.TEST]TQ3079A.IFF;31 opened for read
%LSLLIB-E-IFFCREATE, IFF error creating file "LSL$DISK:[TESTDIR]TQTEST.IFF"
%IFF-E-PARS, error in IFF filename
-RMS-E-DNF, directory not found
  ELAPSED:    0 00:00:01.04  CPU: 0:00:00.43  BUFIO: 7  DIRIO: 3  FAULTS: 102
```

In this example the user attempted to check the IFF file specified as TQ3079A and expanded to LSL\$IF:TQ3079A.IFF, for FRT errors against the FRT file specified as OS, (expanded to LSL\$FRT:OS.FRT), and for DUPLICATE points. The default crossing segment check is negated with the /NOCROSS qualifier and the /LOG qualifier is specified to send supplementary messages to the screen. The input file is opened successfully however, the output marker file specified as LSL\$DISK:[TESTDIR]TQTEST could not be created. The accompanying messages suggest that the specified directory does not exist. Program execution is terminated on this error and \$STATUS is set to SS\$_ABORT.

```
$ ICHECK/LOG/TOUCH=3/NOCROSS TEST/LAYER=(8:9)/LITES=IF:TESTLCM<CR>
%LSLLIB-I-IFFOPENED, DUA0:[IMP.TEST]TEST.IFF;2 opened for read
%ICHECK-I-LSTOPNOUT, list file SYS$OUTPUT opened for output
```

```
===== ICHECK =====
```

```
Input file:   LSL$IF:TEST.IFF;
```

```
ICHECK invoked by: TOM, Process: Mr Timms, on: 29-JAN-87 18:02:18
```

```
Command line:
```

```
ICHECK/LOG/TOUCH=3/NOCROSS TEST/LAYER=(8:9)/LITES=IF:TESTLCM
```

```
%ICHECK-I-LITOPNOUT, LITES2 command file IF:TESTLCM.LCM opened for output
```

```
%ICHECK-I-LAYER, layer 1 will be ignored
```

```
%ICHECK-I-LAYER, layer 2 will be ignored
```

```
%ICHECK-I-LAYER, layer 4 will be ignored
```

```
%ICHECK-I-LAYER, layer 8 will be processed
```

```
Feature FSN 114 (113) - contains touching segments near 13232.451 1564.254
```

```
Feature FSN 129 (120) - contains touching segments near 15208.502 2218.757
```

```
Feature FSN 129 (120) - contains touching segments near 15205.673 2224.870
```

```
Feature FSN 10064 (134) - contains touching segments near 17250.594 7568.748
```

```
Feature FSN 10065 (135) - contains touching segments near 18016.363 7380.196
```

```
Feature FSN 130 (166) - contains touching segments near 6559.333 285.557
```

```
Feature FSN 59 (167) - contains touching segments near 15398.444 5817.501
```

```
Feature FSN 10251 (215) - contains touching segments near 11801.632 2041.983
```

```
Feature FSN 107 (223) - contains touching segments near 7944.607 3638.555
```

```
Feature FSN 94 (235) - contains touching segments near 5600.565 5202.999
```

```
Feature FSN 10264 (236) - contains touching segments near 5489.138 5214.673
```

```
Feature FSN 100 (237) - contains touching segments near 5667.402 6586.296
```

```
%ICHECK-I-LAYER, layer 9 will be processed
```

```
%ICHECK-I-LAYER, layer 10 will be ignored
```

```
A total of 12 touching segments found
```

```
ELAPSED:    0 00:00:51.56  CPU: 0:00:32.00  BUFIO: 37  DIRIO: 218  FAULTS: 161
```

```
$
```

This example shows a successful run of ICHECK with checks for touching segments specified with the /TOUCH qualifier with a touch tolerance of 3.0 IFF units. The default check for crossing segments is negated with the /NOCROSS qualifier. Output is directed to SYS\$OUTPUT by the default /OUTPUT qualifier and /LOG, while output to a LITES2 command file (IF:TESTLCM.LCM) is also specified. The /LAYER qualifier restricts checking to the specified layers 8 and 9. The messages as each layer is encountered show that only these layers are processed. A total of 12 touching segments were found in these layers at the specified positions. A summary of the error total is output and \$STATUS is set to SS\$_NORMAL.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user. They are often used to supply explanatory information in support of a warning or error message.

LAYER, layer 'integer' will be ignored
LAYER, layer 'integer' will be processed

Explanation: The specified layer in the input IFF file has been found. The message confirms that the current layer is either being ignored or processed.

User action: None.

FRTOPN, FRT file 'file-spec' opened for read

Explanation: The /FRT = 'file-spec' qualifier has been specified and the message is confirming that the FRT (Feature Representation Table) file has been opened.

User action: None.

LITOPNOUT, LITES2 command file 'file-spec' opened for output.

Explanation: The /LITES2 = 'file-spec' qualifier has been specified and the message is confirming that the LITES2 command file has been opened.

User action: None.

LSTOPNOUT, list file 'file-spec' opened for output.

Explanation: The /OUTPUT = 'file-spec' qualifier has been specified and the message is confirming that the output file has been opened.

User action: None.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

INVIFFSTR, invalid IFF file structure in layer 'integer'

Explanation: An incomplete or invalid IFF feature has been encountered in the specified layer. An EF (End Feature) entry has been encountered either outside an IFF feature or, before a FS (Feature Status) or ST (STring) entry has been found. Incorrectly formed TS (Text Status) entries are also considered erroneous.

User action: Examine the input file using IPATCH to find the invalid file structure which caused of the error message. Refer to the IFF User Guide for details of valid IFF file and entry structure. Try to determine the cause of the invalid file structure and if necessary report the circumstances to Laser-Scan.

INVFSNT, invalid IFF FS entry in FSN 'integer' ('integer')

Explanation: An invalid bit structure has been found in the third word of the FS (Feature Status) entry of the specified feature.

User action: Examine the input file using IPATCH to find the invalid FS entry which caused of the error message. Refer to the IFF User Guide for details of valid entry structure. Try to determine the cause of the invalid entry and if necessary report the circumstances to Laser-Scan.

LSTNSPLDEL, listing file not spooled or deleted

Explanation: The program has been unable to spool or delete the listing file specified by the /OUTPUT or /PRINT qualifier.

User action: Check that any listing file specification is correct. Check that the print queues are functioning correctly. If the problem continues notify the system manager.

LSTNSPL, listing file not spooled

Explanation: The program has been unable to spool the listing file specified by the /OUTPUT or /PRINT qualifier.

User action: Check that any listing file specification is correct. Check that the print queues are functioning correctly. If the problem continues notify the system manager.

NOPTSINFT, no points in feature FSN 'integer' ('integer')

Explanation: The specified feature contains an ST entry without any points. ICHECK will ignore any such features.

User action: Try to determine the cause of the ST entry having no points as this is very unusual. If necessary report the circumstances to Laser-Scan.

TOOMNYINV, more than 'integer' invisible steps in feature FSN 'integer'
('integer')

Explanation: The specified feature contains more invisible steps than the ICHECK/IGNORE=INVISIBLE or /IGNORE=POLYGONS functionality can handle in one go. /IGNORE=INVISIBLE and /IGNORE=POLYGONS functionality will not be applied to the offending feature and any touching/intersecting sections of linework caused by invisible moves will be reported.

User action: Note the occurrence of this message and bear it in mind when interpreting the results of the checking run.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ERRCLOINP, error closing input IFF file 'file-spec'
ERRCLOLIT, error closing LITES2 command file 'file-spec'
ERRCLOMRK, error closing Marker file 'file-spec'

Explanation: An error has occurred in closing the specified file.

User action: The accompanying messages should help you to interpret the cause of this error. Check that there is sufficient disk space for the file and that the disk is functioning properly. If the problem continues, notify the system manager.

ERROPFRT, error opening FRT file 'file-spec'
ERROPLIT, error opening LITES2 command file 'file-spec'
ERROPLST, error opening list file 'file-spec'

Explanation: The supplied file specification is erroneous, or the file is nonexistent, or you do not have read access to the file.

User action: The accompanying messages will help you to interpret the cause of the error. Check the file specification for errors, check that you have read access to the file. If the problem continues, notify the system manager.

FAILTRLG, failed to translate logical name 'logical-name'

Explanation: ICHECK is unable to translate the specified logical name.

User action: Use the VMS ASSIGN or DEFINE commands to correctly define the logical name. Re-run ICHECK. Normally the logical names required by ICHECK are defined at user login time. See the "LAMPS Environment Guide" for details of these logical names.

INVBASE, invalid base 'integer' specified

Explanation: An invalid base feature code has been specified either less than 0 or greater than 32762.

User action: Re-run the program with valid base specified.

INVBORD, invalid border 'integer' specified

Explanation: An invalid negative border has been specified. A positive border zone must be specified.

User action: Re-run the program with valid border specified.

INVALFC, invalid Feature Code 'integer' specified

Explanation: An invalid feature-code has been specified either less than 0 or greater than 32767.

User action: Re-run the program with valid feature codes specified.

INVALLAY, invalid LAYER 'integer' specified

Explanation: An invalid layer has been specified either less than 0 or greater than 32767.

User action: Re-run the program with valid layer specified.

INVTUOCH, invalid TOUCH tolerance 'real' specified

Explanation: A invalid negative touch tolerance has been specified. A positive touch tolerance is required.

User action: Re-run the program with valid touch tolerance specified.

STHGTNTH, southern neatline greater than northern neatline

Explanation: The user has given the /NEATLINE=(North=real,South=real) qualifiers but the southern neatline has a greater value than the northern.

User action: Re-run the program with valid neatlines specified.

WSTGTEST, western neatline greater than eastern neatline

Explanation: The user has given the /NEATLINE=(West=real,East=real) qualifiers but the western neatline has a greater value than the eastern.

User action: Re-run the program with valid neatlines specified.

TOOMNINTS, too many intersections in FSN 'integer' ('integer')

Explanation: ICHECK has internal workspace to handle up to 1000 intersections or touches within one feature. This limit has been reached for the specified feature.

User action: If the /TOUCH qualifier has been given check that the tolerance specified is not too large. If this error is not the result of a large touch tolerance then splitting the feature using LITES2 should enable ICHECK to be run again successfully. If this limit causes considerable problems then consult Laser-Scan.

TOOMNPTS, too many points in FSN 'integer' ('integer')

Explanation: ICHECK can handle IFF features containing up to 10000 points. This limit has been reached for the specified feature.

User action: Split the feature using LITES2 and re-run ICHECK. If this limit causes considerable problems then consult Laser-Scan.

UNEXPEOF, unexpected end of IFF file

Explanation: This message indicates there is something seriously wrong with the IFF file which has caused immediate termination of the program. ICHECK has detected the end of the IFF file, but has not detected an IFF 'EJ' entry.

User action: Use IMEND on the file, which will correctly position the EOF marker and insert an EJ entry at the end of the file. Re-run ICHECK on the corrected file.

MESSAGES (OTHER)

In addition to the above messages which are generated by ICHECK itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 3

MODULE IDIFFERENCE

MODULE IDIFFERENCE

REPLACES DAMP module CMPIFF

FUNCTION

IDIFFERENCE is the IMP IFF file comparison utility. It compares the file header entries (the RA,HI,etc), and then for each layer that is present in both files, it compares the features. Features are compared if they have the same FSN (Feature Serial Number). If several features in one layer have the same FSN, only the last encountered is compared. Note that layers and features within the files need not be sorted.

FORMAT

\$ IDIFFERENCE master-file-spec revision-file-spec

Command qualifiers

/DEBUG
/FRT=file-spec
/[NO]IGNORE=([FSFC],[FSSTATUS],[FSPCTEXT],
 [FSUSERWORD],[HI],[MD],[NS],
 [TH])
/LOG
/LOOKAHEAD=integer
/[NO]OS
/OUTPUT=file-spec
/[NO]PLOT=([COORDINATES],[ATTRIBUTES],[OTHER])
/PRINTER
/SUMMARY
/TOLERANCE=([DEFAULT=real],[FACTOR=real])

Defaults

No debug information
No FRT
/NOIGNORE

No logging
/LOOKAHEAD=20
/NOOS
Output to SYS\$OUTPUT
/NOPLOT
Output is not printed
See text
See text

PROMPTS

_File1: master-file-spec
_File2: revision-file-spec

PARAMETERS

master-file-spec

- specifies the master IFF file with which comparison is to be made. Any part of the file specification which is not supplied will be taken from the default specification LSL\$IF:IFF.IFF;0

The master IFF file is referred to as "file 1" by the program, and is regarded as the 'correct' file.

revision-file-spec

- specifies the IFF file which is to be compared with the master file. Any part of the file specification which is not explicitly given will be taken from the parsed master file specification.

If the revision-file-spec is omitted, then IDIFFERENCE will attempt to use the previous version of the master file as the revision file. It generates the revision-file-spec by subtracting 1 from the master file's version number. If no version number is given in the master-file-spec, then IDIFFERENCE will look for version ";-1", which is VMS short-hand for the previous version.

The revision file is referred to as "file 2" by the program.

COMMAND QUALIFIERS**/DEBUG=integer**

- this qualifier requests various types of debugging messages. It is documented here for completeness, but is not recommended for use by customers as it will produce voluminous output.

/FRT=file-spec

- this qualifier specifies an FRT file which will be read to provide ACD definitions. These specify customer or application specific AC and CB column types and names. If no FRT file is read, then only the Laser-Scan default ACD definitions will be understood by IDIFFERENCE. This means that only AC and CB data with these ACD types will be compared according to their datatype, and labelled correctly in difference listings. If an FRT file is specified, its specification will be filled out from the default file-spec LSL\$FRT:.FRT;0

/IGNORE=(FSFC,FSSTATUS,FSPCTEXT,FSUSERWORD,HI,MD,NS,TH)
/NOIGNORE (default)

- this qualifier specifies whether an entry/or list of entries in the input IFF files is/are to be ignored during the comparison process of IDIFFERENCE.

The following keyword arguments to the /IGNORE qualifier specify the IFF entries, or part entries, which are to be excluded from the file comparison:

Part-entry keywords:

- o FSFC - ignore the feature code (FC) field of the FS entry of all features.
- o FSSTATUS - ignore the status field of the FS entry of all features.

- o FSPCTEXT - ignore the process code/text field of the FS entry of all features.
- o FSUSERWORD - ignore the user-word field of the FS entry of all features.

File level entry keywords:

- o HI - ignore the history entries (HI) of the IFF files.
- o MD - ignore the Map Descriptor (MD) entries of the two IFF files.
- o NS - ignore the New Section entries (NS) of the two IFF files.

Feature level entry keywords:

- o TH - ignore the thickness entry (TH) of all features in the two IFF files.

The default value for this qualifier is /NOIGNORE.

/LOG

- this qualifier results in supplementary messages being sent to SYS\$OUTPUT. Messages are produced when files are successfully opened.

/LOOKAHEAD=integer

- this qualifier sets the acceptance window for point comparison within a feature. If there are no more than the given number of different points between equal points, then the feature being compared is considered to match (compare with the /MAXIMUM_DIFFERENCES and /WINDOW qualifiers of the VMS DIFFERENCE utility). The default value for /LOOKAHEAD is 20.

/OS

/NOOS (default)

- this qualifier forces IDIFFERENCE into Ordnance Survey (UK) mode. This differs from the default mode of IDIFFERENCE in that:
 - o the contents of the HI (HISTORY), MD (Map Descriptor), CC (Cubic Coefficient) and CP (Control Point) entries are ignored. Absence of these entries is still diagnosed, however.
 - o the first few entries in the MH (Map Header) entry will be treated as if the MH is in OS format. This means that differences in the OS job number, map reference and source scale fields will be diagnosed as such.
 - o the contents of layers 0, 11 and 32 are ignored - a message is output if any of these are encountered.

/OUTPUT=file-spec

- by default all output is to SYS\$OUTPUT. This qualifier specifies that output should be redirected to the named file, with an additional header at the start of the file noting the command line, etc. If 'file-spec' is omitted, then a file IDIFFERENCE.LIS will be generated. Note that /LOG messages are still output to SYS\$OUTPUT, and not to the /OUTPUT file.

/PLOT=(COORDINATES,ATTRIBUTES,OTHER)

/NOPLOT (default)

- this qualifier requests that IDIFFERENCE produce three IFF files to reflect the differences between the master and revision files. The plot IFF files have the same file-spec as the master file, except that the file name of each file has extra characters added to it:
 - o "_DEL" - features deleted - that is, present in the master file but not in the revision file
 - o "_CRE" - features created - that is, present in the revision file but not in the master file
 - o "_DIF" - features present in both the revision and master files, but different from each other

The arguments to the /PLOT qualifier are used to specify which features are to be output to the third (difference) file.

- o If the COORDINATES option is selected, then it outputs features
 - o with differing X, Y or Z coordinates
 - o with differing visibility (ie where a line segment is visible in one file, and invisible in the other)
 - o with differing numbers of points
- o If the ATTRIBUTES option is selected, then it outputs features
 - o with differing point attributes (other than X, Y or Z)
 - o with differing numbers of point attributes (ie CB columns)
 - o with differing ACs
- o If the OTHER option is selected, then it outputs features with any other difference.
- o If no options are specified (ie /PLOT by itself) then the effect is as if all options had been selected - features differing in any way will be output.

In addition to any features which are found to be different, the plot files also contain a frame representing the area covered by the two files. The frame is placed in a separate layer. The number of this layer is determined using the maximum layer number from the 2 input files, plus one.

/PRINTER

- if this qualifier is specified, then program output will be written to a file, which will be queued for printing on SYS\$PRINT and then deleted when IDIFFERENCE finishes. If /OUTPUT has also been specified, then the file concerned is the /OUTPUT file. Otherwise, a default file called IDIFFERENCE.LIS will be used.

/SUMMARY

- if this qualifier is specified, then only the final summary of differences will be output.

/TOLERANCE=(DEFAULT=real,FACTOR=real)

- this qualifier specifies the tolerances to be used in determining whether real (floating point) attributes (including X and Y) are to be considered as identical. The numbers specified may not be negative. If an attribute has an interval defined in its ACD, then the tolerance used for comparison is the FACTOR value multiplied by that interval. Otherwise, the DEFAULT value is used. If the user does not specify the FACTOR value, then a default of 1.0 is used. If the user does not specify the DEFAULT value, then IDIFFERENCE uses the maximum coordinate range divided by 10000. The maximum coordinate range is deduced by taking the range entries of both input files into consideration.

When comparing real attributes (and for line strings this includes the X and Y coordinates), two values a1 and a2 are considered equal if $ABS(a1-a2)$ is less than or equal to the relevant tolerance.

DESCRIPTION**General**

IDIFFERENCE takes two IFF files as input, and detects

- o differences in the file header entries,
- o missing layers,
- o missing features,
- o differences in features, including
 - o changes in feature code,
 - o differing ancillary codes,
 - o differing texts and text classifications,
 - o differing rotations, and
 - o changes in the coordinates and coordinate attributes within features.

Depending on the command qualifiers all the information regarding these differences will be output to SYS\$OUTPUT or to a listing file. If there is more than one map in either input file, only the first will be compared. IDIFFERENCE expects FSNs (Feature Serial Numbers) to be unique within each layer.

What is compared

IDIFFERENCE compares the

- o RA (Range),
- o HI (History),
- o MH (Map Header),
- o CP (Control Points),
- o CC (Cubic Coefficients),
- o first NS (New Section),
- o features within layers that occur in both files

IDIFFERENCE ignores junction entries. Specifically, it does not look for SH or JB entries amongst the file header entries, and it ignores any JP entries found within a feature.

Various obsolete entries are not recognised - these are

- o TC (Transmitted Comment)
- o CH (literal CHaracter),
- o SS (Symbol Select),
- o SL (Symbol Library)

Voids (VO entries) are always ignored.

Note that when comparing the 'file header' entries (ie everything before the first layer of the files - the RA,HI,etc), the order of the entries is not relevant to IDIFFERENCE. Before reading in each header entry, IDIFFERENCE rewinds to the start of the file and then searches forwards for the entry.

IDIFFERENCE also ignores those entry/list of entries that are explicitly specified in the /IGNORE qualifier - these are

- o FSFC (Feature Code),
- o FSSTATUS (Feature Status),
- o FSPCTEXT (Feature PC/TEXT),
- o FSUSERWORD (Feature User-Word),
- o HI (Map History),
- o MD (Map Descriptor),
- o NS (New Section),
- o TH (Thickness)

Comparing real values

When comparing floating point quantities, a direct check for equality is not sensible. Instead, IDIFFERENCE checks that real values are the same within a tolerance. Three tolerances are used:

- o The **standard real tolerance** is 10^{-6} (ten to the power of minus six), and is used when comparing real quantities in the file header entries, and in the RO entry.
- o The **default real tolerance** is used for comparing X, Y and Z coordinate values. It is used for all CB and AC real values where there is no interval specified in the ACD for that quantity (which

is why it is used for X, Y and Z). It is either specified by the user with the /TOLERANCE qualifier, or is taken as being 1/1000th of the maximum range of the data.

- o The **factor real tolerance** is used for comparing CB and AC real values where an interval was specified in the ACD for that quantity. It is defined as a multiple of that interval, where the multiple is either specified by the user with the /TOLERANCE qualifier, or is taken as being 1.0.

The RA (RAnge) entries

The RA entries are compared to see whether the range of coordinates described is the same. Range elements are considered equal if they differ by less than the standard real tolerance.

The HI (HIStory) entries

The HI entries are checked to see whether they contain the same number of history records, and whether the history records within each entry are identical. If they do not match then the complete HI entry from each file is output.

HI entries are ignored only when the qualifier /IGNORE=HI is specified.

The MH (Map Header) entries

The fields of the MH are compared in turn. Differences are reported on the basis of which longword field of the MH does not match, and each mismatched field is output as a hexadecimal quantity.

Note that if the /OS qualifier is selected, then the first four fields in the MH (job number, scale and SW corner easting and northing) are recognised separately, and output in an appropriate form.

The MD (Map Descriptor) entries

The MD entries are checked to ensure that they are of a valid type, and that they are both the same length. If they are compatible, then their contents are compared, and if any of the fields vary then the complete map descriptor will be output. Real MD elements are considered equal if they differ by less than the standard real tolerance, integer elements are required to match exactly.

MD entries are ignored only when the qualifier /IGNORE=MD is specified.

The NS (New Section) entries

IDIFFERENCE (like most of the other IMP utilities) is not intended to deal with files that have just been digitised, and are thus split into multiple sections. Files produced by LASERAID should be processed with LAPROCESS to perform any transformations necessary, and all new files should be processed by IMERGE or LITES2 to consolidate the separate sections in which files are normally digitised.

IDIFFERENCE assumes that there is only one section in the IFF files being compared, and only compares the first NS entry in each file. If the entries do not match, then they will both be output.

NS entries are ignored only when the qualifier /IGNORE=NS is specified.

The CC (Cubic Coefficients) entries

Again, it is assumed that the file has been processed to consolidate sections. Thus IDIFFERENCE will only look for one CC entry in each file. The entries for the two maps are compared. If any differences are found the whole of the CC entry for each file is output. Coefficients are considered equal if they differ by less than the standard real tolerance.

The CP (Control Point) entries

Again, it is assumed that the file has been processed to consolidate sections. Thus IDIFFERENCE will only look for one CP entry in each file. If the control points of the two files are different in any respect then the complete CP entry for each file is output. Control points are considered equal if they differ by less than the standard real tolerance.

The Layers

IDIFFERENCE does not require that layers be in order within the IFF files being compared, nor does it require that they be 'continuous'. That is, if a file has several layers with the same layer number (which is stored in the NO entry), then IDIFFERENCE will recognise the data within them as all being part of the same layer.

After comparing the file header entries (above), IDIFFERENCE scans the input files to establish where all of the NO (New Overlay) entries are. It then compares the layers in ascending numerical order. If a layer is only present in one file, then this is reported (and no comparison is made of the features within that layer).

The Features

Within each layer, features are compared according to their FSN (Feature Serial Number). That is, if a feature in layer 3 in one file has FSN 77, it will be compared with the feature with FSN 77 in layer 3 in the other file. Internal sequence numbers are ignored by IDIFFERENCE.

IDIFFERENCE does not require that features be sorted into FSN order - when it starts comparing a layer, it builds an internal table of the addresses of each feature. It can then compare them in ascending order of FSN by looking up the location of the relevant feature in each file.

If there are several features with the same FSN in a layer, then IDIFFERENCE will use the last of the features (that is, the one nearest the end of the IFF file) as the one to be compared. Each time

that a duplicate FSN is encountered, IDIFFERENCE outputs a message warning that it is using the latest version of the feature.

When features are compared, they may be counted as **identical**, **different** or **unmatched**. The summary of differences at the end of the program lists how many features were in each category. A single feature is only counted once, and if it is found to be both **different** and **unmatched** then it is counted as **unmatched**.

For each feature, the FS (Feature Status), AC (Ancillary Code), TS (Text Status), TH (Text Height or THickness), RO (ROtation), TX (TeXt) and CB, ST or ZS (coordinate block or string) entries are compared. A feature is rejected as malformed if it does not contain an FS and at least one CB (or ST/ZS) entry, and if it does not end with an EF entry.

The entries are compared as follows:

- o FS (Feature Status)

The text/symbol bits in word three of the FS entry are decoded for each feature to determine if it is a text, a symbol or a line. If both features do not have the same bits set, they are considered **unmatched**, and comparison of the FS entry is abandoned. User must note that this comparison will be performed only when the command qualifier /IGNORE=FSPCTEXT is not specified.

If the text/symbol bits match, then the rest of the FS entries are compared. Any differences result in the features being considered **different**.

- o The first word is the feature code. This word is ignored when the qualifier /IGNORE=FSFC is specified.
- o The second word contains status bits, and is only compared for line features - it is displayed in hexadecimal if it fails to match. This word is ignored when the qualifier /IGNORE=FSSTATUS is specified.
- o For text features, the third word contains the text position code, type style and name category. These are compared as separate items. For linear and symbol features, the third word contains the process code. This word is ignored when the qualifier /IGNORE=FSPCTEXT is specified.
- o The fourth word is not normally used, but is checked just in case, and displayed in hexadecimal if it differs. This word is ignored when the qualifier /IGNORE=FSUSERWORD is specified.

- o ACs (Ancillary Codes)

If only one feature has AC entries, then a warning message is output and the features are considered **different**.

If both features contain the same number of ACs, these are compared in the order that they occur. If any difference is found, or if the features have a different number of ACs, then all the ACs for both features are printed out, and the features are **different**.

- o TS (Text Status)

The TS entry starts a text component. This is effectively a 'sub-text-feature', and may contain all of the entries found within a normal text feature, except for FS and AC entries. A text component is terminated by the next TS or by the EF (End of Feature) entry, and is compared as if it were indeed a sub-feature - that is, each text component is compared in turn, in the order that they are encountered in the features.

If only one of the features being compared has a text component, then a warning is given and the features are considered **different**. Note that in this case, the feature without a text component will be compared to the entries before the text components in the other. This will generally produce a series of "entry 'entry' missing" messages, as the feature with text components should only have an FS and possibly ACs before its first TS entry.

If the features contain differing numbers of text components, then as many as possible are compared. The features will be considered **different**.

For each text component, the TS entry is compared, and then the rest of the text component is compared, in the same way as a normal feature. Words one and three of the TS entry are compared in the same way as words one and three of an FS entry (qv), except that word three must have the text bit set. Words two and four are reserved, and should be zero.

- o TH (THickness or Text Height)

If only one feature has a TH entry, then a warning message is output, and the features are considered **different**.

If both features contain TH entries, they are compared. If they do not match, then the features are considered **different**.

TH entries are ignored only when the qualifier /IGNORE=TH is specified.

- o RO (ROtation)

If only one feature has an RO entry, then a warning message is output, and the features are considered **different**.

Otherwise, the two RO entries are compared, and if they do not match the feature is considered **different**. Note that the rotations are considered to match if they differ no more than the standard real tolerance.

- o TX (TeXt)

If only one feature has a TX entry, then a warning message is output, and the features are considered **different**.

Otherwise, if the texts have a different number of characters, or if the characters in the texts do not match, then the differing texts are printed out, and the features are considered to be **different**.

- o CB (Coordinate Block) and ST or ZS (coordinate strings)

Every feature must contain at least one X,Y coordinate pair, held in a CB, ST or ZS entry.

IFF revision level 0 files hold their coordinates in ST or ZS (2d or 3d coordinate string) entries. IFF revision level 1 files use CB (Coordinate Block) entries instead. IDIFFERENCE converts all ST and ZS entries to CB entries internally - it thus never actually sees any of the older type of entry.

The actual distribution of points within coordinate blocks is not important - that is, if one feature has two CBs of 150 and 30 points, and the other has one CB containing 180 points, IDIFFERENCE will still compare each point with its corresponding point in the other file.

Before comparing the CB entries for a feature (or text-component), IDIFFERENCE scans the relevant CB entries, to determine how many columns (attributes) are used within the feature, what they are, and how many points there actually are. If the columns (attributes) in the two features differ, a warning will be output, and the features will be considered **different**.

Each point x_1, y_1, \dots from file 1 is compared with the corresponding point x_2, y_2, \dots in file 2, and they are considered to match if all of the attributes (including the X and Y coordinates) are found to match. Real attributes are compared using the appropriate tolerance - for X, Y and Z coordinates this will be the **default real tolerance**.

If the two points match, but their visibility does not (that is, there is an invisible line segment ending at one, and a visible line segment ending at the other), then they will be output, and the feature will be considered **different**.

If the points do not match, then the point in file 1 is compared with the next point in file 2, and so on until 'lookahead' points from file 2 have been compared. 'lookahead' is the value specified with the /LOOKAHEAD qualifier, and defaults to 20. If the points have still not matched, the next point from file 1 is checked against these same 'lookahead' points from file 2 (note that this includes the point in file 2 which did not match initially). This too is repeated until 'lookahead' points from file 1 have been tried.

If at any stage in this process a match is found, then comparison of the feature continues from the matched points, and a list of the unmatched points (and the first matched point) is output. In this case the features are considered **different**.

If a match is not found, the comparison is abandoned and a list of the unmatched points tried is output. The features are considered **unmatched**. Features are also considered **unmatched** if all of their points fail to match, even if the number of points is less than 'lookahead'.

Difference in CBs are counted slightly differently than differences in other entries within a feature. Four separate counts are kept:

- o features that have differing columns (attributes) in their CBs.
- o features that differ in geometry - this is taken to mean that their CBs differed in X, Y or Z, or in visibility.
- o features that differ in the other attributes of a point.
- o features that differ because one feature has more points left, after all other comparisons have been completed.

Note that comparison of one feature might be counted towards more than one of these totals.

Summary of differences

When the comparison is complete a summary of the differences found is output. This takes the form:

Summary of differences

The following header entries did not match:		(1)
RA (RAnge)		
MD (Map Descriptor)		
'integer' different layers were found		(2)
Number of layers compared	'integer'	(3)
Missing layers in file 1	'integer'	(4)
Missing layers in file 2	'integer'	(5)
'integer' features required comparison:		(6)
Identical	'integer'	(7)
Different	'integer'	(8)
Unmatched	'integer'	(9)
How features differed by entry type:		(10)
FS entry	'integer'	
CB entries - geometry	'integer'	
CB entries - attributes	'integer'	
Missing features in file 1	'integer'	(11)
Missing features in file 2	'integer'	(12)
Duplicated FSNs in file 1	'integer'	(13)
Duplicated FSNs in file 2	'integer'	(14)

If any of the file header entries was not matched, then line (1) is output, followed by a list of those entries that were different. In this example, the RA and MD entries differed.

Line (2) notes how many different layers were present in the files. Line (3) then records how many of these were present in both files, and lines (4) and (5) indicate how many layers were in one file but not the other. Note that the quantities in lines (3) to (5) sum to the number in line (2).

Line (6) indicates how many features were actually compared. This does not include any features in layers that were ignored, nor does it include features that were missing, or ignored because of duplicated FSNs. Line (7) shows how many features were considered to be **identical** - ie no differences at all. Line (8) shows the number of **different** features, and line (9) the number of **unmatched** features.

If any of the features differed, then line (10) will be output, followed by an indication of how many features differed by each entry type. Note that difference within CB entries (and thus also ST or ZS entries) are counted in four ways - see the section on the comparison of CB entries above. In this example, the only differences observed were in FS entries, and in the actual geometry and attributes of CB entries.

Lines (11) and (12) record how many features were present in one file, but not in the other. Note that if a layer in a file contained several features with the same FSN (Feature Serial Number), and there are no features with that FSN in that layer in the other file, then this only contributes a count of 1 to the 'missing' total.

Lines (13) and (14) record how many features were ignored because they had the same FSN as another feature in the same layer.

EXAMPLES

\$ IDIFFERENCE TJI1 TJI2 /FRT=RONACD<CR>

File 1 is LSL750\$DUA0:[LSL.IFF]TJI1.IFF;4

File 2 is LSL750\$DUA0:[LSL.IFF]TJI2.IFF;15

Comparing with lookahead 20, point tolerances: default 0.03412, factor 1.00

HI (HISTORY) entry missing in file 1

HI (HISTORY) entry missing in file 2

Comparing layer 1

Feature 1 line feature

=====

CB data - differences for file 1

Point	X	Y
200:	17.4755	87.266922
201:	18.21361	91.21772
202:	18.247231	96.905792

...equal

CB data - differences for file 2

Point	X	Y
200:	18.247231	96.905792

...equal

Feature 2 line feature

CB entries - some columns will not be compared

Column Z (93) is only present in file 2

Feature 3 line feature

FS entry: Feature Code - in file 1 = 0, file 2 = 1

=====

TH entry: file 1 = 0, file 2 = 5

=====

CB data - differences for file 1

Point	X	Y	CAPTURE_XY
6:	50.737034	164.85843	62
7:	50.983398	164.81369	62
8:	51.230087	164.77017	62
9:	51.477097	164.72789	62
10:	51.724419	164.68683	62
11:	51.972038	164.64699	62

...equal

CB data - differences for file 2

Point	X	Y	CAPTURE_XY
6:	50.737034	164.85843	61
7:	50.983398	164.81369	61
8:	51.230087	164.77017	61
9:	51.477097	164.72789	61
10:	51.724419	164.68683	61
11:	51.972038	164.64699	62


```

...equal
=====
CB data - differences for file 1
Point          X          Y          CAPTURE_XY
  23:         54.963589    164.26456         62
  24:         55.214287    164.24066         62
  25:         55.465157    164.21799         62
  26:         55.716187    164.19653         62
...equal
-----
CB data - differences for file 2
Point          X          Y          CAPTURE_XY
  23:         54.963589    164.26456         57
  24:         55.214287    164.24066         57
  25:         55.465157    164.21799         57
  26:         55.716187    164.19653         62
...equal
=====
CB data - differences for file 1
Point          X          Y          CAPTURE_XY
  27:         55.967377    164.17632         62
  28:         56.218708    164.15732         62
...equal
-----
CB data - differences for file 2
Point          X          Y          CAPTURE_XY
  27:         55.967377    164.17632         57
  28:         56.218708    164.15732         62
...equal
*****
Feature 5                symbol feature
FS entry: differing text/symbol codes - in file 1 =  symbol, in file 2 =  line
=====
--- unmatched ---
*****
Feature 6                symbol feature
=====
AC entries: in file 1
  AC Name          ( Type)          Value Text
  AC CHANGE_DATE    ( 1002)          10-DEC-1987  " "
  AC DATE_EXAMPLE   (12007)          1-JAN-1900  " "
-----
AC entries: in file 2
  AC Name          ( Type)          Value Text
  AC CHANGE_DATE    ( 1002)          10-DEC-1987  " "
  AC DATE_EXAMPLE   (12007)          1-JAN-1988  " "
  AC ?              ( 2009)          8 "K200/3/4-B"
*****
Feature 7                line feature
=====
CB data - differences for file 1
Point          X          Y          Z          SURVEY_DATE
  1:           5.0          3.0          0.0        23-DEC-1962
  2:           6.0          4.0          1.0        14-NOV-1965
...equal
-----

```

CB data - differences for file 2

Point	X	Y	Z	SURVEY_DATE
1:	5.0	3.0	0.0	5-JAN-1988
2:	6.0	4.0	1.0	14-NOV-1965

...equal

=====

CB data - differences for file 1

Point	X	Y	Z	SURVEY_DATE
4:	1.0	9.0	8.0	5-MAY-1966
5:	9.0	1.0	2.0	7-JUL-1945

(ends unequal)

CB data - differences for file 2

Point	X	Y	Z	SURVEY_DATE
4:	8.0	9.0	8.0	5-JAN-1988
5:	9.0	10.0	10.0	5-JAN-1988

(ends unequal)

Feature 8 line feature

=====

CB data - differences for file 1

21 points left in file 1

=====

--- unmatched ---

Feature 9 line feature

=====

CB data - differences for file 1

18 points left in file 1

Summary of differences

1 layer was found

Number of layers compared	1
Missing layers in file 1	0
Missing layers in file 2	0

9 features required comparison:

Identical	1
Different	6
Unmatched	2

How features differed by entry type:

FS entry	2
AC entries	1
CB entries - columns	1
CB entries - geometry	3
CB entries - attributes	2
CB entries - point count	2
TH entry	1

Missing features in file 1	0
----------------------------	---

Missing features in file 2	0
----------------------------	---

Duplicated FSNs in file 1	0
---------------------------	---

```
ELAPSED:      0 00:00:17.53  CPU: 0:00:12.63  BUFIO: 171  DIRIO: 179  FAULTS: 168
```

- o In feature 1, file 1 contains two extra points (numbered as 200 and 201). All points before these are equal, and all points afterwards are also equal.
- o In feature 2, only one file contains a Z column, and this is therefore not compared. Apparently, the X and Y coordinate data matches.
- o In feature 3, there are several differences. The feature code does not match, the contents of the TH entry differs, and two blocks of coordinate data have had their CAPTURE_XY attribute changed.
- o In feature 5, the FS entries contained differing text/symbol bits. The features are thus regarded as being of different graphical type, and therefore **unmatched**. The rest of the feature seems to match.
- o In feature 6, an extra AC is found. Note that all of the ACs are reported.
- o In feature 7, there are various differences in the CB entry. Note that the last point of the feature is different, and thus the "(ends unequal)" message is used, rather than "etc..."
- o In feature 8, the initial points match, but there are 21 more points in file 1. Since this is more than the lookahead (which defaults to 20), the feature is considered **unmatched**
- o In feature 9, there are only 18 extra points, so although the features are different, they are not unmatched.

The summary then reports on the file in general. One layer was compared, and 9 features. Of these, 1 matched, 2 were considered unmatched, and 6 differed in less severe ways. The differences are then itemised according to entries within a feature. Note that the CB differences are split up - one feature had differing numbers of columns between the files, 3 features had differences in the geometry (ie X,Y,Z or visibility), 2 features had differences in CB attributes (all other columns), and 2 features differed in the number of points (or rows) left after the CBs had been compared. Note that although the matched/unmatched/different values add up to the number of feature compared, the entry summary does not, as one feature may differ in several ways.

\$ IDIFFERENCE RWRB RWRD /OUTPUT=RWR<CR>

ELAPSED: 0 00:00:20.26 CPU: 0:00:10.42 BUFIO: 64 DIRIO: 295 FAULTS: 1896

\$ TYPE RWR.LIS<CR>

===== I D I F F E R E N C E =====

IDIFFERENCE invoked by TONY using terminal RTA1: at 5-JAN-1988 14:03:05.29

Command line:

IDIFFERENCE RWRB RWRD/OUTPUT=RWR

=====

File 1 is LSL750\$DUA0:[LSL.IFF]RWRB.IFF;2

File 2 is LSL750\$DUA0:[LSL.IFF]RWRD.IFF;1

Comparing with lookahead 20, point tolerances: default 0.10000, factor 1.00

RA (RAnge) entry in file 1

0.0000	1000.0001	-0.0000	1000.0001
--------	-----------	---------	-----------

RA (RAnge) entry in file 2

0.0000	1000.0001	-0.0000	1000.0001
--------	-----------	---------	-----------

HI (HIstory) entry missing in file 1

HI (HIstory) entry missing in file 2

MH (Map Header) entry

longword 4 in file 1 = 00079D38, file 2 = 0007A120 (hex)

longword 23 in file 1 = 00079D38, file 2 = 0007A120 (hex)

Map descriptors - type 2 MDs do not match

File 1:

- Local origin: 599500.0000, 499000.0000

- Map scale: 1250.0000

- Projection: 101 (UK national grid)

- Spheroid: 9 (Airy)

- Units: 2 (metres)

File 2:

- Local origin: 599500.0000, 499500.0000

- Map scale: 2500.0000

- Projection: 101 (UK national grid)

- Spheroid: 9 (Airy)

- Units: 2 (metres)

Comparing layer 1

Comparing layer 11

Summary of differences

The following header entries did not match:

RA (RAnge)

MH (Map Header)

MD (Map Descriptor)

```

2 different layers were found
  Number of layers compared          2
  Missing layers in file 1           0
  Missing layers in file 2           0

53 features required comparison:
  Identical                          53
  Different                          0
  Unmatched                          0

Missing features in file 1           0
Missing features in file 2           0

Duplicated FSNs in file 1            0
Duplicated FSNs in file 2            0

```

In this example, the output is directed to a file, and this file is then typed. The files differed in their range, map header and map descriptor entries only, as reported in the summary at the end. Note that, for the map descriptor, all of the information compared is reported, even though it is only the scale and local origin that differ.

```

$ IDIFFERENCE FILE1 FILE2 /LOG/PLOT=COORDINATES/OUTPUT=FILE<CR>
%LSLLIB-I-IFFOPENED, LSL750$DUA0:[LSL.IFF]FILE1.IFF;2 opened for read
%LSLLIB-I-IFFOPENED, LSL750$DUA0:[LSL.IFF]FILE2.IFF;1 opened for read
%LSLLIB-I-LOGOPNOUT, log file FILE.LIS opened for write
%LSLLIB-I-IFFOPENED, LSL750$DUA0:[LSL.IFF]FILE2_DEL.IFF;2 opened for write
%LSLLIB-I-IFFOPENED, LSL750$DUA0:[LSL.IFF]FILE2_CRE.IFF;2 opened for write
%LSLLIB-I-IFFOPENED, LSL750$DUA0:[LSL.IFF]FILE2_DIF.IFF;2 opened for write
ELAPSED:      0 00:01:28.09  CPU: 0:01:17.63  BUFIO: 35  DIRIO: 912  FAULTS: 1720

```

In this example, the user has specified /LOG, and thus the opening of all files is reported. The /PLOT=COORDINATES qualifier means that coordinate differences will be output to the output IFF files FILE2_DEL, FILE2_CRE and FILE2_DIF. The /OUTPUT qualifier redirects the actual difference output to the text file FILE.LIS

```

$ IDIFFERENCE TL1659NE.IFF1 TL1659NE_AFTER.IFF1 /SUMMARY<CR>

```

```

Summary of differences
*****

```

```

The following header entries did not match:
  HI (HISTORY)

```

```

1 layer was found
  Number of layers compared          1
  Missing layers in file 1           0
  Missing layers in file 2           0

427 features required comparison:
  Identical                          422
  Different                          4

```

Unmatched	1
-----------	---

How features differed by entry type:

CB entries - geometry	5
CB entries - attributes	5

Missing features in file 1	0
----------------------------	---

Missing features in file 2	0
----------------------------	---

Duplicated FSNs in file 1	0
---------------------------	---

Duplicated FSNs in file 2	0
---------------------------	---

ELAPSED: 0 00:01:23.27 CPU: 0:01:14.67 BUFIO: 33 DIRIO: 905 FAULTS: 1526

In this example, the /SUMMARY qualifier is used to suppress the main difference output, so that only the final summary is produced.

\$ IDIFFERENCE CODETEST CODESORT /OS<CR>

File 1 is LSL750\$DUA0:[OS.IFF]CODETEST.IFF;3

File 2 is LSL750\$DUA0:[OS.IFF]CODESORT.IFF;1

Comparing with lookahead 20, point tolerances: default 0.10000, factor 1.00

HI (HISTORY) entry missing in file 1

HI (HISTORY) entry missing in file 2

Comparing layer 1

Ignoring layer 11

Summary of differences

2 different layers were found

Number of layers compared	2
Missing layers in file 1	0
Missing layers in file 2	0

706 features required comparison:

Identical	706
Different	0
Unmatched	0

Missing features in file 1	0
----------------------------	---

Missing features in file 2	0
----------------------------	---

Duplicated FSNs in file 1	0
---------------------------	---

Duplicated FSNs in file 2	0
---------------------------	---

ELAPSED: 0 00:01:26.54 CPU: 0:01:16.78 BUFIO: 35 DIRIO: 1394 FAULTS: 194

5

This example illustrates the result of comparing two files with identical contents. Note that we do not know that the order of occurrence of features within the file is the same, since comparison

is by feature serial number within each layer.

\$ IDIFFERENCE TJI1 TJI3<CR>

File 1 is LSL750\$DUA0:[LSL.IFF]TJI1.IFF;4

File 2 is LSL750\$DUA0:[LSL.IFF]TJI3.IFF;1

Comparing with lookahead 20, point tolerances: default 0.03412, factor 1.00

HI (HISTORY) entry missing in file 1

HI (HISTORY) entry missing in file 2

Comparing layer 1

Feature 1 line feature

=====

CB data - differences for file 1

Point	X	Y
-------	---	---

2:	2.8696361	20.28903
----	-----------	----------

3:	4.7384071	28.668381
----	-----------	-----------

...equal

CB data - differences for file 2

Point	X	Y
-------	---	---

2:	2.8729999	20.188999
----	-----------	-----------

3:	4.7371998	28.669201
----	-----------	-----------

...equal

Summary of differences

1 layer was found

Number of layers compared	1
---------------------------	---

Missing layers in file 1	0
--------------------------	---

Missing layers in file 2	0
--------------------------	---

9 features required comparison:

Identical	8
-----------	---

Different	1
-----------	---

Unmatched	0
-----------	---

How features differed by entry type:

CB entries - geometry	1
-----------------------	---

Missing features in file 1	0
----------------------------	---

Missing features in file 2	0
----------------------------	---

Duplicated FSNs in file 1	0
---------------------------	---

Duplicated FSNs in file 2	0
---------------------------	---

ELAPSED: 0 00:00:12.60 CPU: 0:00:05.82 BUFIO: 51 DIRIO: 178 FAULTS: 1480

In this example, one feature is found to differ. Compare it with the next example.

\$ IDIFFERENCE TJI1 TJI3 /TOLERANCE=(DEFAULT=0.001)<CR>

File 1 is LSL750\$DUA0:[LSL.IFF]TJI1.IFF;4

File 2 is LSL750\$DUA0:[LSL.IFF]TJI3.IFF;1

Comparing with lookahead 20, point tolerances: default 0.00100, factor 1.00

HI (HISTORY) entry missing in file 1

HI (HISTORY) entry missing in file 2

Comparing layer 1

Feature 1 line feature

=====

CB data - differences for file 1

Point	X	Y
2:	2.8696361	20.28903
3:	4.7384071	28.668381
4:	4.315804	32.933281

...equal

CB data - differences for file 2

Point	X	Y
2:	2.8729999	20.188999
3:	4.7371998	28.669201
4:	4.316	32.9333

...equal

=====

CB data - differences for file 1

Point	X	Y
7:	5.1894598	39.32761
8:	7.4283938	43.95649

...equal

CB data - differences for file 2

Point	X	Y
7:	5.1894999	39.330101
8:	7.4278998	43.9566

...equal

Summary of differences

1 layer was found

Number of layers compared	1
Missing layers in file 1	0
Missing layers in file 2	0

9 features required comparison:

Identical	8
Different	1
Unmatched	0

How features differed by entry type:

CB entries - geometry	1
-----------------------	---

Missing features in file 1	0
----------------------------	---

Missing features in file 2	0
Duplicated FSNs in file 1	0
Duplicated FSNs in file 2	0

ELAPSED: 0 00:00:13.93 CPU: 0:00:05.95 BUFIO: 64 DIRIO: 175 FAULTS: 1513

In this example the user has specified the DEFAULT tolerance, to be used for comparing real attributes that do not have an explicit interval in their ACD definition. In the previous example the same two files were compared using the default value of 0.03412, or 1/1000th of the maximum sheet extent. In this example, several more points are discovered to be 'different', due to the finer tolerance.

MESSAGES (GENERAL)

IDIFFERENCE message output differs from the other IMP utilities in two ways:

1. most of its messages are (by definition) informational, since it will normally succeed in comparing two files
2. most of its messages are output as simple lines of text, rather than in the %'utility'-'severity'-'message' format

Thus in the following sections, only those messages that are produced in unusual circumstances, or that might be puzzling, are listed. Also, all messages are presented as they would appear - that is, the full message text is shown, including the %'utility'... message header when appropriate.

MESSAGES (INFORMATIONAL)

Most IDIFFERENCE messages may be regarded as informational. Apart from messages arising directly from the successful comparison of two files, the following may be produced:

FSN 'integer' duplicated in file 'integer' - first occurrence ignored

Explanation: Whilst scanning the current layer for features to be compared, IDIFFERENCE has found a feature with the same FSN as a previous feature in that layer. As the message indicates, the program will use this latest feature for comparison.

User action: None.

****COMPMD**** Type 'integer' MD in file 'integer' has invalid length

Explanation: The program has attempted to read in an MD (Map Descriptor) entry, but the entry is of the wrong length for the indicated version (1 or 2) of MD. The entry will not be compared.

User action: Investigate why the entry is the wrong length using IPATCH - it may be that the MD type field is set incorrectly, since type 1 and type 2 MD entries have different sizes.

****GETCOD**** Unexpected end of file while searching for FS
feature 'integer' in file 'integer'

Explanation: IDIFFERENCE was looking for the FS entry of a feature that only occurs in one of the IFF files, so that it could determine its graphical type. Unfortunately, the feature did not contain an FS entry, and IDIFFERENCE found the end-of-file instead. This indicates that the file is incorrectly terminated in some way. IPATCH will not output a correct message for this feature.

User action: Use IPATCH to examine the end of the file, and use IMEND or LITES2 to correct the file.

****GETEF**** Unexpected end of feature 'integer' in file 'integer'

Explanation: IDIFFERENCE was scanning a feature before comparing it, and found end-of-file before it had encountered an EF entry. IDIFFERENCE will abandon comparison of this feature.

User action: Use IPATCH to examine the end of the file, and use IMEND or LITES2 to correct the file.

****GETEF**** Unrecognised code 'entry' in feature 'integer', file 'integer'

Explanation: IDIFFERENCE found an IFF entry that it did not expect whilst scanning a feature before comparing it. The entry will be ignored.

User action: Dependent upon the entry reported. If you believe that the entry is correct, and that IDIFFERENCE should compare it, then please consult Laser-Scan. Otherwise, determine why the entry occurred within that

feature.

****GETEF**** more than 'integer' AC entries in feature 'integer' in file 'integer'

Explanation: When scanning a feature before comparing it, IDIFFERENCE remembers all the feature's AC entries in an internal buffer. This message indicates that that buffer is now full, and further ACs in that feature will be ignored.

User action: Consult Laser-Scan about increasing the size of the internal buffer.

****READCB**** Too many rows (points) in 'entry' at 'address' in file 'integer' (found 'integer', max is 200)

Explanation: A CB (or ST or ZS) entry with more than 200 points has been encountered. IDIFFERENCE will attempt to use the first 200 points for the purposes of comparison - this is the standard maximum number of points in a single coordinate block.

User action: Use ITOTEXT and IFROMTEXT to correct this particular file, and correct the program that produced the coordinate string.

****READIN**** Non-empty feature with FSN 0 in file 'integer'
- feature ignored

Explanation: IDIFFERENCE has encountered a feature with FSN 0, which does not immediately end with an EF. Historically, features with FSN 0 are 'empty' features, used as place-holders for TC entries. The feature is ignored.

User action: If the feature is meaningful, correct its FSN using IPATCH, IRENUMBER, etc. If the feature is intended as an empty feature 0, then use ITOTEXT and IFROMTEXT to remove its contents.

****READIN**** Unexpected end-of-file in file 'integer'
scanning layer 'integer' which starts at 'address'
attempting to continue

Explanation: This error should never occur - it probably reflects an internal consistency error in the program.

User action: Please report this problem to Laser-Scan with as many details as possible.

****SCANSUB**** Unexpected end of feature 'integer' in file 'integer'

Explanation: IDIFFERENCE was scanning a text component before comparing it, and found end-of-file before it had encountered an EF entry. IDIFFERENCE will attempt to compare as much of the component as it has read.

User action: Use IPATCH to examine the end of the file, and use IMEND or LITES2 to correct the file.

****SCANSUB**** Unknown code 'entry' in feature 'integer', file 'integer'

Explanation: IDIFFERENCE found an IFF entry that it did not expect whilst scanning a text component before comparing it. The entry will be ignored.

User action: Dependent upon the entry reported. If you believe that the entry is correct, and that IDIFFERENCE should compare it, then please consult Laser-Scan. Otherwise, determine why the entry occurred within that text-component.

****SCANSUB**** No CB entry in component

Explanation: IDIFFERENCE has not found any coordinate entries in a text component. This means that the component is erroneous. The program will compare those entries it did find.

User action: Investigate why the text component does not contain a CB (or ST or ZS) entry. If the file was produced by a user written program, correct the program; if it was produced by a Laser-Scan program, please consult us with as many details as possible.

****SCANSUB**** Unexpected end-of-file in feature 'integer' in file 'integer'

Explanation: IDIFFERENCE was scanning a text component to locate all of its entries, and found end-of-file before finding an EF. The program will attempt to continue comparison, as if it had found an EF.

User action: Use IPATCH to examine the end of the file, and use IMEND or LITES2 to correct the file.

****SCANTS**** Unexpected end of file scanning feature 'integer' in file 'integer'

Explanation: IDIFFERENCE was scanning a composite text to locate all of the TS entries, and found end-of-file before finding an EF. The program will attempt to continue comparison, as if it had found an EF.

User action: Use IPATCH to examine the end of the file, and use IMEND or LITES2 to correct the file.

****SCANTS**** more than 'integer' TS entries in feature 'integer' in file 'integer'

Explanation: When comparing composite text features, IDIFFERENCE remembers the start of each text component in an internal buffer. This error indicates that the program has found more text components than it can remember. It will compare the components that it has room for.

User action: Consult Laser-Scan about increasing the size of the internal buffer.

****WRPLOT**** Layer mismatch

Explanation: This message indicates that an error has occurred whilst outputting data to one of the IFF plot files. It should never occur, and indicates that the program has become confused as to which layer it is outputting - an internal array has not been sorted correctly. IDIFFERENCE will continue outputting data.

User action: Please report this problem to Laser-Scan, with as much information as possible.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The normal cause is an error related to command line processing and file manipulation.

%IDIFF-E-2NDNAME, cannot parse file 'file-spec' to deduce a second file

Explanation: If the user has only specified the master file-spec, then IDIFFERENCE will try to compare it with the previous version of the file. This message indicates that something went wrong with that process, and will be accompanied by appropriate LSLIB error messages.

User action: Dependent upon the associated messages.

%IDIFF-E-LOOKAHEAD, /LOOKAHEAD value 'integer' should be in range 1 to 'integer'

Explanation: An invalid number was specified with the /LOOKAHEAD qualifier.

User action: Specify a value in the range shown by the message.

%IDIFF-E-NEGTOL, /TOLERANCE value 'real' should not be negative

Explanation: A negative tolerance value has been specified with the /TOLERANCE qualifier. This does not make sense.

User action: Specify a non-negative value.

%IDIFF-E-NOPREV, cannot produce the previous version of file 'file-spec'

Explanation: This message is essentially the same as the %IDIFF-E-2NDNAME message above - the same comments apply.

User action: As for the 2NDNAME error.

%IDIFF-E-NORANGE, cannot compare IFF files without RA entries

Explanation: Since IDIFFERENCE may use the range entries to establish a tolerance for comparing real numbers, it requires that all files to be compared contain a range. The most likely cause of this error is an attempt to compare two Laseraid patch files.

User action: Do not use IDIFFERENCE on these files. If they are indeed patch files, use ITOTEXT and compare the resultant text files with DIFFERENCE.

%IDIFF-E-OUTOPEN, error opening text output file 'file-spec'

Explanation: The program could not open the /OUTPUT file. The LSLIB messages output with this error should clarify the problem.

User action: Dependent upon the LSLIB messages.

%IDIFF-E-PLOTCRE, cannot parse plot IFF file spec 'file-spec'

Explanation: To produce a file-spec for the /PLOT IFF files, IDIFFERENCE must parse the master file-spec. This message indicates that it failed to do so. Associated LSLLIB error messages should clarify the situation.

User action: Dependent upon the LSLLIB messages.

%IDIFF-E-PLOTSPEC, cannot parse file-spec 'file-spec' to produce plot IFF file-spec

Explanation: To produce a file-spec for the /PLOT IFF files, IDIFFERENCE adds four characters to the end of the file name. This message indicates that it could not do this. Associated LSLLIB messages should help clarify the situation, but it is probable that the original file name was too long to have four characters added.

User action: Dependent upon the LSLLIB messages.

%IDIFF-E-SCANCB, error scanning CB entries in feature 'integer' in file 'integer'

Explanation: Before comparing them, IDIFFERENCE scans the CB entries for each feature or text component to work out how many columns are present, what they are, etc. This message indicates that the scanning process has failed. It will be followed by another message indicating the nature of the problem.

User action: Dependent upon the following messages.

MESSAGES (FATAL)

These messages indicate that the program has failed in some catastrophic manner. IDIFFERENCE fatal error messages will always be followed by one of the following sequences of text:

*** FATAL EXIT ***

This is almost certainly due to a bug in the program.
Please report it to Laser-Scan with as many details as possible

or

*** FATAL EXIT ***

This is due to some limitation in the program.
If this causes serious problems, consult Laser-Scan about increasing the limit.

%IDIFF-F-GETNOV, cannot find next layer - DIFNOS is 'integer'

Explanation: An internal error has occurred in the mechanism which locates the next layer in the IFF file.

User action: Report the problem to Laser-Scan, with as much information as possible.

%IDIFF-F-MAXNOS, more than 'integer' NO entries, after 'address' in file 'integer'

Explanation: The specified input file contains more NO (start of layer) entries than will fit into the internal buffers. This does not necessarily mean that the file contains too many **layers**, but rather that it is too fragmented.

User action: Use IMERGE to consolidate the layers. If the problem continues, then either there is a bug in the program, or the IFF file has a very large number of layers. In either case, consult Laser-Scan.

'name' lost position in input IFF file 'integer' at address 'address'
abandoning comparison

'name' lost position in output IFF file 'file-spec' at address 'address'
abandoning comparison

'name' lost position in IFF file on lun 'integer' at address 'address'
abandoning comparison

Explanation: IDIFFERENCE was trying to move to a known position in the relevant IFF file, and for some reason arrived at the wrong place. This represents a bug in the program, specifically a problem in the IDIFFERENCE routine 'name'.

User action: Report the problem to Laser-Scan, with as much information as possible.

****GETEF**** Entry read as 'entry' - should be seen as a CB

Explanation: All ST and ZS entries should be translated into the corresponding CB entry by the IFFLIB library before reaching IDIFFERENCE itself. This message indicates that the translation has failed.

User action: Report the problem to Laser-Scan, with as much information as possible.

****READIN**** found FSN 'integer' in file 'integer' - bigger than 65535 !

Explanation: It should not be possible for the program to encounter an FSN of this size - something has gone wrong internally.

User action: Report the problem to Laser-Scan, with as much information as possible.

****READIN**** found FSN 'integer' in file 'integer' - less than zero !

Explanation: It should not be possible for the program to encounter a negative FSN - something has gone wrong internally.

User action: Report the problem to Laser-Scan, with as much information as possible.

****SCANSUB**** Entry read as 'entry' - should be seen as a CB

Explanation: All ST and ZS entries should be translated into the corresponding CB entry by the IFFLIB library before reaching IDIFFERENCE itself. This message indicates that the translation has failed.

User action: Report the problem to Laser-Scan, with as much information as possible.

MESSAGES (OTHER)

In addition to the above messages which are generated by IDIFFERENCE itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 4

MODULE IFILTER

MODULE **IFILTER**

REPLACES A variety of Laser-Scan smoothing and filtering programs.

FUNCTION

IFILTER is designed to smooth or filter, smooth and then filter, or filter and then smooth IFF coordinate strings using one of the following algorithms:

FILTERS:

- (i) Laser-Scan "BUNCH" least squares filter,
- (ii) Douglas-Peucker filter

SMOOTHING OPTIONS:

- (i) Linear interpolation,
- (ii) Akima cubic interpolation,
- (iii) McConalogue cubic interpolation
- (iv) Bezier interpolation

```
+-----+
|                                     |
|                               WARNING |
| IFILTER produces only (X,Y) coordinate data in the |
| output file. Non-planimetric per-point attributes |
| held in CB data columns are not retained.          |
|                                     |
+-----+
```

FORMAT

\$ IFILTER input-file-spec output-file-spec

Command qualifiers

/[NO]AKIMA=(tolerance-spec[,...])
/[NO]BEZIER=(tolerance-spec[,...])
/[NO]BUNCH=(tolerance-spec[,...])
/[NO]DP[=tolerance-spec]
/FC=integer[,...]
/FILTER_FIRST
/LAYER=integer[,...]
/[NO]LINEAR[=tolerance-spec]

Defaults

/NOAKIMA
/NOBEZIER
/NOBUNCH
/DP
All feature codes.
Smooth before filtering.
All layers.
/NOLINEAR

```
/[NO]LOG                                /NOLOG
/[NO]MCCONALOGUE=(tolerance-spec[,...]) /NOMCCONALOGUE
```

PROMPT

```
_IFF-file:          input-file-spec
_Filtered-IFF-file:  output-file-spec
```

PARAMETERS**input-file-spec**

- specifies the IFF file which is to be filtered. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

output-file-spec

- specifies the IFF file which is to be created to contain the smoothed or filtered data. Any part of the file specification which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output IFF file. If a file with the specified name already exists a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

```
/AKIMA=(SEPARATION:real,CHORD:real,DEVIATION:real)
/NOAKIMA
```

- causes Akima cubic interpolation (smoothing) to be applied to the specified coordinate strings in the input IFF file. The IFF strings may either be taken directly from the input file, or they may first be passed through one of the filter options. If neither filter option is selected, or if /FILTER_FIRST is in effect, then the data resulting from the Akima interpolation are written directly to the output file.

Not all of the keywords are necessary, but at least one must be given. SEPARATION (the minimum distance between output data points) and CHORD (the arc to chord distance between output data points) must both be specified in the same units as the data in the IFF file, and DEVIATION (the angular deviation between output data points) must be specified in radians. Those not specified default to 0.0.

For further details on Akima interpolation and an explanation of the keywords, see the description section below. Some diagrams (Figs 1,2,3) illustrating the effect of Akima interpolation are included in the Examples section.

```
/BEZIER=(CHORD:real,HUG:real,[NO]AVERAGE)  
/NOBEZIER
```

- causes Bezier interpolation (smoothing) to be applied to the specified coordinate strings in the input IFF file. The IFF strings may either be taken directly from the input file, or they may first be passed through one of the filter options. If neither filter option is selected, or if /FILTER_FIRST is in effect, then the data resulting from the BEZIER interpolation are written directly to the output file.

CHORD is the arc to chord distance between output data points, and HUG is the corner hug tolerance. If CHORD is not specified a default value of 0.05 is used; if HUG is not specified a default hug tolerance of 1.0 is used. The minimum HUG tolerance is 0 and the maximum 1.

AVERAGE determines whether a series of new points that define a medial line through the original data points are generated before interpolation. This option is used to remove jagged edges from the input data, and should be applied to data such as that produced by the TVES module VECTORISE with its characteristic 'staircase' appearance. The generation of medial data points is by default not performed (ie. NOAVERAGE).

For further details on BEZIER interpolation see the description section below. Some diagrams (Figs 4,5) illustrating the effect of BEZIER interpolation are included in the Examples section.

```
/BUNCH=(MIN_SEP:real,MAX_SEP:real,LATERAL:real,RESOLUTION:real])  
/NOBUNCH
```

- causes IFILTER to apply the Laser-Scan BUNCH (least squares) filter to the specified coordinate strings in the input IFF file. The IFF strings may either be taken directly from the input file or they may have been passed through one of the interpolation options. The data resulting from application of the BUNCH filter are written directly to the output file, except when /FILTER_FIRST is in effect and interpolation follows.

Values for the keywords must be specified in the same units as the data in the IFF file. MIN_SEP (the minimum point separation after filtering) and LATERAL (the lateral threshold distance) are compulsory, while MAX_SEP (the maximum point separation after filtering) defaults to 0.0 which is interpreted as 'infinity'. Note, since BUNCH is a filter, no new points will be generated if the distance between input points is greater than MAX_SEP. RESOLUTION is the 'resolution' of the data coordinate system. Two or more points are considered to be coincident, if their x component and y component separations are both less than the 'resolution' prior to filtering. This parameter defaults to the map range divided by 1600 (which is equivalent to 0.25mm on the sheet for OS 1/1250 and 1/2500 maps).

For a description of the BUNCH filter and further explanation of the keywords, see the description section below. Some diagrams (Figs 6,7,8) illustrating the effect of BUNCH filter are included in the Examples section.

/DP[=LATERAL:real]
/NODP

- by default IFILTER will apply the Douglas-Peucker filter either directly to strings from the input file or to those strings after they have been interpolated by one of the interpolation options. The data resulting from application of the Douglas-Peucker filter are written directly to the output file, except when /FILTER_FIRST is in effect and interpolation follows.

/NODP causes IFILTER not to apply the Douglas-Peucker filter. The Douglas-Peucker filter is also not applied if the BUNCH filter is selected. It is not possible to use /NODP if no other smoothing or filtering qualifiers have been specified.

LATERAL is the shortest distance by which the new line will deviate laterally from the original line, and must be specified in the same units as the data in the IFF file. If this keyword is not given, it defaults to the map range divided by 1600 (this is equivalent to 0.25mm on the sheet for OS 1/1250 and 1/2500 maps). The lateral tolerance (whether specified or calculated by default) must be greater than 0.005, otherwise that value will be used instead.

For a description of the Douglas-Peucker filter see the description section below. A diagram (Fig 9) illustrating the effect of Douglas-Peucker filter is included in the Examples section.

/FC=integer[,...]

- causes IFILTER to process only those IFF features which have the specified feature code(s). The /FC qualifier will accept single, multiple or ranges of feature code arguments. Ranges may be specified with starting and ending feature codes that are separated by a colon, for example /FC=56:58. When specifying more than one range of feature codes, separate each range with a comma, for example /FC=(1:5,56:89). The maximum number of feature codes which may be specified is 1024.

/FILTER_FIRST

- this causes the relevant filtering algorithm to be applied to coordinate strings before interpolation. The default action is for interpolation to be applied first. This qualifier is only valid if both filtering and smoothing are to be carried out.

/LAYER=integer[,...]

- causes IFILTER to process only those IFF features which lie within the specified layer(s). /LAYER will accept single, multiple or ranges of layer arguments. Ranges may be specified with starting and ending layer numbers that are separated by a colon, for example /LAYER=56:58. When specifying more than one range of layer numbers, separate each range with a comma, for example /LAYER=(1:5,56:89). The maximum number of layers which may be specified is 1024.

/LINEAR[=SEPARATION:real]
/NOLINEAR

- causes IFILTER to apply linear interpolation to the specified coordinate strings in the input IFF file. The IFF strings may either be taken directly from the input file, or they may first be passed through one of the filter options. If neither filter option is selected, or if /FILTER_FIRST is in effect, then the data resulting from the linear interpolation are written directly to the output file.

SEPARATION is the minimum distance required between output data points and must be specified in the measurement units of the data coordinate system. If this keyword is not given, one point will be inserted in the middle of every line segment regardless of segment length.

Note that linear interpolation will not be carried out on two-point coordinate strings, whether taken directly from the input file or generated as a result of filtering.

Note also that if /FILTER_FIRST has not been specified it is important to give the /NODP qualifier. Otherwise the additional points generated by the /LINEAR option will immediately be filtered away again.

/LOG
/NOLOG

- causes supplementary messages to be sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened, and a reassuring message is output each time a new IFF layer is encountered in the input file.

/MCCONALOGUE=(SEPARATION:real,CHORD:real,DEVIATION:real)
/NOMCCONALOGUE

- causes McConalogue cubic interpolation to be applied to the specified coordinate strings in the input IFF file. The IFF strings may either be taken directly from the input file, or they may first be passed through one of the filter options. If neither filter option is selected, or if /FILTER_FIRST is in effect, then the data resulting from the McConalogue interpolation are written directly to the output file.

Not all of the keywords are necessary, but at least one must be given. SEPARATION (the distance between output data points) and CHORD (the arc to chord distance between output data points) must both be specified in the same units as the data in the IFF file, and DEVIATION (the angular deviation) must be specified in radians. Those not specified default to 0.0.

For further details on McConalogue interpolation and an explanation of the keywords, see the description section below. Some diagrams (Figs 10,11,12) illustrating the effect of McConalogue interpolation are included in the Examples section.

RESTRICTIONS

- o /AKIMA, /BEZIER, /MCCONALOGUE and /LINEAR are mutually exclusive
- o /BUNCH and /DP are mutually exclusive
- o /NODP requires one of /AKIMA, /BEZIER, /MCCONALOGUE, /LINEAR or /BUNCH
- o /FILTER_FIRST requires one of /AKIMA, /BEZIER, /MCCONALOGUE or /LINEAR, as well as either /DP or /BUNCH

DESCRIPTION**General**

IFILTER is designed to smooth or filter, smooth and then filter, or filter and then smooth two-dimensional coordinate strings held as IFF ST entries. The program offers two filter options (Douglas-Peucker and BUNCH least squares), and four interpolation options (Akima cubic, Bezier polynomial, McConalogue cubic and linear). IFILTER may be used to smooth or filter data selectively on the basis of layer and/or feature code. All features are copied to the output file, but only those selected will be smoothed or filtered.

IFILTER copies through to the output file the following, without reference to any smoothing or filtering options selected :-

- all input features having 3 points or less.
- all features which lie in IFF layer 0

```
+-----+
|                WARNING                |
| IFILTER produces only (X,Y) coordinate data in the |
| output file. Non-planimetric per-point attributes |
| held in CB data columns is not retained.          |
+-----+
```

BUNCH Filter

The BUNCH filter uses tolerances related to 'best fit' lines through the incoming coordinate strings. The filter performs successive least squares fits through the existing data points until a point lies more than the **lateral** threshold distance (specified using the LATERAL keyword) from the least squares line. This causes a provisional master point to be generated. A new fit is then conducted forwards from the last master point, until the lateral tolerance is again exceeded. The last provisional point is tidied up into a true master point and the intervening points are rejected. The process is repeated until the end of the coordinate string is reached. If the

lateral tolerance is large, it will rarely be exceeded and many points will be thrown away.

The number of points which are kept as master points or are thrown away is additionally controlled by the minimum and maximum separation (MIN_SEP and MAX_SEP) keywords, where MIN_SEP is mandatory but MAX_SEP may be defaulted. The meaning of these keywords is as follows:

- o the minimum separation is the shortest distance allowed between successive master points **along** the line. If this is set to a large value, more points will be thrown away giving increasingly angular linework.
- o the maximum separation is the distance travelled **along** the line before IFILTER must force out a master point. A large value will result in very sparse points along straight and nearly straight lines. A maximum separation of 0.0 (the default) is equivalent to one of infinity, and means that no points will be forced out purely on distance criteria (i.e. a straight line will have two points).

The maximum separation must be greater than or equal to the minimum separation which must be greater than or equal to the lateral threshold distance.

The optional RESOLUTION keyword specifies the resolution of the data coordinate system. Two or more points are considered to be coincident, if their x component and y component separations are both less than the 'resolution' prior to filtering. This being the case, then only one of them will contribute to the least squares line fitting process (the rest are discarded).

Strictly, smoothing is not to be recommended after the BUNCH filter has been applied. The reason for this is that it can result in biased data. The BUNCH filter uses a least squares algorithm to produce coordinates which, when joined by straight lines, approximate the original data in a least squares sense. This typically means, for example, that on curved lines all resulting points tend slightly towards the **outside** of the original curve. Application of one of the curve interpolation algorithms will then result in a new curve biased towards the outside of the original one.

Douglas-Peucker Filter

The Douglas-Peucker filter works by joining the first and last points of the line being filtered with a straight line. The longest perpendicular to this straight line which cuts the line being filtered is then found. The two parts of the original line are then treated separately and the same process applied to each of them. This is repeated for successively smaller sections of the original line until any further perpendiculars which might be derived would be shorter than the lateral tolerance (specified by the LATERAL keyword). The new line consists of the succession of points where the perpendiculars cut the original line.

Curve Interpolation

Three curve interpolation methods are available; these are Akima, Bezier and McConalogue:

- o Akima is a cubic arc blending method which preserves linearity if possible.
- o Bezier is a polynomial algorithm.
- o McConalogue is a quadratic arc blending method.

Akima's method tends to give a much 'tighter' curve than McConalogue's, and unwanted inflection points are rare. On the other hand, that tightness means that the second derivative discontinuities tend to be greater and may be visible.

The values specified for the three qualifier keywords control the density of interpolated points. The approximate separation of points (d) is given by:

$$d = a + 2*\text{SQRT}(2br) + cr \quad ('r' \text{ is radius of curvature})$$

which means (if other coefficients were zero) that

- **a** gives a constant separation of '**a**' IFF units (SEPARATION keyword)
- **b** gives a constant 'arc to chord' distance of '**b**' IFF units (CHORD keyword)
- **c** gives a constant angular deviation of '**c**' radians (DEVIATION keyword)

Typically only one of the keywords need be given (SEPARATION is usually sufficient), however the flexibility is provided to enable precise results to be obtained by experienced users.

LITES2 users should note that these three coefficients correspond to those specified in the TOLERANCE CURDRAW and TOLERANCE CURGEN commands. The difference in LITES2 is that the coefficients are given in mm, not IFF units.

Bezier is a polynomial interpolation algorithm which is widely used in interactive graphics to obtain approximate solutions to curve fitting problems. The Bezier algorithm varies from the Akima and McConalogue algorithms, in that the resulting curve does not pass exactly through the original data points, since the curve is generated from a set of guide points derived from the original data. This fact should be appreciated when applying the algorithm to cartographic data.

The Bezier CHORD and HUG tolerance keywords control the shape of the interpolated curve and the number of points generated. The HUG tolerance value should be in the range 0 to 1. When HUG is 0.0, smoothing will be minimal with the output line being approximately the same as the input. When HUG is 1.0 smoothing will be greater. The

CHORD tolerance value is dominant in determining the final shape of the curve.

The AVERAGE keyword is applicable only to the BEZIER qualifier, and determines whether a set of points defining a medial line through the original data are first generated. If the option is specified, it is the medial points that are passed to the Bezier algorithm, and from which guide points are generated. The keyword should be specified if interpolation is being applied to data generated from the TVES module VECTORISE, or the original data shows excessive jaggedness.

IFF Files With Junction Structure

If the input IFF file contains junction structure entries, IFILTER will generate a filtered file with a junction structure. IFILTER assumes that a node is recorded at the start and end of each feature. If features exist in the input file that contain more than 2 node (JP) entries, only the JP entries relating to the start and end nodes are preserved in the output file.

If the AVERAGE keyword is used with the BEZIER qualifier, then IFILTER ensures the node points in the input junction structure file, are preserved in the output file. The same restriction is not applied if the input file contains no junction structure.

EXAMPLES

```
$ IFILTER CCB2 NEW_CCB2<CR>
```

```
ELAPSED:    0 00:00:07.91  CPU: 0:00:03.20  BUFIO: 5  DIRIO: 54  FAULTS: 152
$
```

This example shows a successful run of IFILTER in default mode, i.e. applying only the Douglas-Peucker filter to coordinate strings in LSL\$IF:CCB2.IFF. The filtered data have been written to LSL\$IF:NEW_CCB2.IFF.

```
$ IFILTER/MCCONALOGUE=(SEPARATION:0.3,CHORD:0.2,DEVIATION:0.15)/NODP -<CR>
_$      CCB2 NEW_CCB2<CR>
```

```
ELAPSED:    0 00:00:13.21  CPU: 0:00:03.37  BUFIO: 4  DIRIO: 53  FAULTS: 109
$
```

In this example IFILTER has successfully applied the McConalogue curve interpolation algorithm to coordinate strings in LSL\$IF:CCB2.IFF. Filtering has been suppressed and the interpolated data output to LSL\$IF:NEW_CCB2.IFF.

```
$ IFILTER/AKI=(SEP:0.3,CHO:0.2)/BUN=(MIN:0.3,LAT:0.2)/LAY=(1,2)/LOG -<CR>
_$      OEEPECON CONTOURS1<CR>
```

```
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]OEEPECON.IFF;12 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]CONTOURS1.IFF;1 opened for write
%IFILT-I-LAYERPROC, layer 1 being processed
%IFILT-I-FEATCOPY, FSN 45 (3) copied without modification
%IFILT-I-FEATCOPY, FSN 71 (12) copied without modification
%IFILT-I-FEATCOPY, FSN 135 (17) copied without modification
%IFILT-I-LAYERPROC, layer 2 being processed
%IFILT-I-FEATCOPY, FSN 158 (25) copied without modification
%IFILT-I-FEATCOPY, FSN 159 (26) copied without modification
%IFILT-I-FEATCOPY, FSN 160 (27) copied without modification
%IFILT-I-FEATCOPY, FSN 161 (28) copied without modification
%IFILT-I-FEATCOPY, FSN 162 (29) copied without modification
%IFILT-I-FEATCOPY, FSN 163 (30) copied without modification
%IFILT-I-FEATCOPY, FSN 164 (31) copied without modification
%IFILT-I-LAYERCOPY, layer 32 being copied without modification
%IFILT-I-FEATCOPY, FSN 9996 (51) copied without modification
%IFILT-I-FEATCOPY, FSN 9997 (52) copied without modification
%IFILT-I-FEATCOPY, FSN 9998 (53) copied without modification
%IFILT-I-FEATCOPY, FSN 9999 (54) copied without modification
ELAPSED:    0 00:02:29.06  CPU: 0:00:50.56  BUFIO: 28  DIRIO: 375  FAULTS: 117
$
```

This example illustrates a successful run of IFILTER applying the Akima curve interpolation algorithm followed by the least squares BUNCH filter. The point separation for Akima has been calculated from the given constant separation and arc to chord distance, the angular deviation defaulting to 0.0. For the BUNCH filter, the minimum point separation and lateral tolerance have been specified, with no maximum separation. The default is assumed for the resolution as this is an OS

map. Coordinate strings in layers 1 and 2 of LSL\$IF:OEEPECON.IFF have been selected for interpolation and filtering, with those in the remaining layer (32) simply being copied over to the output file LSL\$IF:CONTOURS1.IFF. Additional informational messages have been requested; these include messages indicating whether each layer is being processed or copied unchanged, and listing any line features to which interpolation and filtering have not been applied. Note that the qualifiers and keywords in this example have been abbreviated.

\$ IFILTER/LIN=SEP:0.4/DP=LAT:0.8/FILTER OEEPECON CONTOURS2<CR>

ELAPSED: 0 00:02:36.92 CPU: 0:01:01.70 BUFIO: 26 DIRIO: 1641 FAULTS: 153
\$

In this example, the Douglas-Peucker filter has been applied to coordinate strings in LSL\$IF:OEEPECON.IFF, followed by linear interpolation. The filtered and interpolated data have been output to LSL\$IF:CONTOURS2.IFF.

\$ IFILTER/BUN=(MIN:0.2,MAX:0.5,LAT:0.3) OEEPECON FILT_CON<CR>

%IFILT-E-INVMINLAT, /BUNCH - MIN_SEP less than LATERAL

ELAPSED: 0 00:00:00.58 CPU: 0:00:00.22 BUFIO: 1 DIRIO: 0 FAULTS: 89
\$

This example shows the result of specifying inconsistent keywords with the BUNCH filter qualifier. The program has issued an error message and terminated without creating an output file or performing any processing.

\$ IFILTER/FILTER_FIRST CCB2 MY_CCB2<CR>

%CLI-W-CONFLICT, illegal combination of command elements - check documentation
\FILTER_FIRST\
ELAPSED: 0 00:00:00.39 CPU: 0:00:00.15 BUFIO: 2 DIRIO: 0 FAULTS: 95
\$

This example illustrates an error detected by the DCL command line interpreter (CLI). In this case the /FILTER_FIRST qualifier conflicts with the default operation of the program, which is that filtering only (with the Douglas-Peucker filter) should take place. The 'RESTRICTIONS' section states that /FILTER_FIRST requires one of /AKIMA, /MCCONALOGUE or /LINEAR, as well as either /DP or /BUNCH.

On the following pages are diagrams illustrating the various IFILTER operations. The top left hand picture, in all cases, is the original input data to IFILTER. Moving from left to right, top to bottom are pictures showing the effect of varying just one keyword. Note this is not the case in Fig 7 where it is necessary to vary the MIN_SEP keyword as well as LATERAL keyword.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). Most will only appear if /LOG has been specified on the command line, unless they are used to provide supplementary information after an error has occurred.

BADFEAT, error occurred processing feature 'integer' ('integer')

Explanation: An error has caused processing of the IFF file to be abandoned. A previous error message will indicate the nature of the problem. This message gives the FSN of the feature that was being processed when the error occurred.

User action: As for the accompanying error message.

CHKARGS, Check validity of all qualifier argument values

Explanation: An arithmetic error has caused processing of the IFF file to be abandoned. A previous error message will indicate the nature of the problem, for example floating division by zero. The most likely cause for an arithmetic error is the specification of command qualifier argument values that are inappropriate to the data being processed.

User action: Guidance on the effects of qualifier argument values for all the interpolation and filtering options is provided in the example daigrams of the IFILTER Description section of this manual.

FEATCOPY, FSN 'integer' ('integer') copied without modification

Explanation: The given feature is being copied to the output file without smoothing or filtering. This may be because its feature code has been excluded, or because the feature contains three data points or less. Note, however, that any superimposed points will still be removed from the coordinate strings.

User action: If in doubt, check why the feature was ignored.

LAYERCOPY, layer 'integer' being copied without modification

Explanation: The given layer is being copied to the output file with none of the features within it undergoing smoothing or filtering. Either the layer has been excluded by the /LAYER qualifier, or layer zero has been encountered (this is never processed).

User action: None.

LAYERPROC, layer 'integer' being processed

Explanation: The program has reached the given layer in the input file, and will commence filtering and/or smoothing as appropriate.

User action: None.

MESSAGES (WARNING)

These messages indicate that IFILTER has been unable to guarantee that a specified feature has been correctly processed, or that a specified IFF entry is correct.

DELOUTFIL, output IFF file deleted

Explanation: This is normally a continuation message and indicates that the output file has been deleted.

User action: None.

EXTCOLIGN, Only X and Y coordinates processed in FSN 'integer' (integer)

Explanation: IFILTER has encountered a "non-simple" CB entry containing more data than the standard (X,Y) coordinates. These may be per-point attributes. IFILTER does not reproduce these extra CB columns in the output file, because the actual coordinates output are different from the input coordinates.

User action: Check that no useful information is being lost.

FEATINCOMP, incomplete FSN 'integer' ('integer') in input file not copied

Explanation: This message indicates that a feature found in the input file is incomplete, having either no FS entry or no coordinates, and will not appear in the output file.

User action: Examine the input file using IPATCH, and if necessary amend the feature using LITES2 or ITOTEXT/IFROMTEXT and rerun IFILTER.

ONEPTST, FSN 'integer' ('integer') output with one point ST

Explanation: The given line feature has only one point. This may be due to the input IFF file containing a one-point coordinate string before an invisible segment, or it may be that the specified filter tolerances are too harsh.

User action: Check the validity of any filtering qualifiers specified. Check that the feature in the input file is valid. If the problem persists, preserve the input file, make a note of the qualifiers used, and report the error to Laser-Scan. The output file can then be checked and edited using IPATCH and LITES2 or ITOTEXT/IFROMTEXT.

SUPERIMPPT, superimposed point ignored in FSN 'integer' ('integer')

Explanation: The input IFF file contains coincident points. Only one of these is retained.

User action: None.

OUTFILCLO, file closed without updating RA

Explanation: This is normally a continuation message and indicates that the range entry in the output file is likely to be incorrect.

User action: Depends on the main error which caused this action, but the range entry in the output file may need to be corrected. This can be achieved by using IMERGE.

RANOTFND, cannot find RA entry in output IFF file

Explanation: Having completed reading the input file, with the appropriate features smoothed and filtered as required, the range entry in the output file has to be updated. This entry cannot be found, indicating a probable programming error. The program recovers from this error, and finishes normally.

User action: Preserve the input file, make a note of the qualifiers used, and report the error to Laser-Scan. The range entry in the output file may need to be corrected. This can be achieved by using IMERGE.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

EOIINFPEAT, unexpected end of input IFF file within feature

Explanation: The input IFF file terminated within a feature, before an IFF EF entry was encountered.

User action: Use IMEND/TRUNCATE to correctly terminate the file. Re-run IFILTER on the repaired IFF file.

ERRBEZIER, arithmetic error during Bezier interpolation

Explanation: An error has occurred when applying the Bezier interpolation algorithm.

User action: Please submit a SPR to Laser Scan, and if possible supply the input IFF file

ERRCLOIN, cannot close input IFF file

Explanation: The input IFF file could not be closed. This probably indicates a serious problem, and the preceding IFF library message will give more details.

User action: Investigate the cause of the error according to the IFF library message.

ERRCLOOUT, cannot close output IFF file

Explanation: The output IFF file could not be closed. This probably indicates a serious problem, and the preceding IFF library message will give more details.

User action: Investigate the cause of the error according to the IFF library message.

ERRWRTOUT, cannot write to output IFF file

Explanation: An error occurred while attempting to write to the output IFF file. This probably indicates a serious problem, and the preceding IFF library message will give more details.

User action: Investigate the cause of the error according to the IFF library message.

ILEGFC, illegal feature code 'integer' with /FC

Explanation: The feature code specified as an argument to the /FC qualifier does not lie in the range 0 to 32767.

User action: Respecify the command line, taking care that feature codes fall within the allowed range.

ILEGLAY, illegal IFF layer number 'integer'

Explanation: The layer number specified as an argument to the /LAYER qualifier does not lie in the range 0 to 32767.

User action: Respecify the command line, taking care that layer numbers fall within the allowed range.

INVHUG, /BEZIER - Corner hug tolerance not in the range 0.0 to 1.0

Explanation: The corner hug tolerance supplied with the /BEZIER qualifier may only be in the range 0 to 1.

User action: Respecify the command line, ensuring that the value for HUG is in the range 0 to 1.

INVMAXMIN, /BUNCH - MAX_SEP less than MIN_SEP

Explanation: The /BUNCH qualifier was specified, and the value given for the MAX_SEP keyword was less than that given for the MIN_SEP keyword.

User action: Respecify the command line, ensuring that the value for MAX_SEP is greater than or equal to MIN_SEP

INVMINLAT, /BUNCH - MIN_SEP less than LATERAL

Explanation: The /BUNCH qualifier was specified, and the value given for the MIN_SEP keyword was less than that given for the LATERAL keyword.

User action: Respecify the command line, ensuring that the value for MIN_SEP is greater than or equal to LATERAL.

OUTBUFFULL, output buffer full

Explanation: The number of points in the line after filtering and/or smoothing is more than the allowed maximum of 15000.

User action: Respecify the command line, but choose the filtering and/or smoothing qualifier keywords such that fewer points are produced. If line features with large numbers of points are required, please submit a software modification request to Laser-Scan.

REQPOSVAL, require positive value for /'qualifier' 'keyword'

Explanation: The given qualifier keyword requires a value which is greater than zero.

User action: Respecify the command line, taking care that the relevant value is greater than zero.

TOOMANYFC, 'integer' feature codes with /FC, maximum is 'integer'

Explanation: More than the maximum allowed number of feature codes have been specified with the /FC qualifier.

User action: Reissue the IFILTER command and qualifiers, taking care that the number of feature codes specified with /FC does not exceed the maximum.

TOOMANYLAY, 'integer' layers with /LAYER, maximum is 'integer'

Explanation: More than the maximum allowed number of layers have been specified with the /LAYER qualifier.

User action: Reissue the IFILTER command and qualifiers, taking care that the number of layers specified with /LAYER does not exceed the maximum.

TOOMANYPTS, FSN 'integer' ('integer') has too many points

Explanation: The given line feature as read from the input file has too many points and cannot be processed. The maximum allowed is 15000.

User action: Split up the feature in the input file using LITES2 and then reissue the IFILTER command line. If line features with large numbers of points are required, please submit a software modification request to Laser-Scan.

UNEXPEOF, unexpected end of input IFF file

Explanation: The input IFF file terminated unexpectedly, before an IFF EJ entry was encountered.

User action: Use IMEND/TRUNCATE to correctly terminate the file. Re-run IFILTER on the repaired IFF file.

MESSAGE (FATAL)

This message indicates a fatal error in processing which has caused the program to terminate. No output IFF file will be produced.

ARITHMETIC, arithmetic fault detected - possibly caused by inappropriate qualifier argument values

Explanation: An arithmetic error has caused processing of the IFF file to be abandoned. A following error message will indicate the nature of the problem, for example floating division by zero.

User action: A following informational message will provide suggested user action.

MESSAGES (OTHER)

In addition to the above messages which are generated by IFILTER itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 5

MODULE IFIXAREA

MODULE IFIXAREA

REPLACES DAMP module IFIX

FUNCTION

IFIXAREA is designed to check an IFF file for errors relating to area features. Features which are found to be in error are fixed and all features are written to a new output file. The errors to be checked for are specified on the IFIXAREA command line, and may be any combination of the following:

- o Repeated adjacent vertices.
- o 'Almost repeated' vertices - i.e. adjacent points in a feature. which become equal when their coordinates are rounded to the nearest integer, or lie within a specified tolerance of each other.
- o Open features - i.e. having at least three points and unequal start and end points.
- o The direction of feature digitising (the usual orientation for an area feature being anticlockwise).

By default, all features in the input IFF file containing 2-dimensional coordinate strings are checked.

```
+-----+
|                WARNING                |
| IFIXAREA only produces coordinate (X,Y,Z) data in the |
| output file. Non coordinate attributes held in CB    |
| data columns are not retained.                    |
+-----+
```

FORMAT

\$ IFIXAREA input-file-spec output-file-spec

Command qualifiers

/[NO]AC
/ANTI_CLOCKWISE
/CLOCKWISE
/[NO]CLOSE[=real]
/[NO]DELETE
/DFAD
/FC=integer[,...]
/LAYER=integer[,...]
/[NO]LOG

Defaults

/NOAC
/ANTI_CLOCKWISE
/ANTI_CLOCKWISE
/NOCLOSE
/NODELETE
See text.
All feature codes.
All layers.
/NOLOG

/NEAR[=real]	See text.
/OUTPUT[=file-spec]	/OUTPUT=SYS\$OUTPUT
/[NO]PRINTER	/NOPRINTER
/[NO]REVERSE	/NOREVERSE

PROMPT

_IFF-file:	input-file-spec
_Fixed-IFF-file:	output-file-spec

PARAMETERS**input-file-spec**

- specifies the IFF file to be fixed. Any part of the file-spec which is not specified will be taken from the default specification 'LSL\$IF:IFF.IFF'.

fixed-file-spec

- specifies the IFF file which is to be created. Any part of the file-spec which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output IFF file. If a file with the specified name already exists a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

/AC
/NOAC

- forces selection of DFAD features by type 7 ACs rather than selection by feature code. In level 1 DFAD data, area features can be recognised by the specification "typ=2" in the text field of the type 7 (DFAD) AC. /AC is only valid in conjunction with the /DFAD qualifier, and is thus invalid with the /FC qualifier.

/ANTI_CLOCKWISE

- when used in conjunction with the /REVERSE qualifier, results in all closed features being checked for direction of digitising. Any features which are not digitised in an anticlockwise direction will be made so. If /REVERSE is specified without a direction qualifier, then /ANTI_CLOCKWISE is the default.

/CLOCKWISE

- when used in conjunction with the /REVERSE qualifier, results in all closed features being checked for direction of digitising. Any features which are not digitised in a clockwise direction will be

made so. If /REVERSE is specified without a direction qualifier, then /ANTI_CLOCKWISE is the default.

/CLOSE[=real]
/NOCLOSE

- requests that IFIXAREA should check for open features and if any are found, close them. A feature is considered to be open if it has three or more points, and the distance between the start and end points is less or equal to the specified close tolerance. If no tolerance is specified, then all open features are closed. The tolerance is specified in IFF units.

Features are closed by adding an extra point at the end of the feature with the same coordinates as the start of the feature. However, if /NEAR is specified, then features being closed whose endpoints are within the 'almost repeated' tolerance are closed by moving the end point onto the start point, rather than by adding an extra point.

/DELETE
/NODELETE

- requests the deletion of repeated adjacent vertices.

/DFAD

- specifies that the input file contains DFAD data. In level 2 DFAD data, area features can be distinguished as having feature code 102. Unless the /AC qualifier is used to enforce level 1 DFAD conventions (q.v.), corrections will be applied to all DFAD level 2 area features within the file or specified layers. /DFAD is invalid for use with /FC.

If /DFAD is specified, the functions specified by the /CLOSE, /REVERSE and /NEAR qualifiers are performed. There is no need to specify /CLOSE, /REVERSE and /NEAR explicitly.

/FC=integer[,...]

- process only features with the specified feature codes. Ranges of feature codes may be specified with starting and ending feature code numbers that are separated by a colon (/FC=230:890). When specifying more than one range of FCs, separate each range with a comma (/FC=(20:40,60:89)) This qualifier is invalid with /DFAD. A maximum of 1024 feature codes may be specified.

/LAYER=integer[,...]

- process only the specified layers. Ranges of layers may be specified with starting and ending layer numbers that are separated by a colon (/LAYER=56:58). When specifying more than one range of layer numbers, separate each range with a comma (/LAYER=(1:5,56:89)). A maximum of 1024 layer numbers may be specified.

/LOG
/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT and to an output file if one has been specified with the /OUTPUT or /PRINT qualifiers. Supplementary messages are generated when a file is successfully opened, and a reassuring message output each time a new IFF layer is encountered in the IFF file. A report is made of each feature number and the correction performed. With large files this output can be voluminous !

/NEAR[=real]

- requests that IFIXAREA should check for 'almost repeated' vertices, i.e. those that lie within the distance specified of each other. If such vertices are found, then the second of each pair is not output to the output IFF file. The tolerance used in the comparison may be specified as 'real', a value in IFF units. If no tolerance is specified, then if NINT(x1)-NINT(x2) and NINT(y1)-NINT(y2) are both 0, the adjacent vertices x1,y1 and x2,y2 are considered to be 'almost adjacent'. Note that if too large a tolerance is specified, then excessive numbers of vertices may be thrown away!

/OUTPUT [=file-spec]

- by default output will be to SYS\$OUTPUT. This option allows the user to redirect output to the specified file.

/PRINTER
/NOPRINTER

- queues the IFIXAREA output for printing under the name given by the /OUTPUT qualifier. If you specify /PRINTER without the /OUTPUT qualifier, the output is directed to a file named SYS\$DISK:FIXAREA.LIS which is queued automatically for printing and then deleted.

/REVERSE
/NOREVERSE

- requests the reversal of features which have digitising order at variance to the direction specified by the /CLOCKWISE or /ANTI_CLOCKWISE qualifiers. The default direction for /REVERSE is anticlockwise.

RESTRICTIONS

- o /ANTI_CLOCKWISE and /CLOCKWISE are mutually exclusive
- o /DFAD and /FC are mutually exclusive
- o /AC is only valid with /DFAD

- o at least one check/fix qualifier must be specified

DESCRIPTION

Checking takes place by default for all features in the file. Alternatively a restricted class of features may be selected for checking. The restriction may be made by specifying one or more layer numbers which are to be processed with the /LAYER qualifier, and/or by specifying one or more feature codes with the FC qualifier.

Features not checked are copied to the output IFF file, and errors detected in other features are corrected before writing the feature to the output IFF file. The order of error correction is first repeated or 'almost repeated' vertices, then feature closure and finally direction of digitising. In 'almost repeated' vertices, the second vertex is removed. All combinations of **check** options are possible although /NEAR makes the /DELETE qualifier redundant. Not all combinations of **selection** options are valid however (see restrictions section). IFIXAREA aborts if no qualifiers are specified as the file would merely be copied to the output IFF file.

At the end of the run a summary is given containing the total number of features in the file and total numbers of features closed and reversed if the respective qualifiers are specified. If /DELETE or /NEAR is specified, the total number of points removed from all features appears in the summary.

IFIXAREA cannot deal with features containing more than 100000 points.

EXAMPLES

```
$ IFIXAREA TEST1.IFF/LOG/CLOSE/NEAR/REVERSE/PRINT TEST1.NEW<CR>
%IFIX-I-DEFANTICLK, anti-clockwise rotation defaults
%LSLLIB-I-IFFOPENED, DUA3:[TOM.TEST]TEST1.IFF; opened for read
%LSLLIB-I-IFFOPENED, DUA3:[TOM.TEST]TEST1.NEW opened for write
%IFIX-I-LAYPROC, processing layer 1
%IFIX-I-FSNCLO, feature with FSN 337 (377) closed
%IFIX-I-FSNPTDEL, 1 point deleted in feature with FSN 338 (338)
%IFIX-I-NOCLPTS, feature with FSN 338 (338) not closed, only 2 points
%IFIX-I-FSNPTDEL, 1 point deleted in feature with FSN 335 (335)
%IFIX-I-FSNCLO, feature with FSN 335 (335) closed
%IFIX-I-FSNREVD, feature with FSN 335 (335) reversed
%IFIX-I-NOCLPTS, feature with FSN 326 (326) not closed, only 2 points
%IFIX-I-FSNCLO, feature with FSN 336 (336) closed
%IFIX-I-FSNCLO, feature with FSN 370 (370) closed
%IFIX-I-FSNCLO, feature with FSN 369 (369) closed
%IFIX-I-FSNREVD, feature with FSN 369 (369) reversed
%IFIX-I-FSNCLO, feature with FSN 1568 (1568) closed
%IFIX-I-FSNCLO, feature with FSN 1569 (1569) closed
%IFIX-I-FSNREVD, feature with FSN 1569 (1569) reversed
%IFIX-I-FSNCLO, feature with FSN 1570 (1570) closed
%IFIX-I-NOCLPTS, feature with FSN 1633 (1633) not closed, only 2 points
%IFIX-I-FSNCLO, feature with FSN 1681 (1681) closed
%IFIX-I-FSNREVD, feature with FSN 1681 (1681) reversed
%IFIX-I-FSNPTDEL, 1 point deleted in feature with FSN 1682 (1882)
%IFIX-I-FSNCLO, feature with FSN 1682 (1682) closed
%IFIX-I-FSNREVD, feature with FSN 1682 (1682) reversed
%IFIX-I-FSNCLO, feature with FSN 1687 (1687) closed
%IFIX-I-FSNREVD, feature with FSN 1687 (1687) reversed
%IFIX-I-FSNPTDEL, 1 point deleted in FSN 1684 (1684)
%IFIX-I-FSNCLO, feature with FSN 1684 (1684) closed

.....etc.....

%IFIX-I-LAYPROC, processing layer 2
%IFIX-I-FSNREVD, feature with FSN 2030 (2030) reversed
%IFIX-I-FSNPTDEL, 1 point deleted in feature with FSN 1320 (1320)
%IFIX-I-LAYPROC, processing layer 111
%IFIX-I-FSNPTSDEL, 7 points deleted in feature with FSN 1938 (1938)
%IFIX-I-FSNREVD, feature with FSN 1938 (1938) reversed
%IFIX-I-FSNREVD, feature with FSN 2044 (2044) reversed
%IFIX-I-FSNPTDEL, 1 point deleted in feature with FSN 1706 (1706)
%IFIX-I-FSNPTSDEL, 5 points deleted in feature with FSN 154 (154)
%IFIX-I-FSNCLO, feature with FSN 154 (154) closed
%IFIX-I-FSNREVD, feature with FSN 154 (154) reversed
%IFIX-I-FSNPTDEL, 1 point deleted in feature with FSN 1616 (1616)
%IFIX-I-FSNPTSDEL, 8 points deleted in feature with FSN 1776 (1776)
%IFIX-I-FSNPTSDEL, 2 points deleted in feature with FSN 1593 (1593)
%IFIX-I-FSNREVD, FSN 1593 reversed

SUMMARY:
857 features processed
13 features closed
```

29 features reversed

117 points deleted

ELAPSED: 00:00:08.43 CPU: 0:00:04.63 BUFIO: 10 DIRIO: 120 FAULTS: 133
\$

This example demonstrates the use of the /LOG, /CLOSE, /NEAR, /REVERSE and /PRINT qualifiers. The user specified the input file to be LSL\$IF:TEST1.IFF, the output file to be LSL\$IF:TEST1.NEW, and gave the qualifiers to check for closure of features, 'almost repeated' vertices, and direction of digitising. The checks are carried out in the order specified in the documentation and not in the order that they are specified in the command line, so the first check carried out is for 'almost repeated' vertices. The input file is not changed. Features that are not in error are copied to the output IFF file and features in error are 'fixed' and then copied to the output file.

The user did not specify a distance for the /NEAR qualifier so the default is taken. The /CLOSE qualifier checks that all features with three or more points have equal start and end points. In this case the default operates. The /REVERSE qualifier was specified but no direction qualifier specified so again the default ANTI_CLOCKWISE operates, and a message is output to this effect. The /LOG qualifier has the effect of sending supplementary messages to SYS\$OUTPUT. As the example shows messages are output when files are successfully opened, when different layers are encountered, when any changes are made to any of the features, and why any features are not closed. When the end of the file is reached a summary of all the changes made is output with timing information. As the /PRINT qualifier was specified a summary of the program execution is also sent to the printer. This includes confirmation of the input and output IFF files, information about the user, the command line specified and messages about changes made to each feature.

\$ IFIXAREA

_Input-file: **CHELSEA/NEAR/CLOSE/LAY=111/FC=30/REVERSE/CLOCKWISE/OUTPUT<CR>**

_Fixed-IFF-file: **CHELSEA.NEW<CR>**

===== IFIXAREA =====

Input file: LSL\$IF:CHELSEA.IFF;
Output file: LSL\$IF:CHELSEA.NEW;0

IFIXAREA invoked by: TOM, Process: Tom Timms, on: 28-AUG-86 09:29:25

Command line:

IFIXAREA CHELSEA/NEAR/CLOSE/LAY=111/FC=30/REVERSE/CLOCKWISE/OUTPUT CHELSEA.NEW

Copying layer 1

Copying layer 2

Processing layer 111

SUMMARY:

57 features processed

13 features closed

9 features reversed

7 points deleted

ELAPSED: 00:00:08.27 CPU: 0:00:02.84 BUFIO: 4 DIRIO: 92 FAULTS: 120

\$

In this example the user typed the command IFIXAREA and was prompted for the input and output files. The input file specified was CHELSEA which is expanded to the default LSL\$IF:CHELSEA.IFF. The check qualifiers specified were /NEAR, /CLOSE, /REVERSE, as in the previous example but in this case the user specified the /LAYER and /FC qualifiers to limit the checking to features with a feature code of 30 within layer 111. The user also specified the /CLOCKWISE qualifier so that all features with the specified attributes would be checked for clockwise rotation. The /OUTPUT qualifier has the effect of sending supplementary messages about input and output file, the user and the different layers as they are encountered to SYS\$OUTPUT. These messages could have been directed to a file if a file specification had been given with the /OUTPUT qualifier. The output file specification given was CHELSEA.NEW (expanded to LSL\$IF:CHELSEA.NEW by default).

```
$ IFIXAREA CHELSEA/NEAR=1.3/CLOSE/LAY=111/REVERSE/CLOCKWISE/OUTPUT<CR>
_Fixed-IFF-file: CHELSEA.FIX<CR>
```

```
===== IFIXAREA =====
```

```
Input file:  LSL$IF:CHELSEA.IFF;
Output file:  LSL$IF:CHELSEA.FIX
```

```
IFIXAREA invoked by: TOM, Process: Tom Timms, on: 28-AUG-86 09:29:34
Command line:
IFIXAREA CHELSEA/NEAR=1.3/CLOSE/LAY=111/REVERSE/CLOCKWISE/OUTPUT CHELSEA.FIX
```

```
Copying layer 1
Copying layer 2
Processing layer 111
```

```
SUMMARY:
85 features processed
1 features closed
0 features reversed
1 point deleted
ELAPSED: 00:00:06.96  CPU: 0:00:02.91  BUFIO: 5  DIRIO: 91  FAULTS: 139
$
```

In this example the user specified almost exactly the same qualifiers as in the previous example except that the the qualifiers were typed on the same line as the IFIXAREA command. The input file specification is the same as in the previous example but in this case the /FC qualifier was omitted and an argument was specified with the /NEAR qualifier (1.3) to over-ride the default 'almost adjacent' distance. As no output file was specified the program prompted for one. The user typed CHELSEA.FIX as a file specification (confirmed by the output messages).

```
$ IFIXAREA NEWYORK.IFF/OUTPUT=NEWYORK.LIS/CLOSE NEWYORK.CLO<CR>
%LSLLIB-E-IFFOPEN, IFF error opening "LSL$IF:NEWYORK.IFF;" for read
%IFF-E-OPEN, error opening IFF file
-RMS-E-FNF, file not found
ELAPSED: 00:00:00.76  CPU: 0:00:00.30  BUFIO: 1  DIRIO: 1  FAULTS: 105
$
```

In this example the user specified that the file NEWYORK.IFF should be checked for feature closure with the output file being NEWYORK.CLO. An output file NEWYORK.LIS was also requested to contain a summary of program execution. However the input file NEWYORK.IFF was not found in LSL\$IF:, either the filename was incorrect or the logical name LSL\$IF: was not pointing to the correct directory or device.

\$ IFIXAREA DFADWOODS.IFF/OUT/DFAD/AC TESTDFAD<CR>

===== IFIXAREA =====

Input file: LSL\$IF:DFADWOODS.IFF;
Output file: LSL\$IF:TESTDFAD.IFF;0

IFIXAREA invoked by: TOM, Process: Tom Timms, on: 28-AUG-86 09:29:49
Command line:
IFIXAREA DFADWOODS.IFF/OUT/DFAD/AC TESTDFAD

Processing layer 2

SUMMARY:

324 features processed
40 features closed
19 features reversed
3 points deleted

ELAPSED: 00:00:41.16 CPU: 0:00:19.78 BUFIO: 10 DIRIO: 554 FAULTS: 126
\$

This example demonstrates the use of the /DFAD and /AC qualifiers on a suitable DFAD file DFADWOODS.IFF. The output file was specified as TESTDFAD which defaults to LSL\$IF:TESTDFAD.IFF. As /OUTPUT was specified supplementary messages were sent to SYS\$OUTPUT. The /AC qualifier caused selection of DFAD features on the basis of AC rather than feature code. These qualifiers are specific to DFAD type files and should not be used on non-DFAD files.

\$ IFIXAREA TTT/CLOSE/LAY=1/REVERSE/DELETE QQQ<CR>

SUMMARY:

85 features processed
35 features closed
25 features reversed
9 points deleted

ELAPSED: 00:00:07.85 CPU: 0:00:03.64 BUFIO: 4 DIRIO: 103 FAULTS: 121
\$

This example demonstrates a typical production run of IFIXAREA with no supplementary information requested by /LOG or /OUTPUT qualifiers. The file to be fixed was TTT.IFF and the requested checks (on layer 1 only) were for closure, for direction of digitising (with the default anticlockwise rotation assumed), and for repeated adjacent vertices (the /DELETE qualifier). The output IFF file was specified as QQQ.IFF. On program completion a summary was given of the number of features closed and reversed and the number of points deleted.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user.

DEFANTICLK, anticlockwise rotation defaults.

Explanation: The /REVERSE qualifier has been specified with no qualifier to define the rotation of features. The message confirms that the default anticlockwise rotation is taken.

User action: None.

FSNCLO, feature with FSN 'integer' ('integer') closed

Explanation: The specified feature has been closed.

User action: None.

FSNPTSDEL, 'integer' point(s) deleted in feature with FSN 'integer' ('integer')

Explanation: The /DELETE or /NEAR qualifiers have been selected and IFIXAREA is reporting that points have been deleted from the specified feature.

User action: None.

FSNREVD, feature with FSN 'integer' ('integer') reversed

Explanation: The specified feature has been reversed.

User action: None.

LAYCOP, copying layer 'integer'

Explanation: The specified layer in the input IFF file has been found, but it is not one of those selected by the /LAYER qualifier. The message confirms that the current layer is just being copied to the output IFF file.

User action: None.

LAYPROC, processing layer 'integer'

Explanation: The specified layer in the input IFF file has been found, and is being checked.

User action: None.

LSTOPNOUT, list file 'file-spec' opened for output.

Explanation: The /OUTPUT = 'file-spec' qualifier has been specified and the message is confirming that the output file has been opened.

User action: None.

NOCLPTS, feature with FSN 'integer' ('integer') not closed, only 'integer' point(s)

Explanation: The specified feature was not closed as it contained insufficient points.

User action: None.

NOCLINVIS, feature with FSN 'integer' ('integer') not closed, 'integer' invisible segment(s).

Explanation: The specified feature was not closed because it contained the specified number of invisible segment(s).

User action: None, unless invisible lines are not expected, when further investigation may be required.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

ACNOTYFNOVAL, no type field in AC 7 entry FSN 'integer' ('integer').

Explanation: The /DFAD and /AC qualifiers have been specified but the required type field is missing from the type 7 AC text field.

User action: Check that the DFAD data is correctly specified by the AC type 7 using IPATCH. If the AC text is missing then add the correct text using LITES2 or IPATCH and rerun the program.

EXTCOLIGN, Only coordinates processed in CB - feature FSN integer (integer)

Explanation: IFIXAREA has encountered a CB entry containing more than three X,Y,Z data columns. IFIXAREA does not reproduce non-coordinate columns in the output file.

User action: Check that no useful information is being lost.

LSTNSPLDEL, listing file not spooled or deleted

Explanation: The program has been unable to spool or delete the listing file specified by the /OUTPUT or /PRINT qualifier.

User action: Check that any listing file specification is correct. Check that the print queues are functioning correctly. If the problem continues notify the system manager.

LSTNSPL, listing file not spooled

Explanation: The program has been unable to spool the listing file specified by the /OUTPUT or /PRINT qualifier.

User action: Check that any listing file specification is correct. Check that the print queues are functioning correctly. If the problem continues notify the system manager.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ERRCLOINP, error closing input IFF file 'file-spec'
ERRCLOFIX, error closing output IFF file 'file-spec'

Explanation: An error has occurred in closing the specified IFF file.

User action: The accompanying messages should help you to interpret the cause of this error. Check that there is sufficient disk space for the file and that the disk is functioning properly. If the problem continues, notify the system manager.

ERROPLST, error opening list file 'file-spec'

Explanation: The supplied file specification is erroneous, or the file is nonexistent, or you do not have read access to the file.

User action: The accompanying messages will help you to interpret the cause of the error. Check the file specification for errors, check that you have read access to the file. If the problem continues, notify the system manager.

INVALFC, invalid Feature Code 'integer' specified

Explanation: An invalid feature-code has been specified either less than 0 or greater than 32767.

User action: Re-run the program with valid feature codes specified.

INVALLAY, invalid LAYER 'integer' specified

Explanation: An invalid layer has been specified either less than 0 or greater than 32767.

User action: Re-run the program with valid layer specified.

NOQUALIFIERS, no fixing qualifiers specified - aborting

Explanation: None of the checking/fixing qualifiers /CLOSE, /NEAR, /DELETE, /REVERSE, or /DFAD have been specified and the input file would merely be copied to the output file!

User action: Ensure that you have read this documentation before re-running IFIXAREA.

TOOMNYPTS, too many points in feature with FSN 'integer' ('integer')

Explanation: You have a very large feature ! There are more than 100000 points in the feature wiht FSN 'integer' and IFIXAREA cannot process more than that number of points.

User action: Either split the feature using LITES2 or reduce the number of points by thinning the feature using IFILTER.

UNEXPEOF, unexpected end of IFF file

Explanation: This message indicates there is something seriously wrong with the IFF file which has caused immediate termination of the program. IFIXAREA has detected the end of the IFF file, but has not detected an IFF 'EJ' entry.

User action: Use IMEND on the file, which will correctly position the EOF marker and insert an EJ entry at the end of the file. Re-run IFIXAREA on the corrected file.

MESSAGES (OTHER)

In addition to the above messages which are generated by IFIXAREA itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 6

MODULE IFROMTEXT

MODULE IFROMTEXT

REPLACES DAMP module TEXT2I

FUNCTION

IFROMTEXT is a program which converts data held in a text form into an IFF file. IFF files, such as template files or standard grids may be created from scratch, with data input either from a file or directly from the terminal. IFROMTEXT can also be used to edit existing IFF files in conjunction with ITOTEXT. This is done by converting an IFF file into text form using ITOTEXT, editing the text file, and creating a new IFF file from the edited text file using IFROMTEXT. The default file extensions are .IFF for IFF files and .TXT for text files.

FORMAT

\$ IFROMTEXT text-file-spec [IFF-file-spec]

Command qualifiers

Defaults

/[NO]ECHO

/NOECHO

/FRT = file-spec

No FRT file is used

/[NO]LOG

/NOLOG

/ORIGIN = (double_real, double_real)

/NOORIGIN

Input may be read from the terminal by specifying SYS\$INPUT: (or TT:) in place of the input filename.

PROMPTS

Text-file: text-file-spec

IFF-file: IFF-file-spec

If input is from the terminal then IFROMTEXT will prompt for each line of input with:

IFF entry>

PARAMETERS

text-file-spec

- specifies the text file from which an IFF file is to be made. Any part of the file name which is not specified will be taken from the default specification 'SYS\$DISK:IFFTEXT.TXT'.

IFF-file-spec

- specifies the IFF file which is to be created. If any part of the IFF-file specification is not explicitly given, the missing parts will be taken from:

'LSL\$IF:'input-file-spec'.IFF'.

Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

/ECHO
/NOECHO

- input file lines will be echoed as read in.

/FRT = file-spec

- specifies an FRT file which contains feature code graphical types, and an ACD definitions table. Any part of the file specification not supplied is taken from the default LSL\$FRT:FRT.FRT.

This qualifier should only be specified if non-standard ACD definitions are required to define attributes in CB or AC entries, or if text and symbol FS entry fields are to be set automatically (with some default values). See the IFF user guide for a description of standard ACD definitions and FS entry values.

/LOG
/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when files are successfully opened, and a reassuring message is output each time a new IFF layer is written to the IFF file. If missing IFF entries are created by IFFTEXT in order to maintain a valid IFF file structure, /LOG will cause detailed information about these entries to be output.

/ORIGIN = (double_real,double_real)
/NOORIGIN

- specifies an origin offset double real coordinate pair to subtract from any absolute coordinates in an absolute coordinate text file.

DESCRIPTION

General

IFROMTEXT creates a new IFF file from a text file or from data entered by the user at the terminal. IFF entries which are required to produce a valid IFF file are defaulted if not present. The following should be noted:

1. IFROMTEXT produces a default RA (Range) entry as the first entry in the IFF file if there is no RA entry in the text file, to be updated on completion of the text file reading. The user may give 4 values for the range limits after an RA command, which will be used in the output IFF file if all the feature coordinates supplied lie within these limits. If not, the RA values will be ignored and the range calculated from all the supplied coordinates, and the range entry updated accordingly.
2. IFROMTEXT produces a new HI (HISTORY) entry in the IFF file only if the text file contains a history entry after the RA entry or as the first entry in the text file.
3. IFROMTEXT produces a default empty MD2 (Type 2 Map Descriptor) entry in the IFF file if there was no MD entry in the text file. A type 1 map descriptor may be produced with the MD 1 command. See the section on Map Descriptors below for more details.
4. IFROMTEXT produces a new NS (New Section) entry in the IFF file if there was an NS entry in the text file using any text that appears after the NS command. If there was no NS entry, a default one is created from the time and date of program run.
5. IFROMTEXT produces a default CC (Corner Coefficients) entry in the IFF file if there was no CC entry in the text file.
6. IFROMTEXT produces a default CP (Control Points) entry in the IFF file if there was no CP entry in the text file, with values from 0 to 1000 in x and y. The user may supply the required entries for a new CP entry (see the section on IFROMTEXT default entries below for more specific CP entry format).
7. IFROMTEXT produces a default NO (New Overlay) entry in the IFF file if there was no NO entry in the text file with a layer number of 1.
8. IFROMTEXT produces a new NF (New Feature) entry in the IFF file if there was an NF entry in the text file, with a FSN and ISN incremented by 1. The user may supply the FSN to override the default count.

A new ISN (internal sequence number) which is the second value in the NF (New Feature) entry is created for each NF encountered in the text file. They are numbered monotonically, and any ISN values in the text file are ignored.

9. IFROMTEXT produces an FS (feature status) entry in the IFF file, the second element of the line being interpreted as the feature code (FC). Any further elements (only four integer elements in total allowed) are used for the rest of the FS entry fields. If an FRT file is supplied with the /FRT qualifier, and only the feature code is given, the feature code entry is read from the FRT, (a warning message being given if not present in the FRT), and the graphical type extracted. For text and symbol features the third FS entry field is set appropriately, with the rest set to defaults of 0 (so that the text feature positioning code is set to the SW corner).

If there were coordinates encountered without an FS entry, a default FS is created with a feature code of 0. This means that if the /FRT qualifier is given and there is no FC 0 in the FRT, a warning message will result. Also, if there is an FC 0 in the FRT which is a symbol or text feature, the third FS entry field will be set appropriately, with possibly unexpected results.

10. IFROMTEXT produces an AC (Attribute Code) entry in the IFF file for any AC entry in the text file. It checks that the second element of the line is an integer, or an ACD name, which has a valid ACD definition, either in the supplied FRT file, or as one of the default values always available when an FRT file is processed. The remaining line elements are read and interpreted as the AC value, and some range checking depending on the AC datatype is performed. If an ACD definition is not found, a default AC type of 0 is assumed, with an AC value of 0, with suitable warning messages.
11. IFROMTEXT produces an ST (Coordinate String) entry in the IFF file, and any further text file entries are interpreted as coordinates, until another IFF entry command is encountered. If there are further integers after the ST command, they are interpreted as the number of points to follow, and the pen up/down flag. This means that there must follow the correct number of coordinate pairs, and a warning message will be output if not, likewise if the pen flag is incorrect.
12. IFROMTEXT produces a ZS (3-dim string) entry in the IFF file, and any further text file entries are interpreted as coordinates. See the ST section above.
13. IFROMTEXT produces a CB (Coordinate Block) entry in the IFF file, and any further text file entries are interpreted as coordinates. See the section on CB entries below for a more detailed description on CB format.
14. IFROMTEXT produces an EM (End of Map) entry in the IFF file at the end of the text file being read.

15. IFFTEXT produces an EJ (End of Job) entry in the IFF file at the end of the text file being read.
16. IFFTEXT ignores junction pointer entries (SH,JB,JP). ITOTEXT and IFFTEXT cannot be used to edit IFF files with these features.
17. If the input file is invalid or incomplete in any way the appropriate IFF codes are given default values. Informational messages are output when this occurs. If /LOG is specified the default values may also be output.
18. Great care should be exercised in editing the text file. This is particularly important with the MH, MD, CC ,CP and CB entries where any small changes may have large repercussions in the IFF file produced.
19. If end-of-file is detected in the input file **before** an EJ (End of Job) entry has been read an EJ entry is defaulted. Any other entries needed to complete correctly the structure of the IFF file will be created before the defaulted EJ is written to the IFF file. The IFF file is then closed as if a real EJ entry had been read.
20. Comments in the textual representation of the IFF file may be used freely, all text to the right of a comment delimiter ('!') will be ignored. Blank lines may be inserted into the text file to aid clarity. These too will be ignored by IFFTEXT.

IFFTEXT and Map Headers

IFFTEXT produces a default empty MH 0 (Map Header) entry in the IFF file if there was no MH entry in the text file. The map header types and their intended users are:

- | | | |
|---|--|-----------------------|
| 0 | map header is unset | |
| 1 | MCE specific map header | - edit with MCEHED |
| 2 | OSGB specific type 2 map header | - edit with OMHED |
| 3 | OSGB specific type 3 (OSTF) map header | - edit with OS editor |
| 4 | OSGB specific type 4 (CITF) map header | - edit with OS editor |

MCEHED is a utility supplied with the customer-specific package MCE. OSMHED is a utility in the IFFOSTF module of the CONVERT package.

Type 1 map headers are MCE specific and not intended for use by others.

Type 2 map headers are intended for users of OSTF data to hold OS specific header information needed during conversion to and from IFF.

Type 3 OS map headers are intended for use by Ordnance Survey (Great Britain) only and will be edited using OS developed header editors. If the text file contains a type 3 MH entry then the size of the entry is taken from the translation table LSL\$OS_MH_TABLE and the header will be copied to the IFF file as an ASCII block. Note that, for type

3 OSTF headers, the size of the header will be the sum of the -1 and -20 header sizes specified in the translation table.

For type 3 map headers, the sizes (OSTF_MH_ONE and OSTF_MH_TWENTY) in the lookup table are intended to be static for OSTF and are assumed to be correct. This number of bytes will be used and if it is too small then not all of the header will be copied to the IFF MH entry, the remainder being skipped, and the next entry (the MD) will be processed. If the size in the table is larger than the MH entry, then there is a danger that some of the MD entry will be copied in to the header and the MD entry will not be found. However, a default type 2 map descriptor will be created. It is therefore important to ensure that the header size(s) accurately reflect the size of the text MH entry.

If a type 3 header is being manually edited into the text file then it is safer to have a header that is longer than the size specified in the translation table. Any totally blank text lines must be represented as 80 spaces rather than a carriage return.

Type 4 CITF map headers are intended for use by Ordnance Survey (Great Britain) only and will be edited using OS developed header editors. Since type 4 map headers are variable in length, they are copied in lines (of 80 characters each) to the MH entry which will be only as large as required for number of lines in the text file. The size in the lookup table LSL\$OS_MH_TABLE is not used, but the program reads lines until the map descriptor is found which is denoted by 'MD' or '!MD'. Since the first two characters of a CITF header line must be a numeric record descriptor, there should be no text in the header which could be mistaken for the Map Descriptor.

If the MH entry is invalid or missing then a default type 0 header is created. An invalid MH entry is one where the MH entry size is not 0 or 174. This is because MH type 0 has a entry size of zero and MH type 2 (OS) has an entry size of 174 words. For type 3 headers, the size is taken from the translation table and so the size in the MH entry should be zero.

For details of how to set up the translation table, refer to the DATA PREPARATION section of the IFFOSTF chapter in the "Convert User Guide".

A typical type 2 MH entry, when converted to text using ITOTEXT may look like:

MH 174 2

0	436500	122000	1250	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	436500	122000	0	0
50	1250	6500	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	4260880	4260880
4260880	4260880	0	0	0	0
0	0	0	0	0	0

```

      0      0      0      0      0      0
      0      0      0      0      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
      0      0      0      9439170      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0

```

Here is an example of a type 3 OS Map Header when converted to text as an ASCII block using ITOTEXT.

```

MH 0 3
-1  0  436500  122000      50      1250      0      6500 -20      1
      9  86      0      0      0      0      0      0      0      0      0      0      0      0      0      0
  0  0      0      0      0      0      0      0      0      0      0      0      0      0      0      0
  0  0
      0      0

```

IFROMTEXT and Map Descriptors

Two text representations of a version 2 MD (Map Descriptor) entry are produced by the ITOTEXT program. The first is unformatted and is included if ITOTEXT is used for outputting the IFF file for subsequent retranslation with IFROMTEXT, eg:

```

MD      2      17352      0      17224
      0      17786      0      0      0
0
      0      0      0      0      0
0
      etc.

```

The second formatted version 2 MD is preceded by a '!' and is thus ignored by IFROMTEXT, eg:

```

!MD LO  0.0000000000000000  0.0000000000000000
!  SC  0.0000000000000000
!  PR   0
!  UN   0
!  SP   0

```

While it is not recommended to alter map descriptor fields it is

recognised that the user may wish to set the local origin and the scale of the IFF file for subsequent processing by ITRANS. This may be done by inserting LO and SC entries either immediately before or after the MD entry in the text file, eg:

```
LO 500 500
SC 50000
MD "unformatted map descriptor values here"
```

The LO and SC values will then be recognised by IFROMTEXT and included in the appropriate map descriptor fields. Running ITRANS/DESCRIPTOR will verify that the correct values have been inserted in the map descriptor. It must be stressed that LO and SC are not valid IFF entries and are not produced by ITOTEXT.

The LO and SC entries are invalid with a version 1 map descriptor. IFROMTEXT considers a version 1 map descriptor to be one where an alphabetic character follows the MD rather than the numeric values of the unformatted version 2, eg:

```
MD MA -1      0.0      0.0      0.0      0.0
GR  0      0.0      0.0      0.0      0.0
SC      0.0
PS  0  0
AG  0      0.0      0.0      0.0      0.0
```

IFROMTEXT and the /ORIGIN qualifier

IFROMTEXT can read a text file containing double precision (absolute) coordinates (usually created by using the ITOTEXT/ABSOLUTE process), subtracting the supplied origin offset from all coordinate values and outputting an IFF file in local coordinates relative to this origin.

IFROMTEXT and the CB entry

The CB entry is represented in textual form by a number of sub-entries which may be input to IFROMTEXT. A valid text form of the CB entry is as follows:-

CB FL 0	! Flags
GT 1	! Graphical Type
NR 4	! Number of Rows of Data
NC 4	! Number of Columns
NA 2	! Number of Attributes
AT 3 45.2	! Attribute
AT SECONDARY_CODE 8	! Attribute
HC X Y Z 1006	! 4 Column Headers
DA 56.5 54.1 10.1 45	! 4 rows of DATA
21.98 23.5 10.8 62	
89.5 67.2 78.9 ?	! note the absent value
12.34 56.7 31.6 8	

While IFROMTEXT will produce default values if any of the sub-entries are omitted or are invalid it is important that: -

1. The correct number of columns is included.
2. Data values which are absent should be denoted by a ? for numeric, date or time values, or "" for character values, rather than omission.

If either of these conditions is not met then the data entry in the IFF file produced may be corrupt.

Note that attribute codes may be input either by their numeric code (eg 1006), or by the attribute name (eg X) in the ACD table. ITOTEXT, however, always outputs the numeric codes.

Character attributes (maximum 4 characters) should be enclosed in double quote markers while date and time attributes should be input in the standard VMS formats. ie.

13-OCT-1962
23:11:56.2

The IFF User guide should be consulted for a detailed description of the CB entry.

IFFTEXT and Default Procedures

The user is not expected to produce a structurally complete textual representation of an IFF file using a text editor. Any IFF entries which are out of sequence or which are missing will be defaulted and the user informed. Which features are defaulted depends on what level of feature has been input by the user. For example if only an MD entry is included in the input file then only features at map level and above will be defaulted. (MH,MD,EM,RA,EJ). The IFF user guide should be consulted for a detailed description of IFF file structure.

One potentially disruptive default is that used for a missing ST entry. This consists of a single point having the coordinates (0.0,0.0). This will clearly affect the RA (Range) entry of the IFF file. The inexperienced user is recommended to use the /LOG qualifier which will output more detailed messages regarding the default IFF entries generated.

If the input file contains an entry which is incomplete or contains invalid values, then a message may be output informing the user that the remainder of the entry is defaulted. In the case of the CP (Corner points) entry, 16 values are expected. The following is an example of a deficient CP entry in the input file :

```
CP 0 500 0 500
   0 0   0 0
```

After IFFTEXT and a subsequent ITOTEXT the CP entry would appear as follows :

```
CP      0.0      500.0      0.0      500.0
        0.0      0.0      0.0      0.0
      1000.0      0.0      1000.0      0.0
      1000.0      1000.0      1000.0      1000.0
```

The last 8 CP values (which define the SE and NE control points) are IFFTEXT defaults and are not necessarily acceptable to the user. The user should use

IPATCH/WRITE to edit the CP entry.

EXAMPLES

```
$ IFROMTEXT<CR>
_Text-File: TST1/LOG<CR>
_IFF-File: TST1<CR>
%LSLLIB-I-IFFOPENED, DUA0:[BUREAU]TST1.IFF;2 opened for write
%IFROM-I-OPTXT, text file DUA0:[BUREAU]TST1.TXT;0 opened for input
%IFROM-W-ILLEGCUST, wrong customer in MH entry near line 5
%IFROM-I-REMMHDEF, remainder of MH entry defaulted
%IFROM-I-SKIPPING, skipping to next entry
%IFROM-I-LAYWRT, layer 1 written
%IFROM-I-LAYWRT, layer 88 written
%IFROM-I-LAYWRT, layer 66 written
%IFROM-I-LAYWRT, layer 55 written
%IFROM-I-LAYWRT, layer 44 written
%IFROM-I-LAYWRT, layer 11 written
  ELAPSED: 00:00:37.02  CPU: 0:00:08.92  BUFIO: 6  DIRIO: 199  FAULTS: 335
$
```

This example demonstrates how IFROMTEXT prompts for the input and output files. The /LOG qualifier output indicates the defaults applied to the file-specs supplied. The MH (Map Header) entry in the text file contained an invalid MH customer code. IFROMTEXT reports this and outputs a default, all zero, MH entry to the IFF file. The user will have to use IPATCH/WRITE to insert the correct values and customer type into the MH entry.

```
$ IFROMTEXT RRRROM.TXT/LOG ROM.IFF<CR>
%LSLLIB-I-IFFOPENED, DUA0:[BUREAU]RRROM.IFF;1 opened for write
%IFROM-I-OPTXT, text file DUA0:[BUREAU]RRROM.TXT;0 opened for input
%IFROM-W-ILLEGRAIGN, illegal RA entry ignored at line 1
%IFROM-I-SKIPPING, skipping to next entry
%IFROM-I-LAYWRT, layer 1 written
%IFROM-I-LAYWRT, layer 11 written
%IFROM-I-CALCRAOUT, calculated RA entry written to IFF file
  ELAPSED: 00:01:10.77  CPU: 0:00:16.93  BUFIO: 6  DIRIO: 252  FAULTS: 324
$
```

Here the user specifies both the input text file and output IFF file on a single command line. IFROMTEXT is unable to read the contents of the RA (Range) entry in the text file and so calculates range values as ST and ZS points are written to the IFF file. When the EJ is read from the text file, or when end-of-file is detected, the calculated range values are patched into the RA entry in the IFF file. The run completed normally. \$STATUS is set to SS\$_NORMAL.


```
$ IFROMTEXT SYS$INPUT: TSTSYS<CR>
IFF entry> NO 1<CR>
%IFROM-I-RADEFAULT, RA entry defaulted
%IFROM-I-MHDEFAULT, MH entry defaulted
%IFROM-I-MD2DEFAULT, type 2 MD entry defaulted
%IFROM-I-NSDEFAULT, NS entry defaulted
%IFROM-I-CCDEFAULT, CC entry defaulted
%IFROM-I-CPDEFAULT, CP entry defaulted
IFF entry> NO 1<CR>
IFF entry> NF 1 1<CR>
IFF entry> ST 1 0<CR>
IFF entry> 23.3 56.7<CR>
IFF entry> EF<CR>
IFF entry> EO<CR>
IFF entry> <CTRL/Z>
%IFROM-I-EOF EJDEF, end-of-file detected - EJ entry defaulted
%IFROM-I-EMDEFAULT, EM entry defaulted
%IFROM-I-CALCRAOUT, calculated RA entry written to IFF file
ELAPSED: 00:02:04.09 CPU: 0:00:03.47 BUFIO: 4 DIRIO: 133 FAULTS: 189
$
```

Because SYS\$INPUT: is specified, IFROMTEXT prompts for the input from the terminal. IFROMTEXT fills in default values for those required IFF entries which are not included. The user terminated the session by pressing <CTRL/Z> instead of typing an 'EJ' entry. As the current map was still open in the output IFF file IFROMTEXT generated a default EM entry to close the map. The run completed normally. \$STATUS is set to SS\$NORMAL.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user. They are used to provide information on the current state of the program, or to supply explanatory information in support of a warning or error message.

ACTXTDEF, AC text defaulted

Explanation: An error occurred while reading the text part of an AC entry command, and default text will be substituted.

User action: Make a note of the feature which has erroneous text and rectify it using LITES2 or edit the textual representation of the IFF file and then re-run IFROMTEXT.

AMENDED, Amended to number counted

Explanation: This message is output when IFROMTEXT has changed a count header value in an ST, ZS or CB entry to the number of subsequent values actually counted.

User action: None.

ATDEFAULT, Attribute value defaulted

Explanation: An error occurred during the reading of the AT (Attribute) field in a CB entry or an expected AT field was not present. IFROMTEXT has included a default value in the CB entry.

User action: None.

CALCRAOUT, Calculated RA entry written to IFF file

Explanation: IFROMTEXT has calculated new range values from the coordinates in the file.

User action: None.

CBDATADEF, CB data defaulted

Explanation: IFROMTEXT has defaulted some CB data.

User action: Use IPATCH to determine whether the default CB entry is acceptable.

CCDEFAULT, CC entry defaulted

Explanation: IFROMTEXT did not find a CC entry command or the CC entry command was not in the correct position within the input file and has generated a default CC entry.

User action: Use IPATCH to determine whether the default CC entry is acceptable.

CHDEFAULT, CH entry defaulted

Explanation: IFFTEXT was unable to read the CH entry command and generated a default CH entry.

User action: Use IPATCH to determine whether the default CH entry is acceptable.

CONSLSL, Consult Laser-Scan as text ST (String) entry has more than 'integer' points

Explanation: IFFTEXT has an internal buffer dimensioned to receive a maximum of 10000 coordinates in a single string. The buffer is segmented into ST's of 200 coordinates on output to the IFF file. The current textual ST entry has more than 10000 coordinates.

User action: Use a text editor on the input file and use more ST entry commands within the indicated IFF feature to reduce the number of points after any single ST to be less than 10000.

CPDEFAULT, CP entry defaulted

Explanation: IFFTEXT did not find a CP entry command or the CP entry command was not in the correct position within the input file. IFFTEXT has generated a default CP entry.

User action: Use IPATCH to determine whether the default CP entry is acceptable.

CSDEFAULT, CS entry defaulted

Explanation: IFFTEXT was unable to read the CS entry command and has generated a default CS entry.

User action: Use IPATCH to determine whether the default CS entry is acceptable.

DATVALDEFAULT, CB Data value defaulted

Explanation: An error occurred during the reading of the CB data fields in a CB entry. IFFTEXT has included a default value in the CB entry.

User action: None.

EMDEFAULT, EM entry defaulted

Explanation: IFFTEXT did not find a EM entry command or the EM entry command was not in the correct position within the input file. IFFTEXT has generated a default EM entry.

User action: Use IPATCH to determine whether the default EM entry is acceptable.

ENDASS, End of entry assumed on line 'integer'

Explanation: IFFTEXT has assumed the entry is complete at the given line number.

User action: Check the input file to ensure that its decision was correct.

ENTIGN, 'string' entry starting at line 'integer' ignored

Explanation: The specified entry in the text file will not be translated to the IFF file.

User action: Check the input file.

EODEFAULT, EO entry defaulted

Explanation: IFFTEXT did not find an EO entry command or the EO entry command was not in the correct position within the input file. IFFTEXT has generated a default EO entry.

User action: Use IPATCH to determine whether the default EO entry is acceptable.

EOFEJDEF, End-of-file detected - EJ entry will be defaulted

Explanation: IFFTEXT has encountered end-of-file in the input file before an IFF EJ (End Job) entry command was read. IFFTEXT has generated a default EJ entry, and indeed any other entries needed to close correctly the structure of the IFF output file.

User action: Use IPATCH to determine whether the default closure of the IFF file is acceptable.

EXTRAIGN, Extra values ignored

Explanation: IFFTEXT has found an unexpected value in the file and has ignored it.

User action: None.

FLAGSDEFAULT, Flags in CB entry defaulted

Explanation: An error occurred during the reading of the FL (Flags) field in a CB entry or the FL field was not present. IFFTEXT has included a default value in the CB entry.

User action: None.

FSDEFAULT, FS entry defaulted

Explanation: IFROMTEXT did not find a FS entry command within the current feature or the FS entry command was not in the correct position within the feature and has generated a default FS entry.

User action: Use IPATCH to determine whether the default FS entry is acceptable.

GTPEDEFAULT, Graphical Type in CB entry defaulted

Explanation: An error occurred during the reading of the GT (Flags) field in a CB entry or the GT field was not present. IFROMTEXT has included a default value in the CB entry.

User action: None.

HCDDEFAULT, Column Headers in CB entry defaulted

Explanation: An error occurred during the reading of the HC (Column Headers) field in a CB entry or an expected HC field was not present. IFROMTEXT has included a default value in the CB entry.

User action: None.

HINOTOUT, HI entry not output at line 'integer'

Explanation: The HI entry was not output.

User action: Check the input file.

LAYWRT, Layer 'integer' written

Explanation: The layer with the specified number has been written to the IFF file.

User action: None.

LODEFAULT, LO defaulted

Explanation: An error occurred in the reading of an LO command. Default values have been included in the local origin field in the output map descriptor. The defaults will be taken from the input MD if it exists.

User action: Action will be determined by the accompanying warning message.

MD2DEFAULT, Type 2 MD entry defaulted

Explanation: IFROMTEXT did not find an MD entry command or the MD entry command was not in the correct position within the input file and has generated a default type 2 MD entry. Use ITRANS to set any origin offset or projection information.

User action: Use IPATCH to determine whether the default MD entry is acceptable. It will, of course, be entirely unset.

MHDEFAULT, MH entry defaulted

Explanation: IFROMTEXT did not find a MH entry command or the MH entry command was not in the correct position within the input file. IFROMTEXT has generated a default MH entry.

User action: Use IPATCH to determine whether the default MH entry is acceptable. It will, of course, be entirely unset.

NATTDEFAULT, Number of Attributes in CB entry defaulted

Explanation: An error occurred during the reading of the NA (Number of Attributes) field in a CB entry or the GT field was not present. IFROMTEXT has included a default value in the CB entry.

User action: None.

NCOLDEFAULT, Number of Columns in CB entry defaulted

Explanation: An error occurred during the reading of the NC (Number of Columns) field in a CB entry or the GT field was not present. IFROMTEXT has included a default value in the CB entry.

User action: None.

NFDEFAULT, NF entry defaulted

Explanation: IFROMTEXT did not find a NF entry command or the NF entry command was not in the correct position within the input file. IFROMTEXT has generated a default NF entry.

User action: Use IPATCH to determine whether the default NF entry is acceptable.

NODEFAULT, NO entry defaulted

Explanation: IFROMTEXT did not find a NO (new layer) entry command or the NO entry command was not in the correct position within the input file. IFROMTEXT has generated a default NO entry.

User action: Use IPATCH to determine whether the default NO entry is acceptable.

NOSTATDEF, NO status defaulted

Explanation: IFROMTEXT did not find a NO (new layer) status, IFROMTEXT has generated a default NO entry.

User action: Use IPATCH to determine whether the default NO entry is acceptable.

NROWDEFAULT, Number of rows in CB entry defaulted

Explanation: An error occurred during the reading of the NR (Number of Rows) field in a CB entry or the GT field was not present. IFFTEXT has included a default value in the CB entry.

User action: None.

NSDEFAULT, NS entry defaulted

Explanation: IFFTEXT did not find an NS entry command, IFFTEXT failed to read the NS entry command, or the NS entry command was not in the correct position within the input file. IFFTEXT has generated a default NS entry.

User action: Use IPATCH to determine whether the default NS entry is acceptable.

OPTXT, Text file 'file-spec' opened for input

Explanation: The specified text file has been successfully opened.

User action: None.

RADEFAULT, RA entry defaulted

Explanation: IFFTEXT did not find a RA entry command or the RA entry command was not in the correct position within the input file. IFFTEXT has generated a default RA entry.

User action: Use IPATCH to determine whether the default RA entry is acceptable.

REMACDEF, Remainder of AC entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

REMCDEF, Remainder of CC entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

REMCHIGN, Remaining characters ignored

Explanation: The remaining characters in the IFF entry are ignored.

User action: None.

REMCPCDEF, Remainder of CP entry defaulted

Explanation: An error in the entry command resulted in IFFTEXT defaulting the remaining values.

User action: User action will depend upon the accompanying warning message.

REMFSDDEF, Remainder of FS entry defaulted

Explanation: An error in the entry command resulted in IFFTEXT defaulting the remaining values.

User action: User action will depend upon the accompanying warning message.

REMHCDDEF, Remainder of Header codes defaulted

Explanation: An error occurred while reading the HC (Column Header) fields of the CB entry or some of the expected values were not present. This has resulted in IFFTEXT defaulting the remaining values.

User action: User action will depend upon the accompanying warning message.

REMMDDDEF, Remainder of MD entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

REMMHDEF, Remainder of MH entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

REMSTDEF, Remainder of ST entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

REMTSDEF, Remainder of TS entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

REMZSDEF, Remainder of ZS entry defaulted

Explanation: The remaining characters in the IFF entry are defaulted.

User action: None.

RODEFAULT, RO entry defaulted

Explanation: An error occurred in the reading of an RO command. A default value has been included in the rotation field in the output.

User action: See accompanying warning message.

SCDEFAULT, SC defaulted

Explanation: An error occurred in the reading of an SC command. A default value has been included in the scale factor field in the output map descriptor. The default will be taken from the input MD if one exists.

User action: See accompanying warning message.

SKIPPING, Skipping to next entry

Explanation: IFROMTEXT is ignoring all text until another IFF entry command or another CB entry command is found.

User action: None.

SLDEFAULT, SL entry defaulted

Explanation: IFROMTEXT could not read an SL entry command or the SL entry command was not in the correct position within the input file. IFROMTEXT has generated a default SL entry.

User action: Use IPATCH to determine whether the default SL entry is acceptable.

SSDEFAULT, SS entry defaulted

Explanation: IFROMTEXT could not read a SS entry command the SS entry command was not in the correct position within the input file. IFROMTEXT has generated a default SS entry.

User action: Use IPATCH to determine whether the default SS entry is acceptable.

STDEFAULT, ST entry defaulted

Explanation: IFROMTEXT did not find an ST entry command or the ST entry command was not in the correct position within the input file. IFROMTEXT has generated a default ST entry.

User action: Use IPATCH to determine whether the default ST entry is acceptable.

TCDEFAULT, TC entry defaulted

Explanation: IFROMTEXT could not read a TC entry command. IFROMTEXT has generated a default TC entry.

User action: Use IPATCH to determine whether the default TC entry is acceptable.

THDEFAULT, TH entry defaulted

Explanation: IFROMTEXT could not read a TH entry command or the TH entry command was not in the correct position within the input file. IFROMTEXT has generated a default TH entry.

User action: Use IPATCH to determine whether the default TH entry is acceptable.

TSDEFAULT, TS entry defaulted

Explanation: IFROMTEXT did not find a TS entry command or the TS entry command was not in the correct position within the IFF feature. IFROMTEXT has generated a default TS entry.

User action: Use IPATCH to determine whether the default TS entry is acceptable.

TXDEFAULT, TX entry defaulted

Explanation: IFROMTEXT could not read a TX entry command or the TX entry command was not in the correct position within the input file. IFROMTEXT has generated a default TX entry instead.

User action: Use IPATCH to determine whether the default TX entry is acceptable.

ZSDEFAULT, ZS entry defaulted

Explanation: IFROMTEXT could not read a ZS entry command or the ZS entry command was not in the correct position within the input file. IFROMTEXT has generated a default ZS entry.

User action: Use IPATCH to determine whether the default ZS entry is acceptable.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user or that the program will attempt to overcome.

ACAFTERSTIGN, AC after ST or ZS entry ignored at line 'integer'

Explanation: An AC entry is present after an ST or ZS entry. This is illegal.

User action: Check the input file.

CHAROUTRAN, Character value 'char' outside range on line 'integer'

Explanation: A character attribute value in a CB entry is outside the range specified in the ACD table.

User action: Check the input file.

CHRTRUNC, Character string too long - truncated on line 'integer'

Explanation: A character string is too long for the maximum 4 character attribute value in the CB entry and has been truncated.

User action: Check the input file to ensure no important information has been lost.

CORRUPT, 'string', 'char', on line 'integer' is corrupted

Explanation: The whole line may be corrupt up to the given character.

User action: Check the input file.

DATOUTRAN, Date value 'string' outside range on line 'integer'

Explanation: A date attribute value in a CB entry is outside the range specified in the ACD table.

User action: Check the input file.

ERRINMH, Error in MH entry length given on line 'integer'

Explanation: There is an error in the given length of the MH entry command.

User action: Check the input file.

HCAFTERDA, HC after DA in CB ignored on line 'integer'

Explanation: An HC (Column Headers) field is present after the DA (data field) in a CB entry. This is illegal.

User action: Check the input file.

ILLEGCHAR, 'string' expected on line 'integer', illegal 'char' found

Explanation: An unexpected character has been found on the given line, instead of the specified variable type (eg. integer).

User action: Check the input file.

ILLEGCUST, Wrong customer in MH entry at line 'integer'

Explanation: There is an error in the customer value of the MH entry command.

User action: Check the input file.

ILLEGLINIGN, Expecting IFF code, line 'integer' ignored

Explanation: An IFF entry command was expected, but not found.

User action: Check to see that the invalid line does not hold important information.

ILLEGLOMDIGN, Illegal LO with version 1 map descriptor ignored

Explanation: An LO command has been found with a version 1 MD. It is only valid with a version 2 MD.

User action: Use ITRANS to set up a version 2 map descriptor if required.

ILLEGNEGAC, Illegal negative AC on line 'integer'

Explanation: A negative AC code has been found in the input file.

User action: Check the input file.

ILLEGNEGFS, Illegal negative FS on line 'integer'

Explanation: A negative feature code has been found in the input file.

User action: Check the input file.

ILLEGNEGNF, Illegal negative NF entry on line 'integer'

Explanation: A negative feature serial number has been found in the input file.

User action: Check the input file.

ILLEGNEGNO, Illegal negative NO entry on line 'integer'

Explanation: A negative layer number code has been found in the input file.

User action: Check the input file.

ILLEGRAIGN, Illegal RA entry ignored at line 'integer'

Explanation:

User action:

ILLEGSCMDIGN, Illegal SC with version 1 map descriptor ignored

Explanation: An SC command has been found with a version 1 MD. It is only valid with a version 2 MD.

User action: Use ITRANS to set up a version 2 map descriptor if required.

ILLEGSEP, Illegal separator 'char' on line 'integer'

Explanation: The values on the given line are separated by an illegal character.

User action: Check that the separator is a space or a comma.

ILLEGSTAT, Illegal status value on line 'integer'

Explanation: The NO status value should be zero.

User action: IFFROMTEXT defaults to zero.

ILLEGTHIGN, TH entry not allowed here and ignored at line 'integer'

Explanation: An IFF entry has been found in an illegal position.

User action: Check the input file.

INTOUTRAN, Integer value 'integer' outside range on line 'integer'

Explanation: An integer attribute value in a CB entry is outside the range specified in the ACD table.

User action: Check the input file.

MDOUTORDER, MD code on line 'integer' out of order

Explanation: The version 1 MD codes are in the wrong order.

User action: Use IPATCH or IINFO to check MD entry in input file.

MISSHC, Too few Header codes in CB entry at line 'integer'

Explanation: IFFROMTEXT expected more header codes in the HC field of the CB entry. The number expected is the value specified in the NC (Number of Columns) field of the entry.

User action: Check the input file.

MISSTDEF, Missing ST entry defaulted for feature at line 'integer'

Explanation: A feature has no ST entry.

User action: Check the input file.

MISSTS, Missing TS entry before line 'integer' - IFF file is corrupt

Explanation: A feature has no TS entry.

User action: Check the input file.

MISSTXT, No text given on line 'integer'

Explanation: A TX entry contains no text.

User action: Check the input file.

NAAFTERAT, NA after AT in CB ignored

Explanation: An NA (Number of Attributes) field is present after the AT (Attributes) in a CB entry. This is illegal.

User action: Check the input file.

NAOUTRAN, Number of attributes 'integer' outside range in CB on line 'integer'

Explanation: The number of attributes specified in the NA field of the CB entry is outside the maximum allowable number.

User action: Check the input file.

NCAFTERDA, NC after DA in CB ignored on line 'integer'

Explanation: An NC (Number of Columns) field is present after the DA (data) in a CB entry. This is illegal.

User action: Check the input file.

NCAFTERHC, NC after HC in CB ignored on line 'integer'

Explanation: An NC (Number of Columns) field is present after the HC (Column Headers) in a CB entry. This is illegal.

User action: Check the input file.

NCOUTRAN, Number of columns 'integer' outside range in CB on line 'integer'

Explanation: The number of rows specified in the NR field of the CB entry is outside the maximum allowable number (currently 20).

User action: Check the input file.

NPTSOUTRAN, 'integer' points for ST or ZS on line 'integer' out of range

Explanation: The number of points specified in the ST or ZS entry command is outside the current string range of 1 to 10000. A negative string count is illegal.

User action: Edit the file and split string into smaller strings, none of which have more than 10000 coordinates.

NRAFTERDA, NR after DA in CB ignored on line 'integer'

Explanation: An NR (Number of Rows) field is present after the DA (Data) in a CB entry. This is illegal.

User action: Check the input file.

NROUTRAN, Number of rows 'integer' outside range in CB on line 'integer'

Explanation: The number of rows specified in the NR field of the CB entry is outside the maximum allowable number (currently 200). IFFRTEXT will split CBs such that none has more than 200 rows of data.

User action: Check the input file.

REAOUTRAN, Real value 'real' outside range on line 'integer'

Explanation: A real attribute value in a CB entry is outside the range specified in the ACD table.

User action: Check the input file.

REMPTSLOST, Remainder of points lost

Explanation: Some data points have been lost.

User action: Check the input file.

REPATIGN, Repeated AT in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFRTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPCCIGN, Repeated CC ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFRTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPCIGN, Repeated CP ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPDAIGN, Repeated DA in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPFLIGN, Repeated FL in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPFSIGN, Repeated or assumed FS entry ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPGTIGN, Repeated GT in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPHCIGN, Repeated HC in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPLOIGN, Repeated LO ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPMIGN, Repeated MD ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPMHIGN, Repeated MH ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPNAIGN, Repeated NA in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPNCIGN, Repeated NC in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPNOIGN, Repeated or assumed NO ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPNRIGN, Repeated NR in CB ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPNSIGN, Repeated NS ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPRAIGN, Repeated RA ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPROIGN, Repeated RO entry ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPSCIGN, Repeated SC ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPTHIGN, Repeated TH entry ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

REPTXIGN, Repeated TX entry ignored at line 'integer'

Explanation: The specified IFF entry command has been illegally repeated or repeated after being assumed by IFFTEXT.

User action: Check the input file for repeated entries or invalid structure.

STAFTERZSIGN, ST entry after ZS in same feature at line 'integer' ignored

Explanation: An ST entry occurred illegally in the same feature as a previous ZS entry. Strings of different dimensions may not be mixed.

User action: Check the input file.

TIMOUTRAN, Time value 'string' outside range on line 'integer'

Explanation: A time date attribute value in a CB entry is outside the range specified in the ACD table.

User action: Check the input file.

TOOMNYHC, More Header Codes than number of columns in CB

Explanation: More header codes than the number of columns specified in the NC field of the CB entry were found.

User action: Check the input file.

TXTOUTRAN, Text on line 'integer' exceeds maximum 'integer' characters

Explanation: The text on the line is greater than the current limit of 255 characters for text strings.

User action: Amend the text entry to fit.

UNEXPEOE, Unexpected end of entry on line 'integer'

Explanation: The data for an entry command has been terminated prematurely.

User action: Check the given line to see that all the required values are present.

UNEXPMDENT, Expecting MA, GR, SC, PS, or AG on line 'integer'

Explanation: An unknown version 1 MD code is present.

User action: Check the MD values.

UNKNACDNAM, Attribute Name 'string' not found in ACD

Explanation: The ACD name specified as the second element of the text file AC entry line was not found in either the FRT file specified, or in the default ACs supplied.

User action: Check the input file.

UNKNCUST, Unknown customer given in MH entry on line 'integer'

Explanation: The MH entry command contains an unknown customer value.

User action: Check the input file.

WRNGNATT, Wrong number of attributes in CB entry on line 'integer'

Explanation: The number of attributes specified in the NA field of the CB is different from the number of attributes actually read.

User action: Check the input file.

WRNGNPTS, Incorrect number of points in ST or ZS at line 'integer'

Explanation: The number of points specified in the ST/ZS entry command does not correspond with the number of coordinates actually given.

User action: Check the input file.

WRNGNROW, Wrong number of data rows in CB entry on line 'integer'

Explanation: The number of rows of data read in the NR field of the CB is different from the number of rows actually read in the DA field.

User action: Check the input file.

ZSAFTERSTIGN, ZS entry after ST in same feature at line 'integer' ignored

Explanation: A ZS entry occurred illegally in the same feature as a previous ST entry. Strings of different dimensions may not be mixed.

User action: Check the input file.

MESSAGES (ERROR)

These messages indicate an error in processing which will cause the program to terminate. The most likely causes are a corrupt or otherwise invalid input file, or an error related to command line processing and file manipulation.

ERRCLOTXT, Error closing input file 'string'

Explanation: An error has occurred during file closing.

User action: The accompanying messages will help you to interpret the cause of the error. If the problem continues, notify the system manager.

ERROPTXT, Error opening text file 'string' for input

Explanation: The supplied file specification is erroneous, or the file is non-existent, or you do not have read access to the file.

User action: The accompanying messages will help you to interpret the cause of the error. Check the file specification for errors, check that you have read access to the file. If the problem continues, notify the system manager.

ERRORIGIN, Error giving origin offset values with /ORIGIN

Explanation: An error has occurred during the reading of the /ORIGIN qualifier values.

User action: Resubmit the command line.

ERRREADFRT, Error reading from FRT file 'string'

Explanation: An error has occurred while trying to read from the input FRT file.

User action: The accompanying messages will help you to interpret the cause.

ERRREADINP, Error reading from input at line 'integer'

Explanation: An error has occurred while trying to read from the input file.

User action: The accompanying messages will help you to interpret the cause of the error.

MESSAGES (FATAL)

These messages indicate a severe error in processing, or some form of system failure, which has caused the program to terminate.

UNKNIFFENT, Unknown IFF code 'char' - IFFTEXT internal consistency error -
please notify Laser-Scan

Explanation: IFFTEXT has an internal consistency error related to the specified IFF entry command.

User action: This error should not occur. Please preserve both the input and output files and then notify Laser-Scan.

MESSAGES (OTHER)

In addition to the above messages which are generated by the program itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library and by the Laser-Scan I/O library, LSLLIB. IFF library messages are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user, and Laser-Scan are consulted, then the output file should be preserved to facilitate diagnosis. LSLLIB messages are introduced by '%LSLLIB' and are generally self-explanatory. They are used to explain the details of program generated errors.

CHAPTER 7

MODULE 1 INFO

MODULE IINFO

REPLACES DAMP modules IFINFO and SCI

FUNCTION

IINFO gives brief general information about the specified IFF file.
The default information given is as follows:

- o Full file specification
- o RA (RAnge) entry
- o HI (HIstory) entry
- o CP (Control Point) entry
- o Information per layer: number of features, number of points and line length
- o Information per feature code: number of features, number of points and line length
- o Totals information

FORMAT

\$ IINFO IFF-file-spec

Command qualifiers

/[NO]ABSOLUTE
/[NO]BRIEF
/EAST
/[NO]HEIGHT
/[NO]LAYER
/[NO]MAP_DESCRIPTOR
/[NO]MH_TYPE
/NORTH
/[NO]OUTPUT
/[NO]PRINTER
/[NO]REVISION_LEVEL
/[NO]SCAN
/SOUTH
/[NO]SYMBOLS
/[NO]TIMINGS
/WEST

Defaults

/NOABSOLUTE
/NOBRIEF
none
/NOHEIGHT
/NOLAYER
/NOMAP_DESCRIPTOR
/NOMH_TYPE
none
/NOOUTPUT
/NOPRINTER
/NOREVISION_LEVEL
/NOSCAN
none
/NOSYMBOLS
/NOTIMINGS
none

PROMPT

_IFF-File: IFF-file-spec

PARAMETER

IFF-file-spec

- specifies the IFF file which is to be processed. Any part of the file name which is not specified will be taken from the default specification 'LSL\$IF:IFF.IFF'.

COMMAND QUALIFIERS

/ABSOLUTE

/NOABSOLUTE (default)

- displays the RA (Range) and CP (Control Point) entries in absolute coordinates. Any window limits given with the /NORTH etc. qualifiers are treated as absolute coordinates.

/BRIEF

/NOBRIEF (default)

- suppresses the display of entries in the IFF file, but displays the revision level number. /BRIEF is allowed only in conjunction with either the /REVISION_LEVEL or /MH_TYPE qualifier.

/EAST = real

- defines the eastern boundary of a window to delimit which feature points are used in the output statistics of point count, heights and line length. If the /ABSOLUTE qualifier is also present, the supplied real number is treated as a coordinate in absolute space.

/HEIGHT

/NOHEIGHT (default)

- displays the maximum and minimum heights from ZS (3-D point) entries, integer heights from type 2 AC (Ancillary Code) entries and real heights from type 3 AC entries.

/LAYER

/NOLAYER (default)

- instructs the program to output feature code data such as line lengths etc. on a by-layer basis. If a layer is in two parts within the IFF file then the values within each are displayed separately in order of occurrence.

/MAP_DESCRIPTOR
/NOMAP_DESCRIPTOR (default)

- displays the contents of the MD (Map Descriptor) entry, in readable character format.

/MH_TYPE
/NOMH_TYPE (default)

- displays the map header (MH) entry type of the IFF file. The /BRIEF qualifier can be specified simultaneously to suppress the display of the entries in the IFF file.

Currently, Laser-Scan support the following map header types:

- 0 map header is unset
- 1 MCE specific map header
- 2 OSGB specific type 2 map header
- 3 OSGB specific type 3 (OSTF) map header
- 4 OSGB specific type 4 (CITF) map header

Type 1 map headers should be edited using MCEHED and type 2 map headers should be edited using OSMHED.

Type 3 and 4 OS map headers are intended for use by Ordnance Survey (Great Britain) only and will be edited using OS developed header editors.

/NORTH = real

- defines the northern boundary of a window to delimit which feature points are used in the output statistics of point count, heights and line length. If the /ABSOLUTE qualifier is also present, the supplied real number is treated as a coordinate in absolute space.

/OUTPUT [= file-spec]
/NOOUTPUT (default)

- output is directed to the specified file-spec in the current directory. Any parts of the file-spec not specified are taken from the default 'IINFO.LIS'. If this qualifier is not given, output will go to 'SYS\$OUTPUT:'.

/PRINTER
/NOPRINTER (default)

- output is directed to a file named IINFO.LIS which is queued automatically for printing and then deleted.

/REVISION_LEVEL
/NOREVISION_LEVEL (default)

- displays the revision level of the IFF file in addition to the entries. The /BRIEF qualifier can be specified simultaneously to suppress the display of the entries in the IFF file.

/SCAN

/NOSCAN (default)

- reports on all layers, giving a description of all feature serial numbers, whether present, missing (i.e. any gaps) or duplicated. Any TC (Transmitted Comment) or CH (Character string) entries are also reported.

/SOUTH = real

- defines the southern boundary of a window to delimit which feature points are used in the output statistics of point count, heights and line length. If the /ABSOLUTE qualifier is also present, the supplied real number is treated as a coordinate in absolute space.

/SYMBOLS

/NOSYMBOLS (default)

- specifies that some of the program statistics are output to global DCL symbols.

/TIMINGS

/NOTIMINGS (default)

- specifies that all the elapsed time and CPU time statistics contained in the IFF HI (HISTORY) entry are summed and the total times reported.

/WEST = real

- defines the western boundary of a window to delimit which feature points are used in the output statistics of point count, heights and line length. If the /ABSOLUTE qualifier is also present, the supplied real number is treated as a coordinate in absolute space.

DESCRIPTION

IINFO does not affect the IFF file being processed. It is purely informational, and outputs data concerning the IFF file specified as requested by the user via the command qualifiers.

If no qualifier is set, then only the default information is output to the screen (or wherever SYS\$OUTPUT points):

- o Full file specification
- o RA (RAnge) entry
- o HI (HIstory) entry
- o CP (Control Point) entry
- o Information per layer: number of features, number of points and line length
- o Information per feature code: number of features, number of points and line length
- o Totals information

The number of points and line-length will include values for both 2-d and 3-d strings. If no 2-d or 3-d strings are encountered, these values will be output as 0. The totals information given is as follows:

- Total number of layers
- Total number of features
- Total number of 2-D points
- Total number of 3-D points
- Total 2-D line-length
- Total 3-D line-length

A window of interest with the IFF file may be created using the qualifiers /NORTH, /SOUTH, /WEST and /EAST. If a window is specified, then any feature points outside the supplied values are ignored in the calculation of point count, line length and height statistics. The feature and layer statistics are unchanged, and any feature totally outside the supplied window will be listed with its attributes set to zero. If /ABSOLUTE is present as well, the supplied window values are treated as absolute coordinates.

If /ABSOLUTE is specified the default information (as above) is output with the RA (RAnge) and CP (Control Point) entries displayed in absolute coordinates.

If /HEIGHT is specified, the maximum and minimum height values found in the IFF file will be displayed. Three maxima and minima are derived:

1. Maximum and minimum heights from integer height (type 2) AC entries.
2. Maximum and minimum heights from real height (type 3) AC entries.
3. Maximum and minimum Z values held in ZS (3-D string) entries.

If /LAYER is specified then, in addition to the default output, feature code information such as number of points and line lengths is totalled per layer.

If /MAP_DESCRIPTOR is specified then the Map Descriptor information is output. Both types of Map Descriptor entry are catered for.

If /NORTH is specified, then any feature points north of the supplied value are ignored in the calculation of point count, line length and height statistics.

If /OUTPUT is specified then all output, except error messages, is directed to the specified file (default IINFO.LIS), instead of to the currently setup SYS\$OUTPUT:.

If /PRINTER is specified then all output, except error messages, is directed to the file IINFO.LIS which is queued automatically for printing, and then deleted.

If /SCAN is specified, the following information is output:

- o A report on all layers, giving a description of all feature serial numbers, whether present, missing (i.e. any gaps) or duplicated.
- o The presence of any TC (Transmitted Comment) or CH (CHaracter string) entries.

In this case, the /TIMINGS, /MAP_DESCRIPTOR and /HEIGHT qualifiers are not appropriate. A message to this effect is displayed if either is specified with /SCAN, and the /TIMINGS, /MAP_DESCRIPTOR or /HEIGHT qualifiers are ignored.

The use of the /SCAN qualifier generates a quick summary of the IFF file, and therefore the output generated is unaffected by any windowing with /NORTH = n etc., and as no coordinates are output, /ABSOLUTE is irrelevant, and the /SYMBOLS qualifier will not create any output DCL symbols.

If /SYMBOLS is specified, the program statistics are output to global DCL symbols. Those created are:

1. Range values (modified by the origin offset if /ABSOLUTE qualifier given).
 - o IINFO_RA_XMIN
 - o IINFO_RA_XMAX
 - o IINFO_RA_YMIN
 - o IINFO_RA_YMAX
2. Window values, as given by /NORTH etc. If no window qualifiers are given, the window defaults to the IFF file Range values (modified by the origin offset if /ABSOLUTE qualifier given).
 - o IINFO_WIN_XMIN
 - o IINFO_WIN_XMAX
 - o IINFO_WIN_YMIN
 - o IINFO_WIN_YMAX
3. The Control Point values.
 - o IINFO_CP_SW1 etc.
4. Maximum and minimum height values from any ZS coordinate entries.
 - o IINFO_MINZ
 - o IINFO_MAXZ
5. Maximum and minimum height values from any AC type 2 (integer) entries.
 - o IINFO_INT_MINZ
 - o IINFO_INT_MAXZ
6. Maximum and minimum height values from any AC type 3 (real) entries.
 - o IINFO_REAL_MINZ
 - o IINFO_REAL_MAXZ
7. The number of layers and features.
 - o IINFO_NUM_LAY

- o IINFO_NUM_FTR
8. The total number of 2d and 3d points.
- o IINFO_NUM_2DPTS
 - o IINFO_NUM_3DPTS
9. The total lengths of 2d and 3d lines.
- o IINFO_LEN_2DLIN
 - o IINFO_LEN_3DLIN

If /TIMINGS is specified, the timing information held in the history record will be summed and output. This gives the total elapsed and CPU times spent on the file so far. If there is no history entry in the IFF file then a message is output to this effect, and no timings information can be displayed.

EXAMPLES

\$ IINFO/HEIGHT INFO_TEST<CR>

IFF file : DUA0:[LSL.IFF]INFO_TEST.IFF;5

Created on : 25-NOV-1987 12:45:39.69

Range Information

Xmin =	0.000	Xmax =	234.567
Ymin =	0.000	Ymax =	948.397

Control Point Information

NW	0.000	1000.000	0.000	1000.000
SW	0.000	0.000	0.000	0.000
SE	1000.000	0.000	1000.000	0.000
NE	1000.000	1000.000	1000.000	1000.000

History Record Information

This IFF file contains no history entry

Height Range Information

Minimum Z height	=	100.000
Maximum Z height	=	983.622
Minimum integer height	=	-10
Maximum integer height	=	1000
Minimum real height	=	-100.582
Maximum real height	=	101.123

Information per Layer

Layer Number	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
1	4	4	0	0.000	0.000
2	9	6	3	0.000	0.000

Information per Feature Code

Feature Code	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
0	13	10	3	0.000	0.000

Totals Information

```

Total number of layers =                2
Total number of features =              13
Total number of 2-D points =            10
Total number of 3-D points =              3
Total 2-D line-length =                 0.000
Total 3-D line-length =                 0.000
ELAPSED:    0 00:00:01.39  CPU: 0:00:00.89  BUFIO: 71  DIRIO: 5  FAULTS: 163
$

```

This example demonstrates the output to be expected when the /HEIGHT qualifier is set. It assumes the logical name LSL\$IF: has been set up to point to the correct directory.

\$ IINFO/MAP_DESCRIPTOR/TIMINGS LESLEY<CR>

```

IFF file      : DUA0:[IFF.IMP.IINFO]LESLEY.IFF;7
Created on   : 14-JUL-1986 17:38:03.09
Updated on   : 14-JUL-1986 17:39:53.69  (Revision number : 3)

```

Range Information

```

Xmin =      -0.643      Xmax =      400.093
Ymin =      -0.145      Ymax =      400.000

```

History Record Information

Date	Time	Username	Program	Function	Elapsed	CPU	STATUS
06-JAN-1986	17:30	TIM	IFROMTEXT	Create	00:00:26	00:00:10	00000001
12-MAR-1986	13:31	CLARKE	ITRANS	Create	00:00:04	00:00:02	00000001
23-APR-1986	10:02	TIM	IED	Update	00:01:00	00:00:07	00000001
23-APR-1986	10:21	TIM	IED	Update	00:00:37	00:00:00	00000001
15-MAY-1986	12:35	RON	IED	Update	00:00:58	00:00:00	00000001
31-MAY-1986	17:38	CLARKE	ITRANS	Create	00:00:03	00:00:01	00000001
19-JUN-1986	09:58	TOM	IED	Update	00:00:09	00:00:00	00000001
01-JUL-1986	15:02	TOM	IED	Update	00:00:07	00:00:00	00000001
14-JUL-1986	17:39	TIM	IED	Update	00:00:18	00:00:00	00000001

Type 2 Map Descriptor Information

Local Origin : 0.000, 90.000

Scale : 1000.000

Projection Code : 101 (UK National Grid)

Spheroid Code : 0 (Airey)

Unit Code : 2 (Metres)

Control Point Information

NW	0.000	400.000	0.000	400.000
SW	0.000	0.000	0.000	0.000
SE	400.000	0.000	400.000	0.000
NE	400.000	400.000	400.000	400.000

Information per Layer

Layer Number	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
1	17	231	156	770.742	2795.237
11	0	0	0	0.000	0.000
44	6	19	0	25.120	0.000
55	6	31	20	61.461	623.010
66	2	164	0	361.901	0.000
88	1	4	0	6.007	0.000
151	0	0	0	0.000	0.000
1234	32	86	1	149.693	0.000
9999	1	1	0	0.000	0.000

Information per Feature Code

Feature Code	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
0	4	17	0	25.120	0.000
5	6	175	0	373.926	0.000
13	4	5	0	0.000	0.000
123	28	309	0	920.435	0.000
150	10	0	120	0.000	2011.770
600	3	0	45	0.000	988.223
717	1	1	0	0.000	0.000
800	3	3	0	0.000	0.000
840	3	0	12	0.000	418.254
987	3	26	0	55.443	0.000

[illegible]

Feature 170 repeated
Feature 170 repeated
Feature 170 repeated
Feature 170 repeated
Feature 170 repeated
Feature 170 repeated
Feature 170 repeated
In Layer 1

Feature 170 repeated

In layer 55
Before feature 280
7865 - CH : "This text came from a CH entry"

In layer 44
Before feature 300
8022 - TC : "This TC is outside a feature"

Before feature 330
8054 - TC : "This is a pointless line of text"

Before feature 190
8115 - CH : "This text came from a CH entry"

Before feature 423
8188 - TC : "This is a particularly nasty TC"

Layer Number : 1

First feature number is : 170
Feature numbers missing : 171 to 189
Feature numbers missing : 191 to 209
Feature numbers missing : 211 to 217
Last feature number is : 230
Layer contains : 16 features

This layer contains : 16 features,
229 2-D points,
156 3-D points.

Layer Number : 11

This layer contains : 0 features,
0 2-D points,
0 3-D points.

Layer Number : 44

First feature number is : 190
Feature numbers missing : 191 to 198
Feature numbers missing : 200 to 299
Feature numbers missing : 301 to 329
Feature numbers missing : 331 to 421
Last feature number is : 423
Layer contains : 6 features

This layer contains : 6 features,
19 2-D points,
0 3-D points.

Layer Number : 55

First feature number is : 260
Feature numbers missing : 261 to 279
Feature numbers missing : 281 to 309
Feature numbers missing : 311 to 420
Feature numbers missing : 422 to 430
Last feature number is : 432
Layer contains : 6 features

This layer contains : 6 features,
31 2-D points,
20 3-D points.

Layer Number : 66

First feature number is : 320
Feature numbers missing : 321 to 419
Last feature number is : 420
Layer contains : 2 features

This layer contains : 2 features,
164 2-D points,
0 3-D points.

Layer Number : 88

First feature number is : 270
Only one feature in layer

This layer contains : 1 feature,
4 2-D points,
0 3-D points.

Layer Number : 151

This layer contains : 0 features,
0 2-D points,

0 3-D points.

Layer Number : 1234

First feature number is : 170
Feature numbers missing : 171 to 174
Last feature number is : 175
Layer contains : 2 features

This layer contains : 2 features,
4 2-D points,
1 3-D point.

Layer Number : 9999

First feature number is : 797
Only one feature in layer

This layer contains : 1 feature,
1 2-D point,
0 3-D points.

Totals Information

Total number of features : 34
Total number of 2-D points : 452
Total number of 3-D points : 177

ELAPSED: 00:00:24.05 CPU: 0:00:04.39 BUFIO: 187 DIRIO: 17 FAULTS: 1953
\$

In this example, the /SCAN and /TIMINGS qualifiers have been specified. The message to say that /TIMINGS information is unavailable with /SCAN has been output, and, of course, no timings information. Note that both 2-d and 3-d points are reported.

\$ IINFO GARBAGE<CR>

%LSLLIB-E-IFFOPEN, error opening LSL\$IF:GARBAGE.IFF;0

%IFF-E-OPEN, error opening IFF file

-RMS-E-FNF, file not found

ELAPSED: 00:00:06.34 CPU: 0:00:00.48 BUFIO: 4 DIRIO: 2 FAULTS: 150

\$

In this example, IINFO has attempted to open the input IFF file and failed to do so. This is either because the file-spec has been mistyped, or the logical name LSL\$IF: has not been set to point to the correct directory.

\$ IINFO/LAYER DUA0:[IAIN]NOTTS

IFF file : LSL\$DATA_ROOT:[LSL.IFF]NOTTS.IFF;3
 Created on : 12-OCT-1987 17:14:53.00
 Updated on : 25-MAR-1988 11:07:16.00 (Revision number : 3)

Range Information

Xmin = -11.300 Xmax = 1000.000
 Ymin = -511.300 Ymax = 511.250

History Record Information

Date	Time	Username	Program	Function	Elapsed	CPU	STATUS
7-OCT-1987	09:27	TOM	OSTF2I	Create	00:00:29	00:00:21	00000001
7-OCT-1987	12:12	TOM	LITES2UIS	Output	00:00:12	00:00:08	00000001
7-OCT-1987	13:51	TOM	LITES2UIS	Output	00:00:31	00:00:16	00000001
7-OCT-1987	16:11	TOM	LITES2UIS	Output	00:00:33	00:00:16	00000001
7-OCT-1987	17:46	TOM	LITES2UIS	Output	00:00:23	00:00:16	00000001
9-OCT-1987	15:47	TOM	IPATCH	Patch	00:00:23	00:00:00	00000001
9-OCT-1987	16:22	TOM	LITES2UIS	Output	00:00:23	00:00:16	00000001
9-OCT-1987	18:08	TOM	LITES2UIS	Output	00:00:23	00:00:16	00000001
12-OCT-1987	17:05	TOM	ISELECT	Create	00:00:14	00:00:08	00000001
12-OCT-1987	17:12	TOM	IRECODE	Create mode	00:00:04	00:00:03	00000001
12-OCT-1987	17:14	TOM	IRECODE	Create mode	00:00:08	00:00:03	00000001

Control Point Information

NW	0.000	500.000	0.000	500.000
SW	0.000	0.000	0.000	0.000
SE	500.000	0.000	500.000	0.000
NE	500.000	500.000	500.000	500.000

Information per Layer

Layer Number	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
0	27	60	0	11500.204	0.000
1	182	2013	0	12067.274	0.000

Information per Feature Code

Feature Code	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
28	77	77	0	0.000	0.000

398	28	62	0	12500.204	0.000
2031	35	364	0	5556.678	0.000
2101	69	1570	0	5510.597	0.000

Totals Information

Total number of layers =	2
Total number of features =	209
Total number of 2-D points =	2073
Total number of 3-D points =	0
Total 2-D line-length =	23567.479
Total 3-D line-length =	0.000

Information per Feature Code/Layer

Layer 0

Feature Code	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
398	27	60	0	11500.204	0.000

Layer 1

Feature Code	No. of Features	No. of 2-D Points	No. of 3-D Points	2-D Line-Length	3-D Line-Length
28	77	77	0	0.000	0.000
398	1	2	0	1000.000	0.000
2031	35	364	0	5556.685	0.000
2101	69	1570	0	5510.522	0.000

ELAPSED: 0 00:00:17.16 CPU: 0:00:07.04 BUFIO: 5 DIRIO: 118 FAULTS: 133
\$

The /LAYER qualifier here means that the feature data is also output on a by layer basis.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user.

ERMAPD, No Map Descriptor data output with /SCAN

Explanation: The operator has specified the /MAP_DESCRIPTOR qualifier with the /SCAN qualifier. These types of information are mutually exclusive, and the /MAP_DESCRIPTOR qualifier will be ignored.

User action: None.

ERTIM, No Timings information output with /SCAN

Explanation: The operator has specified the /TIMINGS qualifier with the /SCAN qualifier. These types of information are mutually exclusive, and the /TIMINGS qualifier will be ignored.

User action: None.

MULCP, multiple Control Point entries encountered

Explanation: More than one Control Point entry has been found in the IFF file. This IFF file has not been processed by IMERGE or LITES2. It retains the multiple section structure of an unprocessed digitising file.

User action: Ensure that this file is not used with any Laser-Scan IFF utility until digitising is complete and the file has been processed using LAPROCESS or IMERGE.

NOHIST, no data in history entry

Explanation: The IFF file contains a history entry, but no records have yet been put in it.

User action: None.

NOLAYERS, no layers defined in the current IFF file

Explanation: No layers were defined in the supplied IFF file.

User action: None.

NOFT, no features defined in the current IFF file

Explanation: No features were defined in the supplied IFF file.

User action: None.

MESSAGES (WARNING)

These messages indicate that an error has occurred which may require some operator action, but which should not stop the processing of the file.

ERRMD2, Map Descriptor not marked as type 2

Explanation: The Map Descriptor entry found is the length of a type 2 map descriptor, but the record is not marked as type 2. The program will not attempt to output any map descriptor data.

User action: None.

ERRCLOUT, error closing output file 'file-spec'

Explanation: On exit, the output file has not been closed properly.

User action: The supplementary message supplied with this message will enable the user to determine the cause of the problem, e.g. insufficient disc space for file allocation. Notify your system manager.

IFFBADCLO, failed to close IFF file 'file-spec'

Explanation: On exit, the IFF file has not been closed properly.

User action: The supplementary message supplied with this message will enable the user to determine the cause of the problem, e.g. insufficient disc space for file allocation. Notify your system manager.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to the command line processing and file manipulation.

BADNF, unexpected end of feature with FSN 'integer' ('integer')

Explanation: A NF entry has occurred in the IFF file before the EF entry relating to the current NF has been found.

User action: Either read the IFF file into LITES2 and exit to a new file, or use ITOTEXT followed by IFROMTEXT which will insert the missing IFF entries without recourse to further user interaction. Re-run IINFO on the corrected file.

BADLAY, unexpected end of layer 'integer'

Explanation: A NO entry has occurred in the IFF file before the EO entry relating to the current NO has been found.

User action: Either read the IFF file into LITES2 and exit to a new file, or use ITOTEXT followed by IFROMTEXT which will insert the missing IFF entries without recourse to further user interaction. Re-run IINFO on the corrected file.

ERROPOUT, failed to open output file 'file-spec'

Explanation: The file for output has not been opened.

User action: The RMS error message output in conjunction with this IINFO message will help to determine the user action required.

MDNOTFOUND, MD entry not found for /ABSOLUTE qualifier

Explanation: There was no Map Descriptor entry in the IFF file for the conversion to absolute coordinates when the /ABSOLUTE qualifier was given.

User action: Amend the IFF file and rerun the program, or rerun without the /ABSOLUTE qualifier to get the information in local coordinates.

UNEXPEOF, unexpected end of IFF file

Explanation: IINFO has detected the end of the IFF file, but has not detected an IFF EJ entry.

User action: Use IMEND/TRUNCATE on the file, which will correctly position the EOF marker and insert an EJ entry at the end of the file. Re-run IINFO on the corrected file.

MESSAGES (OTHER)

In addition to the above messages which are generated by IINFO itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 8

MODULE ILAYFC

MODULE **ILAYFC**

REPLACES **DAMP module LIT**

FUNCTION

ILAYFC changes all feature codes in one or more layers in an IFF file to specified values.

FORMAT

\$ ILAYFC file-spec [file-spec]

Command qualifiers

/CODES=(integer>integer[,...])
/[NO]IN_SITU
/[NO]LOG

Defaults

ILAYFC prompts for codes.
/NOIN_SITU
/NOLOG

PROMPTS

_Input-file: input-file-spec

If /IN_SITU is not specified:

_Output-file: output-file-spec

PARAMETERS

input-file-spec

- Specifies the IFF file from which selections are to be made. Any part of the file-spec which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF;0'.

output-file-spec

- Specifies the IFF file which is to be created. Any part of the file name which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

QUALIFIERS

/CODES=(integer>integer[,...])

- where the first integer of each pair of integers (separated by a '>') is the layer number, the second the feature code which is to replace the feature codes currently in that layer. Failure to specify the /CODES qualifier will result in ILAYFC prompting for input with the prompt:

Layer number, Feature Code ?

This prompt will be repeated for additional arguments until ILAYFC detects a blank line. Processing then begins.

/IN_SITU

/NOIN_SITU

- the default mode of ILAYFC operation is to create a new IFF file to receive the modified data. The /IN_SITU qualifier forces modification to occur in-situ, thus destroying the original data.

/LOG

/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened, and a reassuring message is output each time a new layer is encountered in the IFF file.

DESCRIPTION

If used on an existing IFF file, ILAYFC allows the user to set all of the feature codes in one or more layers to specified values. In most cases ILAYFC will proceed with badly structured input files, changing only the FS entries of features within the specified layers. Thus any output file created may still contain the structural irregularities of the input file. In such cases the user should invoke other IMP utilities in order to cure the problem.

The Feature Code of a feature is carried in the first word of the FS entry of each feature. ILAYFC only changes the first (feature code) word of the FS entry. As the feature code is stored as a word (16 bit integer) its value must lie in the range -32768 to 32767. In practice negative feature codes are not allowed. The other 3 words of the IFF FS entry remain unaltered. Be very careful to ensure that by changing the feature code, conflicts with information carried in the unchanged words of the FS entry are not introduced. Words 2 to 4 of an IFF FS entry are defined in the IFF User Guide.

EXAMPLES

```
$ ILAYFC<CR>
_Input-file: NOTHERE/LOG<CR>
_Output file: FAILURE<CR>
Layer number, Feature code ? 1,23
Layer number, Feature code ? 11,22
Layer number, Feature code ? <CR>
%LSLLIB-E-IFFOPEN, error opening IFF input file LSL$IF:NOTHERE.IFF
%RMS-E-FNF, file not found
  ELAPSED: 00:00:27.87  CPU: 0:00:00.37  BUFIO: 8  DIRIO: 11  FAULTS: 70
$
```

In this example the specified file cannot be opened for input. The accompanying RMS message indicates that the file was not found. The user should check the file specification for errors, check that logical name LSL\$IF: has been assigned to the correct device and directory and check that the file and directory are not protected against read access. \$STATUS is set to SS\$_ABORT.

```
$ ILAYFC
_Input-file: TEST1/LOG/CODES=(1>A23,55>11)<CR>
_Output file: : TEST22<CR>
%ILAYFC-W-BADPARFMT, incorrect format for /CODES parameters 1>A23
Layer number, Feature code ? 1,23HEL
%ILAYFC-W-TWOARGS, two integer arguments required.
Layer number, Feature code ? 1 23
Layer number, Feature code ? 55>11
Layer number, Feature code ? <CR>
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]TEST1.IFF;3 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]TEST22.IFF;1 opened for write
%ILAYFC-I-LAYERCHA, changing all layer 1 feature codes to 23
%ILAYFC-I-LAYERSKP, skipping layer 12
%ILAYFC-I-LAYERCHA, changing all layer 55 feature codes to 11
  ELAPSED: 00:02:22.95  CPU: 0:00:03.31  BUFIO: 13  DIRIO: 18  FAULTS: 110
$
```

This example illustrates several ILAYFC messages, and the procedure for correcting them. First, the /CODES parameters contained a non-decimal number (A23), so ILAYFC prompted for input. In the following case, although the numbers were present, additional characters (HEL) had been input, so ILAYFC prompted for input once again. Notice that in prompt mode a right hand angle bracket, comma or space may be used as the separator between the two integer arguments expected in the input string. Interaction with the user is terminated when a blank line is supplied. \$STATUS is set to SS\$_NORMAL.

\$ **ILAYFC**

_Input-file: **MARSH11/LOG<CR>**

_Output file: **MARSH12<CR>**

Layer number, Feature code ? **1,23**

Layer number, Feature code ? **55 11**

Layer number, Feature code ? **<CR>**

%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]MARSH11.IFF;89 opened for read

%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]MARSH12.IFF;7 opened for write

%ILAYFC-I-LAYERCHA, changing all layer 1 feature codes to 23

%ILAYFC-I-LAYERCHA, changing all layer 55 feature codes to 11

%ILAYFC-I-LAYERSKP, skipping layer 70

%ILAYFC-I-LAYERSKP, skipping layer 71

ELAPSED: 00:02:22.95 CPU: 0:00:03.31 BUFIO: 13 DIRIO: 18 FAULTS: 110

\$

This example illustrates another informational message. The user is informed that two IFF layers, 70 and 71 are not to be altered while all the feature codes in layers 1 and 55 are to be altered to 23 and 11 respectively. \$STATUS is set to SS\$_NORMAL.

\$ **ILAYFC TEST/LOG/CODES=(1>1,88>9,55>17,66>66) DUA0:[DILL.FAILED]TESTOUT<CR>**

%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]TEST.IFF;2 opened for read

%LSLLIB-I-IFFOPENED, DUA0:[DILL.FAILED]TESTOUT.IFF;184 opened for write

%ILAYFC-I-LAYERCHA, changing all layer 1 feature codes to 1

%ILAYFC-I-LAYERCHA, changing all layer 8 feature codes to 9

%ILAYFC-I-LAYERCHA, changing all layer 6 feature codes to 6

%ILAYFC-I-LAYERCHA, changing all layer 15 feature codes to 7

%ILAYFC-I-LAYERSKP, skipping layer 14

%ILAYFC-I-LAYERSKP, skipping layer 11

ELAPSED: 00:00:04.98 CPU: 0:00:02.02 BUFIO: 5 DIRIO: 95 FAULTS: 84

\$

This example shows typical output from a successful run of the program. New layer/feature code allocations have been specified using the /CODES qualifier. \$STATUS is set to SS\$_NORMAL.

\$ **ILAYFC TEST5/IN_SITU/LOG/CODES=(1>9,11>99)<CR>**

%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TRIALS]TEST5.IFF;3 opened for update

%ILAYFC-I-LAYERCHA, changing all layer 2 feature codes to 37

%ILAYFC-I-LAYERCHA, changing all layer 3 feature codes to 349

%ILAYFC-I-LAYERCHA, changing all layer 8 feature codes to 400

ELAPSED: 00:00:04.98 CPU: 0:00:02.02 BUFIO: 5 DIRIO: 95 FAULTS: 84

\$

This example illustrates a successful run using the /IN_SITU option. \$STATUS is set to SS\$_NORMAL.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action from the user.

LAYERSKP, skipping layer 'integer'

Explanation: A new layer has been encountered, it is one in which the feature codes are to be unaltered.

User action: None.

LAYERCHA, changing all layer 'integer' feature codes to 'integer'

Explanation: A new layer has been encountered, it is one in which the feature codes are to be altered to the specified value.

User action: None.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

ARGSREQ, at least one argument pair required.

Explanation: The user has failed to provide any layer/feature code pairs and has pressed carriage return in response to the prompt:

Layer number, Feature Code?

User action: ILAYFC expects at least one layer/feature code pair to work with and will persist in reprompting until one is provided. Use <CTRL/Z> to abort the ILAYFC run if you are unsure of the correct response.

BADCHR, bad character in string - try again

Explanation: A bad character was present in the argument list, remember the format is:

'integer', 'integer'<CR>

User action: ILAYFC will re-prompt for input with:

Layer number, Feature Code?

The user should supply the arguments in the form requested, further arguments can be supplied and the argument list terminated by typing <CR> to the terminal.

BADPARFMT, bad format for /CODES parameters - try again

Explanation: The /CODES parameters have been entered in an incorrect format (see qualifiers section), the user must enter them all again.

User action: None of the codes supplied with the /CODES qualifier will be used, instead ILAYFC will prompt for input with:

Layer number, Feature Code?

The user should supply the arguments in the form requested, further arguments can be supplied and the argument list terminated by typing <CR> to the terminal.

TOOMNYLYRS, only 'integer' layers allowed - continuing

Explanation: The user has supplied too many layer numbers. Processing will proceed for only the specified number of layers.

User Action: Processing of the remaining layers should be completed with a second run of ILAYFC.

TWOARGS, two integer arguments required.

Explanation: Only one integer argument was given when ILAYFC prompted for input with:

Layer number, Feature Code?

The correct format for response is:

<integer> [,] <integer> <CR>

User action: Supply the arguments in the form requested, further arguments can be supplied and the argument list terminated by typing just a <CR> to the prompt.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused ILAYFC to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ERRSSNVIO, successive syntax violations - please read manual

Explanation: After the user has failed to enter the layer number and feature code argument pair in the correct format, in 4 successive attempts, the program exits.

User action: Read this documentation before trying again.

INVFC, feature code not in range 0 32767

Explanation: The user has supplied a feature code which is outside the specified range.

User Action: Specify a feature code that lies within the specified range.

INVLAY, layer number not in range 0 to 32767

Explanation: The user has supplied a layer number which is outside the specified range.

User Action: Specify a layer number that lies within the specified range.

UNEXPEOF, unexpected end of IFF file

Explanation: The specified input IFF file terminated before an IFF EJ entry was encountered.

User action: The valid part of the input IFF file was processed, but the user should use IPATCH to determine the position of the file truncation and then use IMEND to correctly terminate the file. Re-run ILAYFC on the repaired IFF file.

MESSAGES (OTHER)

In addition to the above messages which are generated by ILAYFC itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 9

MODULE IMEND

MODULE **IMEND**

REPLACES DAMP modules MIF and IFT

FUNCTION

IMEND is used to recover IFF files which have been improperly closed as a result of system failure or operator error.

FORMAT

\$ IMEND file-spec

Command qualifiers

/[NO]RANGE
/[NO]TRUNCATE

Defaults

/NORANGE
Truncate file after last complete feature.

PROMPT

_IFF file: IFF-file-spec

PARAMETER

IFF-file-spec

- specifies the IFF file to be mended. Any part of the file-spec which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

COMMAND QUALIFIERS

/RANGE
/NORANGE (default)

- by default IMEND does not update the RA (RAnge) entry after mending the file. The /RANGE qualifier causes IMEND to do a second pass through the IFF file after successful mending. The string (ST,ZS,CB) entries are read and the file (x,y) coordinate range is calculated. The range entry is then updated.

/TRUNCATE

- by default IMEND truncates the IFF file after the last complete map, layer, or feature (EM, EO, or EF entry) found. It then adds EF, EO, EM, and EJ entries as required to correctly terminate the file.

If /NOTRUNCATE is specified (or the file contains no complete map, layer, or feature) then the file is ended after the last complete entry and no extra entries are added.

DESCRIPTION

If an IFF file has been opened for write (or update) and the IFF library is unable to properly close the file then the file may be corrupt. Unclosed files may be caused by system crash, operator error (using <CTRL/Y> for instance), or program failure. To check for this kind of error, the IFF library sets a flag in the forepart of the file when it is opened, and only clears the flag just before the file is closed. If an attempt is made to open a file which has this flag set, the IFF error 'MEND' will occur, and the IFF library will not allow access to the file. Use of IMEND is the only way to regain access to such a file, but IMEND may also be used on files which have been properly closed, but for some reason contain corrupt data, or end part way through a feature.

The particular problems which may occur when a file has been improperly closed (and which IMEND attempts to rectify) are as follows:

- o The forepart will not have been written back to the file. If the file had actually been written to since it was opened, this will mean that the IFF library may be unable to access the data which had been added.
- o The end-of-file pointer may not have been set correctly. An indication of this is that the number of disk blocks used is zero or much less than the number of disk blocks allocated to the file. The DCL 'DIRECTORY/SIZE=ALL' command will supply information about blocks used versus allocated. In this case, the file may not be copied before IMEND is used, as COPY will only copy up to the (incorrect) end of file block.
- o Blocks of data may not have been written to the IFF file. If the file was being written at the end, then the data will probably just be lost, but if the file was being written in the middle, then there is a chance that the file will be corrupted, and that all following data will be lost.

The action taken by IMEND in attempting to repair an IFF file is as follows:

- (i) The file is opened for updating, and its end of file pointer is set to the end of the last allocated block, allowing the filing system to access all the blocks. (This is effectively a SET

FILE/END_OF_FILE operation.)

- (ii) The IFF end of file pointer (LPOSE) is also set to the end of the file, allowing IFFLIB to access all the data.
- (iii) The entries in the file are scanned until either an IFF error occurs, an unexpected entry is found, or the EJ entry is found. The structure is not checked very rigorously. Files may contain maps (delimited by MH/EM), containing layers (delimited by NO/EO), containing features (delimited by NF/EF), containing those entries which are valid within features. The map level is optional. Any EO pointers in NO entries which are incorrect are corrected at this stage.
- (iv) If the file contained no entries at all, then the user is informed that the file is empty.
- (v) If /NOTRUNCATE was used, or the file contained no complete maps, layers, or features, then the IFF end of file is set after the last complete entry.
- (vi) If the file did contain complete maps, layers, or features, and /TRUNCATE (the default) was used, then the file is truncated after the last complete map, layer, or feature, and appropriate EF, EO, EM, and EJ entries are added to terminate the file correctly.
- (vii) If the /RANGE qualifier is specified, IMEND does a second pass through the IFF file after successful mending. The string (ST,ZS,CB) entries are read and the file (x,y) coordinate range is calculated. The range entry is then updated.
- (viii) The file is closed.

After running IMEND the IFF data may subsequently be accessed as usual (but note that some features at the end of the file may have been lost). In particular digitising may be resumed with LASERTRAK using the Laseraid OLD option.

However, note that the recovery of a corrupt IFF file cannot be guaranteed and any file mended with IMEND should be treated with caution.

EXAMPLES

\$ IMEND TST<CR>

```
%LSLLIB-I-IFFOPENED, IFF file DUA3:[BUREAU.TRIALS]TST.IFF;0 opened for mend
%IMEND-I-ENDLAY, end of layer 2
%IMEND-I-ENDLAY, end of layer 32
%IFF-E-IFFERR, IFF error BW    on LUN 1
%IMEND-I-CORRUPT, file corrupt in map 1
%IMEND-I-LSTFTR, last complete feature FSN 9996 (9970) layer 32 map 1
%IMEND-S-IFILREP, IFF file successfully mended
ELAPSED:    0 00:00:02.33  CPU: 0:00:00.71  BUFIO: 7  DIRIO: 37  FAULTS: 88
```

\$

In this example a corrupt, improperly closed IFF file, TST.IFF, has been mended. The diagnostic output shows that the file was opened successfully and that it contained 2 sound layers. The data structure had been corrupted after the feature with FSN (Feature Serial Number) 9666, internal sequence number 9970. The first map was incomplete (no EM entry was found) when the corruption occurred, but no other layers had been commenced.

The IFF error BW is produced by the IFF library, and is one of the IFF errors possible when IMEND reaches the end of sound data and attempts to read corrupted data. It is quite normal for IMEND to produce error messages such as this. IMEND will have truncated the IFF file after feature FSN 9996 (9970) and will have terminated the file with an EO, EM, EJ sequence of IFF entries. All data in the file which was located after feature FSN 9996 (9970) will have been lost. The file should now be available for use, but there can be no guarantee that it is completely sanitary. \$STATUS will be set to SS\$_NORMAL.

\$ IMEND<CR>

_IFF file: GHASTLY.MER/RANGE<CR>

```
%LSLLIB-I-IFFOPENED, IFF file DUA0:[LSL.IFF]GHASTLY.MER;0 opened for mend
%IMEND-I-ENDLAY, end of layer 0
%IMEND-I-ENDLAY, end of layer 1
%IMEND-I-ENDMAP, end of map 1
%IMEND-I-LSTFTR, last complete feature FSN 1956 (1956), layer 1, map 1
%IMEND-S-IFILREP, IFF file successfully mended
%IMEND-I-UPRANGE, updating range entry
%IMEND-I-NEWRANGE, new range entry - XMIN 230.300,  XMAX 450.874, YMIN 0.003,
YMAX 597.839
```

```
ELAPSED:    0 00:00:30.52  CPU: 0:00:00.73  BUFIO: 7  DIRIO: 28  FAULTS: 74
$
```

In this example an improperly terminated IFF file, GHASTLY.MER, has been mended. The diagnostic output shows that the file was opened successfully and that it contained 2 sound layers within one completely sound map. No other maps or layers were commenced (an IFF file corrupt in ... message would have been output). In the absence of any IFF errors, the file was probably intact anyway and correctly terminated with an EJ. The end of file occurred after the feature with FSN (Feature Serial Number) 1962 and internal sequence number 1956. Any entries after the EM for map 1 will have been discarded (but there wouldn't be any expected anyway).

As the /RANGE qualifier was specified, IMEND has performed a second pass through the IFF file to calculate the (x,y) coordinate range. This is patched into the file RA (RAnge) entry. The calculated range is indicated.

The file should now be available for use, but there can be no guarantee that it is completely sanitary. \$STATUS will be set to SS\$_NORMAL.

\$ IMEND PERDU.IFF<CR>

%LSLLIB-E-IFFMODIFY, error opening IFF file DUA3:[BUREAU.TRIALS]PERDU.IFF;0 for update

%IFF-E-IFFERR, IFF error USER on LUN 0

-RMS-E-FNF, file not found

ELAPSED: 0 00:00:18.39 CPU: 0:00:00.34 BUFIO: 4 DIRIO: 9 FAULTS: 42
\$

The file does not exist. Even if the VMS DIRECTORY utility tells you that the file contains zero used blocks, IMEND will still open the file and attempt to mend it. You have either mistyped the file-spec or you have not set logical name LSL\$IF: to point to the correct directory. The error 'USER' is a consequence of the routine used by IMEND to fix the end of file pointers failing to find the file - normal programs would give error 'OPEN' under these circumstances.

\$ IMEND RUINED.IFF<CR>

%LSLLIB-E-IFFMODIFY, error opening IFF file DUA3:[BUREAU.TRIALS]RUINED.IFF;0 for update

%IFF-E-IFFERR, IFF error READ on LUN 0

-RMS-E-EOF, end of file detected

ELAPSED: 0 00:00:01.39 CPU: 0:00:00.12 BUFIO: 4 DIRIO: 9 FAULTS: 11
\$

The RMS (Record Management Services) error message "end of file detected" indicates that the file is completely empty. It does not even contain an IFF "forepart" in the first disk block. There is no hope for the recovery of this file. All data contained in the file has been lost.

\$ IMEND SADNESS.IFF<CR>

%LSLLIB-I-IFFOPNED, IFF file DUA0:[LSL.IFF]SADNESS.IFF; opened for mend

%IMEND-I-IFFEMPTY, IFF file is empty

ELAPSED: 0 00:00:01.39 CPU: 0:00:00.12 BUFIO: 4 DIRIO: 9 FAULTS: 11
\$

Compare this example with the one above. This file did contain a forepart, and was successfully opened, but it did not contain any entries. As in the previous example there is no hope for the recovery of this file.

\$ **IMEND DEMO/NOTRUNCATE<CR>**

%LSLLIB-I-IFFOPENED, IFF file DUA3:[BUREAU.TRIALS]DEMO.IFF;0 opened for mend

%IMEND-I-ENALAY, end of layer 1

%IMEND-I-CORRUPT, file corrupt in feature with FSN 7 (7), layer 2, map 1

%IMEND-I-LSTFTR, last complete feature FSN 1 (1), Layer 2, Map 1

%IMEND-I-TERM, file has been terminated after the last complete entry

%IMEND-I-IFILREP, IFF file successfully mended

ELAPSED: 0 00:00:05.32 CPU: 0:00:00.54 BUFIO: 8 DIRIO: 17 FAULTS: 104
\$

In this example the /NOTRUNCATE option has been used to preserve all entries in the file. After scanning layer 1, IMEND has detected an error somewhere in the feature with FSN (Feature Serial Number) 7 in layer 2. The last complete feature was feature 1, also in layer 2. Because /NOTRUNCATE has been used, the file will still contain the incomplete feature 7. After examining the file, IMEND may be used again without the /NOTRUNCATE option to remove the part feature if required. \$STATUS is set to SS\$_NORMAL.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user.

CORRUPT, file corrupt in feature WITH FSN 'integer' ('integer'), layer 'integer', map 'integer'

Explanation: Corruption of the IFF file structure has been detected within a map, layer, or feature. The file will be terminated at the last complete map, layer, or feature (or entry if /NOTTRUNCATE was used). If this message does not occur, then it is likely that no data has been lost.

User action: None.

ENDMAP, end of map 'integer'
ENDLAY, end of layer 'integer'

Explanation: The end of the given map or layer has been reached. It is safe to assume that the data before this point is safe (but note that the same layer may occur several times if the layer has become fragmented).

User action: None.

FIXEOP, EO pointer amended in layer 'integer'

Explanation: The pointer to the end of this layer contained within the NO entry has been found to be incorrect. It has been corrected.

User action: None. But if this happens repeatedly, then the program responsible may have a bug which requires fixing.

IFILREP, IFF file successfully mended

Explanation: All is well. IMEND has recovered at least some, if not all of the IFF file.

User action: The IFF file may be used with caution.

IFFEMPTY, IFF file is empty

Explanation: There is no hope for the recovery of this file, it contains no entries at all.

User action: None.

LSTFTR, last complete feature FSN 'integer' ('integer'), layer 'integer', map 'integer'

Explanation: This is the last complete feature in the mended file. If this message is not output, then there were no complete features.

User action: None.

NEWRANGE, new range entry - XMIN 'real', XMAX 'real', YMIN 'real', YMAX 'real'

Explanation: IMEND indicates the new range values calculated from the string (ST,ZS,CB) entries present in the file after mending.

User action: Check that the calculated range is what you expected to have in the file.

NOSTS, file contains no ST entries - range entry will not be changed

Explanation: After mending, the file contains no ST entries. The RA (Range) entry will not be changed.

User action: None.

TERM, file has been terminated after last complete entry

Explanation: Either /NOTRUNCATE was used, or the file contained no complete maps, layers, or features. The file now ends after the last complete entry. No attempt has been made to add EF, EO, EM, or EJ entries.

User action: None.

UPRANGE, updating range entry

Explanation: You have specified the /RANGE qualifier. This causes IMEND to do a second pass through the IFF file after successful mending. The string (ST,ZS,CB) entries are read and the file (x,y) coordinate range is calculated. The range entry is then updated.

User action: None.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation.

NORANGE, file has no range entry

Explanation: IMEND has failed to find the RA (RAnge) entry at the start of the IFF file. It is therefore unable to patch in the calculated (x,y) coordinate range.

User action: Without a range entry, the file is useless. Use ITOTEXT to convert the file into a textual representation. Use a text editor (like VMS EDIT or TPU) to add a range entry. Use IFROMTEXT to convert the text back to an IFF file. Re-run IMEND/RANGE to correctly update the range entry that you created.

MESSAGES (OTHER)

In addition to the above messages which are generated by IMEND itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. It is quite normal for IMEND to produce an IFF error message when corrupt data is encountered - it will terminate the file at this point. If the cause of an error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 10

MODULE IMERGE

MODULE **IMERGE**

REPLACES DAMP utility MER

FUNCTION

IMERGE is an IFF file merge utility. It is used in four circumstances:

- o After a map has been digitised on LASERTRAK and processed by LAPPROCESS, IMERGE should be run to consolidate all the layer parts from different sessions, before further processing.
- o To concatenate two or more IFF files into a single map, possibly offsetting the string coordinates as required using information from the MD entries.
- o To select particular layers from the input files. The layers are selected using the /LAYER qualifier.
- o To split input IFF files into two separate output files on the basis of IFF layer. The output IFF files may use the layer numbers used in the input files or a new single layer may be specified for each output file. The two IFF output files which result from specifying the /SPLIT qualifier are referred to below as the "selections-output-file" and the "rest output file". The selections-output-file contains all the features which lay within IFF layers selected from the input IFF files and the rest-output-file contains the rest of the layers, (those which were not selected). It is only possible to split input IFF files into two output file if layer selections have been specified with /LAYER (see below).

NOTE

By default the CP (Control Point) entry for the output files is taken from the first input file.

FORMAT

\$ IMERGE input-file-spec[,...] output-file-spec

Command qualifiers

/CP_EXPANSION
/[NO]DEBUG
/LAYER=integer[,...]
/[NO]LOG
/OUTPUT [=list-file-spec]

Defaults

Take CP entry from first input file.
/NODEBUG (See text).
Use all layers.
/NOLOG
Do not create a listing file.

/SPLIT [=IFF-file-spec]	Create only one output file.
/[NO]UNIQUE_FSN	/NOUNIQUE_FSN

Input file command qualifiers	Defaults
--------------------------------------	-----------------

/LAYER=integer[,...]	Use all layers.
----------------------	-----------------

Output file command qualifiers	Defaults
---------------------------------------	-----------------

/CP_EXPANSION	Take CP entry from first input file.
/REST_LAYER=integer	Create all the input file layers.
/SINGLE_LAYER=integer	Create all the input file layers.

Example formats:

```
$ IMERGE/LOG/SPLIT=restfile/REST_LAYER=integer/OUTPUT=listfile -<CR>
_$ infile1/LAYER=(integer,...),infile2/LAYER=(integer,...), -<CR>
_$ infile'n'/LAYER=(integer,...) outfile/SINGLE_LAYER=integer<CR>
```

Where specific layer number selections are being made for each individual infile.

Or, alternatively:

```
$ IMERGE/LOG/SPLIT=restfile/REST_LAYER=int/LAYER=(int,...) -<CR>
_$ infile1,infile2,infile3,infile4 outfile/SINGLE_LAYER=int<CR>
```

Where the layer selections made with /LAYER are applied globally to all the input files.

Or, alternatively:

```
$ IMERGE<CR>
_From: /LOG/SPLIT=restfile/REST_LAYER=int -<CR>
_$ infile1/layer=(int,...), -<CR>
_$ infile2/LAYER=(int,...), -<CR>
_$ infile3/LAYER=(int,...), -<CR>
_$ infile4/LAYER=(int,...), -<CR>
_$ infile5/LAYER=(int,...), -<CR>
_$ infile6/LAYER=(int,...), -<CR>
_$ infile7/LAYER=(int,...)<CR>
_To: outfile/SINGLE_LAYER<CR>
```

Where IMERGE is put into prompt mode and continuation lines are used. Up to seven input files may be specified.

PROMPT

```
_From:    input-file-specs
_To:      output-file-spec
```

PARAMETERS**input-file-specs**

- IMERGE can process up to 7 input files separated by commas. The first input file is parsed against the default file specification 'LSL\$IF:IFF.IFF;0'. For any subsequent input file specifications IMERGE parses using the previously supplied file specification as the default for missing specification components. The default thus "rolls along" the command line from left to right, in the same manner as in VMS utilities such as COPY.

output-file-spec

- The output file to contain the selected data. The output file specification is parsed using the last input file as the default for missing specification components.

COMMAND QUALIFIERS**/CP_EXPANSION**

- by default the CP (Control Point) entry for the output file is taken from the first input file on the command line. The /CP_EXPANSION qualifier allows the user to expand the output file IFF CP entry values to take account of CP entries in all the input files.

Great care must be taken in the use of the /CP_EXPANSION qualifier as it is only appropriate for input files which fit adjacently together to produce a rectangular output map. If IMERGE is unable to produce rectangular control points with the /CP_EXPANSION qualifier a warning is issued and the CP entry of the first input file is used instead.

/DEBUG
/NODEBUG

- reserved for Laser-Scan use. The /DEBUG qualifier invokes special IMERGE code to force output of debug messages.

/LAYER=integer[,...]

- by default IMERGE will copy to the output file all the IFF layers found in each input file. /LAYER enables the user to specify which IFF layers are to be selected for mergeing.

/LAYER is a positionally dependent qualifier. Its effect varies depending upon its placement within the command line:

- o If /LAYER is placed after the IMERGE command a 'blanket' input file layer selection is imposed. This will apply to all the input files. If desired, further file-specific layer selections may then be made by placing /LAYER after the particular files from which the specific selections are to be made.
- o If /LAYER is placed after an individual input file specification, it enables the user to select for transfer to the selections-output-file only the IFF layers with the numbers specified from that particular input file. For selections which vary between the different input files the /LAYER qualifier should be repeated after each input file specification from which only specific layers are required. If an individual file-spec does not have an associated /LAYER qualifier then all the layers in the file will be used. If, however, a /LAYER qualifier was placed after the IMERGE command, then the 'blanket' layer selections will be applied.
- o /LAYER cannot be specified after the output file specification.

/LAYER will accept single, multiple or ranges of layer arguments. The maximum number of layers which may be specified is 1024.

/LOG

- this will result in supplementary messages being output. Supplementary messages are generated when a file is successfully opened, and a reassuring message output each time a new IFF layer is created in the output IFF files.

/OUTPUT [=file-spec]

- IMERGE is a complex program, combining IFF files which may have different control points and origin offsets. As many as 7 input files may be combined into a single output file. It is not possible to keep a record of all the input file characteristics in the output IFF file HI (HISTORY) entry.

In a production environment it is often necessary to keep records of program run details for future reference. The /OUTPUT qualifier enables the user to write to the specified file details of the input and output files, together with a record of the command line specified. The default /OUTPUT file specification is SYS\$DISK:IMERGE.LIS;0.

/REST_LAYER=integer

- by default IMERGE will preserve in the rest-output-file all the layer numbers used in the input files. The /REST_LAYER qualifier allows the user to force all the unselected merged data into the rest-output-file layer specified. The /REST_LAYER qualifier is only valid for use with the /SPLIT and /LAYER qualifiers. The layer number specified must lie in the range 0 to 32767.

/SINGLE_LAYER=integer

- by default IMERGE will preserve in the selections output file all the layer numbers used in the input files. The /SINGLE_LAYER qualifier allows the user to force all the merged data sent to the selections output file into the layer specified. The layer number specified must lie in the range 0 to 32767.

/SPLIT [=rest-IFF-file]

- split input IFF files into two separate output files on the basis of IFF layer. This qualifier can only be used in conjunction with the /LAYER qualifier. The two IFF output files which result from specifying the /SPLIT qualifier are referred to as the "selections-output-file" and the "rest-output-file". The selections-output-file contains all the features which were within IFF layers selected from the input IFF files and the rest-output-file contains the rest of the layers i.e. those which were not selected.

The optional argument to /SPLIT is the file-spec of the rest-output-file. If no argument is given, then the default LSL\$IF:IMERGESPLIT.IFF;0 is used. The rest output IFF file may use the layer numbers used in the input files or a new single layer may be specified by using the /REST_LAYER qualifier.

/UNIQUE_FSN

/NOUNIQUE_FSN

- preserves the FSN's of the first file in the list of input files, and, by using the maximum FSN of the first file, automatically rennumbers the FSN's in subsequent input files to create unique FSN's in the output file. If this qualifier is selected, the FSN's in the first input file are assumed to be unique and in an ascending order. The default is /NOUNIQUE_FSN which retains the FSN's of each of the input files, thus allowing the possibility of duplicate FSN's in the output file.

DESCRIPTION

General

Although IMERGE is the IMP IFF file merge utility it offers a range of file processing functions within a production environment. It is used in four circumstances:

1. IFF File mergeing

In its most obvious role IMERGE is used to concatenate two or more IFF files into a single map, possibly offsetting the string coordinates as required using information from the input file IFF MD (Map Descriptor) entries. Up to 7 input files may be specified for a single run of IMERGE.

The input files are merged together on the basis of IFF layer. Each input file is opened in turn and its contents scanned for IFF NO (start of layer) entries. The address of each NO within each input file is tabulated in a layer table. After scanning, the current input file is closed and the next input file opened. When all the input files have been opened, scanned and closed again IMERGE begins the process of file concatenation.

The table of scanned layers is sorted on the basis of layer number and the lowest layer number found in the input files is output first; each input file containing a fragment of the current layer number is opened and its layer fragment copied to the output file.

If examination of the layer table shows that further fragments of the current layer number lie in the currently open input file, then these too are found and copied to the output file.

The current input file is closed and the next input file containing layer fragments which match the current layer number is opened and the process repeated.

When all layer fragments of the current layer number have been extracted from all the input files and copied to the output file, the layer table is read again for the next highest layer number. The process of input file opening and layer fragment extraction is repeated until all layer fragments in the input files have been concatenated into complete layers in the output file.

The output file now has only one map containing all the layers found in all the input files. The layers are ordered on the basis of ascending layer number, all input layer fragments of the same layer number are concatenated into a single layer. The output file IFF header entries are set according to the following rules:

- (i) The RA entry (RAnge) is set to reflect the minima and maxima of the range entries of all the input files. If any input file lacks a range entry (which means that the whole file is probably suspect and IMERGE may find other IFF errors) IMERGE will attempt to calculate the range for the output file by determining the maxima and minima from all coordinates in all the input files. IMERGE will take account of any local offset variations between the origin offsets of the various input files and the output file.
- (ii) The MH (Map Header) is taken from the first input file. If the MH entry missing from the first input file, a warning is issued and the MH is taken from the second (or subsequent) input file. In the event of input from a single file which lacks a MH or multiple files when none of the files contains a MH an output file MH is constructed with all the data fields set to zero.
- (iii) The rule (ii) also applies for the MD, NS, CC and CP entries with the following specific notes.

- o If no MD (Map Descriptor) is found in any input file a type 2 MD will be generated. The origin offset will be unset. Attention is drawn to the notes on use of MD for data offsetting at the end of the DESCRIPTION section.
- o If no NS (New Section) entry is found in any input file a NS entry having the text "IMERGE - NS missing from input files" is created. This entry has a length of 40 bytes and may be modified by the user using IPATCH/WRITE.
- o If no CC (Cubic Coefficients) entry is found in any input file a unit CC entry is created. If no CP (Control Point) entry is found in any input file a CP entry is created with the elements set to 0.0. More appropriate values may be substituted by the user using IPATCH/WRITE.

2. IFF file NS (New Section) concatenation

Because of its ability to concatenate layer fragments, both between and within IFF files, IMERGE has a special place in the LAMPS production environment.

Due to the nature of the digitising process, either using LASERAID or a manual table system, IFF files will usually contain data from several digitising sessions. Each time a digitising session is started the IFF file is extended with a new header sequence of NS (New Section), CC (Cubic Coefficients) and CP (Control Points) entries. A new layer is then opened and the additional digitising added.

It is essential that the multiple sections of an IFF file and the fragmentary layers be concatenated before further LAMPS processing may be conducted. LAPROCESS must first be used if the IFF file was digitised using LASERAID. Then either IMERGE or LITES2 may be used to consolidate all the layer parts from different sessions, before further IMP processing. If IMERGE is used the IFF file should be merged to itself, for example:

```
$ IMERGE/LOG SHEET274NE_CONTOURS.IFF SHEET274NE_CONTOURS.IFF
```

The consolidated file will be given the next highest version number to that of the input file.

NOTE

It is essential that IMERGE is NEVER used on files that were created using LASERAID and that have not been processed using LAPROCESS.

3. Layer selection

IMERGE can be used to merge out of the input files only selected layers. The layers are selected using the /LAYER qualifier. The effect of the /LAYER qualifier is dependent upon its position within the command line. See the QUALIFIERS section for a detailed description of /LAYER.

4. File bifurcation

IMERGE can be used to split input IFF files into two separate output files on the basis of IFF layer.

The output IFF files may use the layer numbers used in the input files or a new single layer may be specified for each output file.

The two IFF output files which result from specifying the /SPLIT qualifier are referred to as the "selections output file" and the "rest-output-file".

The selections-output-file contains all the features which lay within IFF layers selected from the input IFF files and the rest-output-file contains the rest of the layers, i.e. those which were not selected. It is only possible to split an input IFF files into two output files if layer selections have been specified with /LAYER (q.v.).

IMERGE and the IFF MD (Map Descriptor) entry

With the release of the IMP package several enhancements to the IFF library were announced. One of the most powerful of these is the type 2 MD (Map Descriptor) entry.

One of the features of the type 2 MD is the ability to hold an origin offset. This origin offset is a coordinate pair which when added to string coordinate values, will give the coordinates their true position.

One advantage of having an origin offset is the ability to hold and manipulate large coordinate values within the IFF file without loss of precision. The origin offset should be chosen to be largest number which can be subtracted from all the coordinates in the file without making any of them negative. Typically, the origin offset will be the coordinate values of the south west corner of a map sheet.

The IFF library has supported customer specific origin offsetting facilities for a long time within special MH (Map Header) entries. IMERGE does not recognise the offsets in such headers. The type 2 MD origin offset allows IMERGE to provide all the offsetting functionality of previous customer specific utilities as relative offsets between map sheets may be performed now using IMERGE.

IFF files having type 1 MD entries without origin offsets may be easily converted to type 2 MD standard by using ITRANS/DESCRIPTOR. It is of course essential that the origin offset be set up correctly.

IMERGE applies the origin offset in the following manner.

- (i) At the stage when each input file is being scanned for layer fragments the MD entry is located in each file.
- (ii) If the MD found is type 2, the origin offset is read and stored.
- (iii) The origin offset of the first input file is considered to be the offset for the output file.
- (iv) If the origin offsets are found to be different between the first and the second and subsequent input files local offsets are calculated to convert all the IFF string coordinates into common space before copying to the output file. IMERGE does not use the minima and maxima of the input file range entries to calculate the output file RA (RAnge) entry. Instead it has to calculate the IFF RANGE entry based on the input string coordinates plus the local offset data.

If input files with a mixture of MD types are supplied to IMERGE the output file will be worthless if those with type 2 MDs have set origin offset fields while those with type 1 MDs have not taken account of the origin offset. The warning message:

NOV2MD, IFF map descriptor in 'file-spec' is not version 2

should always be heeded and the user should ensure that the coordinates in the specified file are in the correct coordinate space.

IMERGE preserves compatibility with IFF files having type 1 Map Descriptors in the following manner:

- o If none of the input files contain a type 2 MD entry then the output file is created with a type 1 MD copied from the first input file.
- o If IMERGE is supplied with files having a mixture of type 1 and type 2 MD entries (a very dangerous thing to do) the IMERGE creates a MD entry of the type found in the first input file.
- o If all the input files have type 2 MD entries, a type 2 MD is created in the output file, and its contents are copied from the first input file.

In addition to providing an origin offset facility, IMERGE also conducts checks on the map projection data held in a type 2 MD.

The map projection data held in the MD of the first input file is considered to be "truth". The second and subsequent input file MD entries are checked for consistency of map projection constants. Spheroid, measurement unit, projection and projection parameters are compared with those of the first input file. A warning message is issued if any conflict is found. For instance it is clearly useless to merge together files having coordinates on different projections.

IMERGE and the IFF HI (HISTORY) entry

Because IMERGE can accept up to 7 input files it is impossible to copy the IFF HI (HISTORY) entries from all the input files to the output files.

If any one of the input files contains a HI entry, a new HI is created in the output files. The selections-output-file has the words "Merge: select" written into the function field of the HI. The optional rest-output-file has the words "Merge: rest" written into the function field of the HI.

If no input file has a HI entry then no HI entry is created in the output file.

To enable the user to keep track of the history of the input files after merging, IMERGE offers the user the /OUTPUT qualifier. The file created by /OUTPUT contains a listing of the HI entries of all the input files (together with listings of their CP (Control Point) entries and user process information). This file may be stored (or printed) as a record of the merge operation and the status of the input files before they became unified into a single new output file.

EXAMPLES

```
$ IMERGE CONT_NE,CONT_NW,CONT_SE,CONT_SW CONTOURS<CR>
```

```
ELAPSED: 00:00:54.98 CPU: 0:00:42.02 BUFIO: 85 DIRIO: 595 FAULTS: 284
$
```

This example shows a successful run of IMERGE in default mode, i.e. mergeing all data from all the input files (LSL\$IF:CONTOURS_NE.IFF;0, LSL\$IF:CONT_NE.IFF;0, LSL\$IF:CONT_NE.IFF;0 and LSL\$IF:CONT_NE.IFF;0) and creating a single output file (LSL\$IF:CONTOURS.IFF;0). All the input files have valid type 2 MD (Map Descriptor) entries and if the origin offset was set in the IFF MD entries the data have been offset by the appropriate amount in the output file. No listing file is created. \$STATUS is set to SS\$_NORMAL.

```
$  
$ IMERGE DUA0:[IFF.USER.EXPERIMENTS]F204.IFF F204.IFF<CR>  
ELAPSED: 00:00:06.85 CPU: 0:00:01.50 BUFIO: 18 DIRIO: 33 FAULTS: 195  
$
```

This example shows a successful run of IMERGE being used to merge a file to itself. This will serve to consolidate all layer fragments found in the file and will ensure that the file has only one sequence of IFF file header entries. The file had in fact been digitised using LASERAID and had been processed using LAPROCESS before this run of IMERGE. \$STATUS is set to SS\$_NORMAL.

```
$  
$ IMERGE/LAYER=(10,14) SW.MER/LAYER=(7,9), -<CR>  
_$ DUA0:[IFF.USER.EXPERIMENTS]SE.IFF, -<CR>  
_$ NE/LAYER=1,[IFF.TEST]NW<CR>  
_To: WHOLEMAP<CR>  
ELAPSED: 00:00:56.85 CPU: 0:00:19.50 BUFIO: 18 DIRIO: 433 FAULTS: 765  
$
```

This complex example shows IMERGE being used to select only specified layers from the input files
LSL\$IF:SW.MER;0, DUA0:[IFF.USER.EXPERIMENTS]SE.IFF;0,
DUA0:[IFF.USER.EXPERIMENTS]NE.IFF;0 and DUA0:[IFF.TEST]NW.IFF;0. The missing file-spec components of the first input file specification have been derived by parsing against the Laser-Scan default IFF file specification LSL\$IF:IFF.IFF;0. However, the user has given the second input file a full file specification, including device and directory. IMERGE always parses command line file specifications using the previously supplied file specification as the default for missing specification components. The default thus "rolls along" the command line from left to right, in the same manner as in VMS utilities such as COPY. The output file in this example has thus become DUA0:[IFF.TEST]WHOLEMAP.IFF;0.

The /LAYER qualifiers have the following effect.

The first /LAYER qualifier lies after the command verb but before the first command parameter. It is thus considered to have a global interpretation, i.e. the specified layers, 10 and 14, will be extracted from ALL the input files.

The next /LAYER qualifier is placed after the first command parameter; LSL\$IF:SW.MER;0. The placement of this qualifier will ensure that layers 7 and 9 will be extracted from this file in addition to global layers 10 and 14.

As there is no /LAYER qualifier after the second input file only the globally selected layers 10 and 14 will be extracted from it.

The third input file (DUA0:[IFF.USER.EXPERIMENTS]NE.IFF;0) has a /LAYER qualified placed after it. This qualifier ensures that any data lying in layer 1 will be extracted, in addition to global layer selections 10 and 14.

The last input file (DUA0:[IFF.TEST]NW.IFF;0) has no /LAYER qualifier placed after it and so only global layers 10 and 14 will be extracted.

The net result of the global and the two local /LAYER qualifiers is that the output file will contain only IFF layers 1, 7, 9, 10 and 14 from some of the various input files.

```
$
$ IMERGE SW,SE/OUTPUT=MAP7 DUA0:[PROJECT.QUARRIES]SOUTH<CR>
%IMERGE-W-NOV2MD, IFF map descriptor in DUA0:[LSL.IFF]SW.IFF;1 is not version 2
%IMERGE-W-UNSETMD, map descriptor in DUA0:[LSL.IFF]SE.IFF;1 is unset
  ELAPSED: 00:00:24.89  CPU: 0:00:02.99  BUFIO: 28  DIRIO: 80  FAULTS: 176
$
```

This example illustrates the use of the /OUTPUT qualifier to specify a file to contain listed details of the IFF HI (HISTORY) and CP (Control Point) entries, together with origin offset information for the input files. Details of the user process and command line qualifiers used are also recorded. The /OUTPUT listing file is parsed against the default specification SYS\$DISK:IMERGE.LIS, resulting in the file specification SYS\$DISK:MAP7.LIS.

IMERGE has detected two errors in the input files and has issued warning messages. The first input file, DUA0:[LSL.IFF]SW.IFF;1, has a type 1 IFF MD (Map Descriptor) entry. No origin offset information is available for this file. The second file, DUA0:[LSL.IFF]SE.IFF;1, has a type 2 MD entry but the entry has not been set by using either ITRANS/DESCRIPTOR or ISTART. Again, no origin offset information is available for this file. \$STATUS is set to SS\$_NORMAL.

```
$
$ IMERGE/LOG/SPLIT=UNUSED.IFF/REST_LAYER=12<CR>
_From: CHIPPINGS/LAYER=1,BALLAST/LAYER=(1,2,8)<CR>
_To: QUARRY_PLAN_3<CR>
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]CHIPPINGS.IFF;2 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]BALLAST.IFF;6 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]QUARRY_PLAN_3.IFF;12 opened for write
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]UNUSED.IFF;1 opened for write
%IMERGE-I-WRSELLAY, writing selected layers to
DUA0:[LSL.IFF]QUARRY_PLAN_3.IFF;12
Writing layer 1
Writing layer 8
%IMERGE-I-WRESTLAY, writing remaining layers to DUA0:[LSL.IFF]UNUSED.IFF;1
Writing layer 12
  ELAPSED: 00:00:07.49  CPU: 0:00:04.46  BUFIO: 39  DIRIO: 117  FAULTS: 118
$
$
```

This example shows the use of the /SPLIT and the /REST_LAYER qualifiers. Because /SPLIT has been specified, those IFF layers which have been selected using the /LAYER qualifier will be written to the selections-output-file DUA0:[LSL.IFF]QUARRY_PLAN_3.IFF;12. Those layers not selected from the input files, i.e. the rest of the input layers, will be written to the /SPLIT rest output file DUA0:[LSL.IFF]UNUSED.IFF;0. The /REST_LAYER qualifer has been used to specify that all the layers written to the rest-output-file shall be consolidated into a single output layer, layer 12.

The /LOG qualifier has been specified. It causes messages to be output when files are opened and shows the data being split between the two IFF output files. Notice that layer 2 has not been written to the selections-output-file (DUA0:[LSL.IFF]QUARRY_PLAN_3.IFF;12) despite the /LAYER=(1,2,8) qualifier arguments applied to DUA0:[LSL.IFF]BALLAST.IFF;6. This means that DUA0:[LSL.IFF]BALLAST.IFF;6 contains no layer 2.

```
$
$ IMERGE/UNIQUE_FSN<CR>
_From: DUPLICATES1,DUPLICATES2<CR>
_To: UNIQUES<CR>
%IMERGE-W-UNSORTFSN, FSN's in first IFF file are not in ascending order at
Layer 0, FSN 1000
ELAPSED: 00:00:20.09 CPU: 0:00:01.99 BUFIO: 28 DIRIO: 80 FAULTS: 176
$
```

This example illustrates the use of the /UNIQUE_FSN qualifier to force the FSN's of the output file to be unique. Users must note that the FSN's in the first input file should be unique and in an ascending order. However, the program does check that FSN's are in ascending order, and if not a warning warning message is output, as shown in this example.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). Most will only appear if /LOG has been specified on the command line, unless they are used to provide supplementary information after an error has occurred.

CONFLPROJ, conflicting map projection in 'file-spec'

Explanation: If mergeing from more than one input file IMERGE takes pains to check that the type 2 IFF Map Descriptors are all set to common values for projection, measurement units, spheroid etc. The first input file is considered to have the definitive Map Descriptor information. This message indicates that the input files contain data on different projections.

User action: Use ITRANS to transform all the input files into a common projection, measurement system etc.

CONFLSPHERE, conflicting spheroid in 'file-spec'

Explanation: If mergeing from more than one input file IMERGE takes pains to check that the type 2 IFF Map Descriptors are all set to common values for projection, measurement units, spheroid etc. The first input file is considered to have the definitive Map Descriptor information. This message indicates that the input files contain data on different spheroids.

User action: Use ITRANS to transform all the input files into a common projection, measurement system etc.

CONFLUNIT, conflicting map units in 'file-spec'

Explanation: If mergeing from more than one input file IMERGE takes pains to check that the type 2 IFF Map Descriptors are all set to common values for projection, measurement units, spheroid etc. The first input file is considered to have the definitive Map Descriptor information. This message indicates that the input files contain data in different measurement systems.

User action: Use ITRANS to transform all the input files into a common projection, measurement system etc.

CONFLMDPAR, conflicting projection parameters in 'file-spec'

Explanation: If mergeing from more than one input file IMERGE takes pains to check that the type 2 IFF Map Descriptors are all set to common values for projection, measurement units, spheroid etc. The first input file is considered to have the definitive Map Descriptor information. This message indicates that the input files contain data relating to different projection parameters.

User action: Use ITRANS to transform all the input files into a common projection, measurement system etc.

DEF, default values for IFF header entries will be used in output file
'file-spec'

Explanation: Not all IFF file header entries were present in the input IFF files. This is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode. IMERGE will supply default values for any of the following entries which were missing:

- MD, (Map Descriptor),
- MH, (Map Header),
- NS, (New Section)
- CC, (Cubic Coefficients), and,
- CP, (Control Points).

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files to ensure that the default IFF header entry values are acceptable.

FIRSTCP, defaulting to CP entry of first input file

Explanation: The /CP_EXPANSION qualifier has been specified for use on input files which will not form a rectangular merged map area. IMERGE will use the Control Point entry of the first input file as the pattern for the output files CP entry.

User action: Consider why the /CP_EXPANSION qualifier was specified. Modify the CP entry in the IFF output files using IPATCH/WRITE.

NOIFFHI, file has no IFF HI entry

Explanation: The input IFF file that is currently open is of the old pre-IMP pattern and probably also lacks a type 2 map descriptor (for which a separate warning will be issued).

User action: None.

USE, using IFF 'code' entry from 'file-spec'

Explanation: The first input file lacked the specified IFF header entry. IMERGE attempts to continue execution by taking the missing entry from an input file which was specified later in the command line. The lack of an IFF header entry is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode.

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files to ensure that the substitute IFF header entry values are acceptable.

WRSELLAY, writing selected layers to 'file-spec'

Explanation: This is output as an indication of what IMERGE is currently doing. IMERGE has scanned all the input files for layer fragments and has built up internal tables to indicate the addresses of input layer fragments, the order of layer output and the order of input file opening. IMERGE has now started output of the concatenated layer fragments to the selections-output-file.

User action: None.

WRESTLAY, writing remaining layers to 'file-spec'

Explanation: This is output as an indication of what IMERGE is currently doing. IMERGE has scanned all the input files for layer fragments and has built up internal tables to indicate the addresses of input layer fragments, the order of layer output and the order of input file opening. IMERGE has now started output of the concatenated layer fragments to the rest-output-file.

User action: None.

MESSAGES (WARNING)

These messages indicate that IMERGE has been unable to guarantee that a specified feature has been correctly processed.

CONFLMD, conflicting Map Descriptor information

Explanation: If mergeing from more than one input file IMERGE checks that the type 2 Map Descriptors are all set to common values for projection, measurement units, spheroid etc. The first input file is considered to have the definitive Map Descriptor information. The accompanying messages will indicate what aspect of the Map Descriptor is at fault and in which file.

User action: Use ITRANS to transform all the input files into a common projection, measurement system, etc. Re-run IMERGE.

DUPINFIL, input IFF files share the same specification

Explanation: IMERGE checks that the user has not specified the same input file more than once. "Double digitising" in the output file will result if IMERGE is allowed to proceed.

User action: Respecify the command line but omit one of the duplicate file-specs.

DUPOUTFIL, output IFF files share the same specification

Explanation: If the user uses the /SPLIT=filespec qualifier, IMERGE checks that the user has not supplied the same specification for the rest output file and the selections-output-file. It would be very dangerous to continue as although the two files output will contain different data, they will have adjacent version numbers in the same directory and are liable to be purged. In addition the files will have almost identical creation times and will be difficult to distinguish within a production environment.

User action: Respecify the command line, ensuring that the output files do not share the same specification.

INAPPCPS, inappropriate CP entries for /CP_EXPANSION qualifier

Explanation: The /CP_EXPANSION qualifier has been specified for use on input files which will not form a rectangular merged map area.

User action: Consider why the /CP_EXPANSION qualifier was specified. As IMERGE is unable to expand the control points for you, respecify the IMERGE command line but omit the /CONTROL_POINTS qualifier. After re-running IMERGE use IPATCH to change the control point entry in the output files.

INVALCP, inappropriate CP entry for /CP_EXPANSION qualifier in file 'file-spec'

Explanation: Either the input file lacks a CP entry or a CP entry has been read that has zero extent.

User action: Check the offending file with IINFO and modify the CP entry using IPATCH/WRITE

INVCPQUAL, /CP_EXPANSION specified when mergeing input file to itself

Explanation: This would be a pointless exercise as the Control Points will be EXACTLY the same regardless of the /CP_EXPANSION qualifier.

User action: Re-specify the command line.

INSUFHD, insufficient IFF header entries in input files

Explanation: The input IFF files do not have all the usual (and required) header entries. This is most irregular, and is probably the result of output from an erroneous program, or the mis-use of IPATCH in /WRITE mode. IMERGE will apply default values any of the following entries which were missing:

- MD, (Map Descriptor),
- MH, (Map Header),
- NS, (New Section)
- CC, (Cubic Coefficients), and,
- CP, (Control Points).

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files to ensure that the default IFF header entry values are acceptable.

MISSCC, missing IFF CC entry in first input file

Explanation: The first IFF file does not have a CC (Cubic Coefficients) entry. This is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode. IMERGE will attempt to take the CC entry from the second or subsequent input file, and use this as the CC entry for the output files.

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files.

MISSCC, missing IFF CP entry in first input file

Explanation: The first IFF file does not have a CP (Control Points) entry. This is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode. IMERGE will attempt to take the CP entry from the second or subsequent input file, and use this as the CP entry for the output files.

User action: Determine the cause of the input file corruption and rectify. Check the CP entry in the IMERGE output files. If /CP_EXPANSION was specified then the results of the control point expansion will probably be erroneous.

MISSMH, missing IFF MH entry in first input file

Explanation: The first IFF file does not have a Map Header entry. This is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode. IMERGE will attempt to take the MH entry from the second or subsequent input file, and use this as the MH entry for the output files.

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files.

MISSNS, missing IFF NS entry in first input file

Explanation: The first IFF file does not have a New Section entry. This is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode. IMERGE will attempt to take the NS entry from the second or subsequent input file, and use this as the NS entry for the output files.

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files.

NOMD, no map descriptor found in 'file-spec'

Explanation: The specified IFF file does not have a map descriptor entry. This is most irregular, and is probably the result of output from a badly written user routine or the mis-use of IPATCH in /WRITE mode. IMERGE will attempt to continue.

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files.

NOV2MD, IFF map descriptor in 'file-spec' is not version 2

Explanation: IMERGE expects input files to have type 2 map descriptors as it offers offset merging functionality based on the contents of the map descriptor. IMERGE is downwards compatible with old pattern IFF files which have type 1 map descriptors, but no origin offset facility is supported for the earlier pattern files.

User action: If origin offsetting is required use ITRANS/DESCRIPTOR to create a copy of the IFF file having a type 2 map descriptor.

QUALPOSINV, /LAYER qualifier positioning invalid

Explanation: The /LAYER qualifier must appear AFTER the command verb but BEFORE the input file parameters if global layer selections are desired. Alternatively the /LAYER qualifier may be placed AFTER any of the input file parameters but BEFORE the output file parameter if individual input file layer selections are desired.

User action: Respecify the command line.

UNSETMD, map descriptor in 'file-spec' is unset

Explanation: The Map Descriptor in the specified input file is unset.

User action: If origin offsetting is required or other input files have characteristics which require the map descriptor to be set, use ITRANS/DESCRIPTOR to set up the map descriptor. Re-run IMERGE.

UNEXPENT, unexpected IFF 'code' entry at address 'int' in IFF file
'file-spec' - attempting to continue

Explanation: The specified IFF entry type was found among the header entries of the specified IFF file. This is most irregular, and is probably the result of output from an erroneous program, or the mis-use of IPATCH in /WRITE mode. IMERGE will attempt to continue.

User action: Determine the cause of the input file corruption and rectify. Check the IMERGE output files.

UNSORTFSN, FSN's in first IFF file are not in ascending order at Layer 'number',
FSN 'code'

Explanation: The current FSN found is either duplicated or not in order. IMERGE will attempt to continue leaving those duplications or irregularly-ordered features in the output file(s).

User action: Check the first input file to ensure that all FSN's are unique and in an ascending order.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. The output files will be deleted.

EONOTFND, IFF EO entry not found in 'file-spec'

Explanation: While scanning the specified IFF input file for layer fragments an IFF EO entry was found to be missing or the address of an EO was incorrect within the corresponding NO entry.

User action: Use LITES2 (or ITOTEXT followed by IFROMTEXT) to repair the file.

ILEGLAY, illegal IFF layer number 'integer'

Explanation: The layer number specified as an argument to a /LAYER qualifier does not lie in the range 0 to 32767 or contains a non-numeric character.

User action: Respecify the command line.

IFFNOLYRS, no layers found in IFF file 'file-spec'

Explanation: IMERGE operates at the IFF layer level. Unfortunately, the specified input file does not contain any layers at all.

User action: Use ITOTEXT followed by IFROMTEXT to repair the file. Re-run IMERGE.

MERNOLYRS, no selected layers to merge in IFF file 'file-spec'

Explanation: Layer selection was specified with the /LAYER qualifier. The specified input file contains none of the selected layers.

User action: Use IINFO to determine the layers within the offending file. Re-specify the IMERGE command line.

MERNOSPLYRS, no layers left for /SPLIT. All have been selected

Explanation: Although the user expected layers to be left over after selections with /LAYER there are none left to put into the rest-output-file. The /SPLIT qualifier was therefore superfluous and an empty rest-output-file would result if execution was allowed to continue.

User action: Re-specify the command line, either reducing the number of layers selected for the selections-output-file or omitting completely the /SPLIT qualifier.

MISSQ, missing qualifier; /REST_LAYER requires /LAYER

Explanation: If /SPLIT is specified without having made layer selections from the input files there can be no layers left after selection for the rest-output-file. Without /SPLIT and /LAYER, /REST_LAYER is useless.

User action: Consider why you want the /SPLIT qualifier and then re-type the command line making appropriate layer selections.

MISSQ, missing qualifier; /SPLIT requires /LAYER

Explanation: If /SPLIT is specified without having made layer selections from the input files there can be no layers left after selection for the rest output file.

User action: Consider why you want the /SPLIT qualifier and then re-type the command line making appropriate layer selections.

TOMNYLAY, too many layers found in input files (max allowed is 'integer')

Explanation: The input files together contain more than the indicated maximum number of layers that IMERGE can tabulate.

User action: Consider why there are so many layers. Run IMERGE on only some of the input files at a time if you wish to concatenate a lot of layer fragments, then merge together the files with the concatenated layers.

UNEXPEOF, unexpected end of IFF file 'file-spec'

Explanation: The specified input IFF file terminated before an IFF EJ entry was encountered.

User action: Use IMEND to correctly terminate the file. Re-run IMERGE on the repaired IFF file. If the problem persists try reading the file into LITES2 and then exit. Re-run IMERGE.

MESSAGES (FATAL)

These messages indicate a fatal error in processing which has resulted in immediate program termination and may indicate an error in the program itself. The output files will be deleted.

NONOTFND, IFF NO entry not found in 'file-spec'

Explanation: IMERGE has scanned all the input files for layer fragments and has built up internal tables to indicate the addresses of input layer fragments, the order of layer output and the order of input file opening. IMERGE has started output of the concatenated layer fragments to the rest output file but now cannot find one of the fragments in the specified file. The address of the offending layer is probable corrupt.

User action: Please take a copy of all the input files and then notify Laser-Scan. Try ITOTEXT followed by IFROMTEXT to repair the file. Re-run IMERGE.

MESSAGES (OTHER)

In addition to the above messages which are generated by IMERGE itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 11

MODULE IPATCH

MODULE IPATCH

REPLACES DAMP module IED

FUNCTION

IPATCH is the IFF file patch editor. It allows the user to inspect, edit or delete the various entries in the file. The file is not copied, and changes made in write mode are performed in-situ. Thus it is not possible to add extra information.

The default mode of operation is to open the IFF file for reading only.

FORMAT

\$ IPATCH IFF-file-spec

Command qualifiers

/[NO]COMMANDS[=file-spec]
/[NO]FRT[=file-spec]
/[NO]JOURNAL[=file-spec]
/[NO]LOG
/[NO]REVISION_LEVEL=integer
/[NO]STATUS
/WRITE

Defaults

/NOCOMMANDS
/NOFRT
/NOJOURNAL
/NOLOG
/REVISION_LEVEL=-1
/STATUS
Open file for read only

PROMPTS

_IFF file: IFF-file-spec

IPATCH is an interactive, command driven utility. Command input is expected when the following prompt is issued:

IPATCH>

Editing commands may also prompt for values - the prompt used depends upon the answer required.

PARAMETER

IFF-file-spec

- specifies the IFF file to be edited.

Any part of the file-spec that is not specified will be completed from the default of "LSL\$IF:IFF.IFF;0"

QUALIFIERS

/COMMANDS [=file-spec]
/NOCOMMANDS

- specifies that commands will be taken from a file, instead of from prompts to the user. The file-spec defaults to 'IFF-name'.IPATCH, where 'IFF-name' is the name of the IFF file being edited.

If the last command in the command file is not EXIT, then a warning message will be given, and IPATCH will start to prompt for commands.

Great care should be taken when using the /COMMANDS switch, as IPATCH will not stop editing when an error occurs. This can mean that a command file could have disastrous effects due to a small difference between the file on which the commands were journalled and the file on which /COMMANDS is used.

/FRT=file-spec
/NOFRT

- specifies an FRT (Feature Representation File) to be read. Missing parts of the file-spec are taken from 'LSL\$FRT:.FRT'.

Only the ACD (Attribute Code Definition) part of the FRT is relevant to IPATCH. If no FRT file is given, then only the default attribute codes will be known to IPATCH.

/JOURNAL [=file-spec]
/NOJOURNAL

- specifies that commands will be journalled to a file. The file-spec defaults to 'IFF-name'.IPATCH, where 'IFF-name' is the name of the IFF file being edited.

Note that everything that the user types is journalled, before the program attempts to obey the line. Thus the journal file contains a complete record of the IPATCH session. The journal file can be very valuable documentation when a problem with IPATCH is being reported to Laser-Scan.

If <CTRL/Z> is used to exit from IPATCH, this will be journalled as if the EXIT command had been used. Therefore all normal journal files will end with EXIT.

/LOG
/NOLOG

- specifies that extra information is to be written to the terminal.

When the program starts up, the name of the IFF file, and of any command or journal files, are logged. When the program exits, a message containing the IFF file name is output.

If /COMMAND has been used to specify that commands be taken from a file, the /LOG command makes IPATCH reflect each command line to the terminal as it is read.

/REVISION_LEVEL=integer
/NOREVISION_LEVEL

- specifies the IFF input revision level to be used for reading the IFF file. The default of -1 causes all entries to be displayed exactly as they are in the file, but 0 may be used to force CB entries to be displayed as ST or ZS entries, while 1 may be used to force ST or ZS entries to be displayed as CB entries.

/STATUS
/NOSTATUS

- specifies that the program should write and maintain the status area at the top of the screen. This is more or less the effect that would be gained if the command STATUS ON could be given as soon as the program starts up.

This is the default state when the program is running with its output directed to a terminal. To suppress the status area, use the /NOSTATUS switch.

/WRITE

- specifies that the IFF file is to be opened for writing. The default is to open the IFF file for reading only, which means that it can be inspected but not edited. When /WRITE is specified, entries may be edited or deleted. Note that all edits take effect immediately - they are not buffered up for execution later.

DESCRIPTION

General

IPATCH is the primary means of inspecting an IFF file at the entry level, and of altering the data in that form. It thus provides a sophisticated set of movement and search commands to allow movement within the file, and editing commands tailored to each entry to facilitate alteration of data. It is also possible to delete an entry or a series of entries.

The things that IPATCH will not do are:

- o edit the HI entry - the history entry is maintained by utility programs as a record of an IFF file's life, and may not be altered.
- o edit the MH entry - apart from amending the customer number. The format of the map header is customer dependent, so a program tailored for the particular customer should be used.

- o edit the MD entry - the map descriptor describes the current form of the positional data in the IFF file, and is only changed when the data itself is changed. Use the ITRANS utility to transform the IFF file.
- o edit a broken file - use the IMEND utility to mend an improperly terminated IFF file. IMEND should always be used on a broken file before an attempt is made to do anything with it.

The dangers of using IPATCH are:

- o editing is performed in-situ, so that any edit performed is permanent. Always take a copy of the IFF file before editing it with IPATCH.
- o it is possible to move to a random position in the IFF file, using the FIND command. Editing at such a random position can then scramble the entries that enclose the position.
- o if an editing session has been journalled to a file, great care should be taken when replaying it, as IPATCH will almost always continue after a problem.

How the PREVIOUS command works

The PREVIOUS command allows movement backwards in the IFF file. Unfortunately, the IFF library does not support sequential backwards movement - it provides a next-entry call and a go-to-this-position call, but not a previous-entry call. Therefore IPATCH has to maintain some sort of list of where the entries before the current one are.

What IPATCH actually does is keep a list of the addresses of NF entries, starting at the start of file. This list is updated every time the NEXT or TYPE commands are used to move beyond its end - effectively the list is 'stretched' by successive NEXT commands.

<-----scanned-area----->

RA.....NF.....NF.....NF.....NF.....NF.....NF..

↓
^

current position

Whilst the current position is between two NF entries in the list, IPATCH also reads in the addresses of all the entries in the relevant 'feature', where a 'feature' is considered to start at one NF entry and end just before the next.

The program can thus move backwards within the current 'feature' very quickly. If it needs to move further back, it will read each previous 'feature' in as it goes, until it reaches the required entry.

<-----scanned-area----->

RA.....NF.....NF.....NF.....NF.....NF.....NF..

↓
^

current position

However, searching for an entry, or using the FIND or EOF commands, can move the current position beyond the end of the area that IPATCH has in its list. In this case, the status area will note that "PREVIOUS will have to scan". If the PREVIOUS command is given, IPATCH will have to extend the list by scanning from the last NF it knows about up to the NF after the current position.

```
<-----scanned-area----->
RA.....NF.....NF.....NF.....NF.....NF.....NF..
                                     |
                                     ^
                                current position
```

The list can now be used to move backwards. However, it should be realized that in a large IFF file the new current position could easily be a very long way beyond the scanned area. In these circumstances, the PREVIOUS command could involve quite a long delay as IPATCH looks for all the intermediate NF entries.

A side effect of having the PREVIOUS list is encountered when using the FIND command to position within the IFF file. If the address moved to is within the scanned area, then IPATCH will be able to tell if the entry found is reachable by a sequence of NEXT commands from the start of the file. If this is not the case, then the entry is "not simply reachable", and a warning message will be output to that effect. In this case, it is likely that the entry is not real, but just a coincidental bit pattern that looks superficially like an IFF entry.

Wildcarding

The wildcard character, *, may be used as a placeholder when searching and when editing in line mode. It allows the specification of a value when earlier values are not required. For instance, to search for an AC with a particular text, give the command

```
IPATCH> AC * * "Text that is wanted by the user"<CR>
```

This says that the AC type and value may take any value, and that it is only the text that is important.

Similarly, editing the AC when it is found could use wildcards - for instance

```
IPATCH> / * * "Text that was wanted"<CR>
```

edits the text, ignoring the AC type and value.

Comments

The ! character (exclamation mark) introduces a comment - anything after the comment character is not to be obeyed by the program. The comment character may occur at the start of a line, or after a valid command.

If journalling is in progress (due to the /JOURNAL switch) then comments are journalled like any other input.

Searching

It is possible to ask IPATCH to find the next occurrence of a particular entry, possibly specifying values for the contents of that entry.

IPATCH will move forwards through the IFF file until it either finds an entry which satisfies the search request, or reaches the end of file. If it finds an entry, then it makes that entry current. If it reaches end of file, it gives a warning message and does not move the current position - ie the entry that was current before the search is still current.

The basic search request is simply the name of the entry - for instance

```
IPATCH> AC<CR>
```

will look for the next AC entry, and

```
IPATCH> FS 245<CR>
```

will look for the next FS entry, with feature code 245. Note that this search request is equivalent to the (unnecessarily) wildcarded request

```
IPATCH> FS 245 * * * <CR>
```

Finally, the search request

```
IPATCH> NF * 701<CR>
```

will search forwards for an NF entry with internal sequence number 701, and any feature serial number.

IPATCH COMMANDS

*

Moves the current position back to the start of the file.

FORMAT: *

DESCRIPTION:

The * command moves the current position back to the start of the IFF file, before the first entry.

This command is the only command which may be followed by another command on the same line. This second command will be obeyed after moving to the start of the file - for instance *NEXT would move to the first entry in the file. Note that it is not necessary to separate the * and the succeeding command with spaces.

Examples:

IPATCH> ***<CR>**
00000000: Start of file
IPATCH>

IPATCH> ***NEXT 5<CR>**
000011D6: NS "created by IFROMTEXT at 15:39:49 on 30-OCT-85"
IPATCH>

?

Prints the current entry out again.

FORMAT: ?

DESCRIPTION:

The ? command prints the current entry out again, in the same format as produced by NEXT. It does not change the current entry.

Examples:

```
0000123B: NF      170      1
IPATCH> ?<CR>
0000123B: NF      170      1
IPATCH>
```

/

Introduces edit mode.

FORMAT: /
 / values
 / command value

DESCRIPTION:

The / command introduces edit mode.

There are three forms of edit mode:

- o Prompt mode, where the program prompts for values
- o Line mode, where the new values are presented on the command line in the appropriate order (possibly including wildcards)
- o Command mode, where the field to be edited is identified by a command followed by the new value

Note that it is only possible to edit an entry when the IFF file has been opened /WRITE.

Prompt mode

/

If the / command is given on a line by itself, then prompt mode is entered. A prompt is given for each field of the current entry, and a new value may be given as reply. A reply of <CR> is taken to mean that the original value should be used. <CTRL/Z> is treated as <CR>

Prompt mode is not supported for the CP, CC, ST, ZS, or CB entries.

Line mode

/ values

If the / command is immediately followed by appropriate values on the command line, then those values will be used to replace the fields of the current entry, in the relevant order. The wildcard character (*) may be used as a position holder for a field which is not to be changed.

Line mode is not supported for the CP, CC, ST, ZS, or CB entries.

Command mode

/ command value

Almost all entries may be edited in command mode. For each such entry, a series of command mnemonics are supplied to allow individual fields (or parts of fields) to be altered. For instance, the FS entry edit command /FC 5 sets the feature code to 5.

See the documentation for each IFF entry for a definition of the edit commands available.

Examples:

```
00001236: NO      1    0000  00000000
IPATCH> / LAYER 2<CR>
00001236: NO      2    0000  00000000
IPATCH> / * * 1714<CR>
00001236: NO      1    0000  00001714
IPATCH> NEXT<CR>
0000123B: NF      170      1
IPATCH> /<CR>
FSN:
ISN: 170
0000123B: NF      170      170
IPATCH> NEXT<CR>
0000123E: FS      123    0000    0000    0000
IPATCH> / TEXT<CR>
0000123E: FS      123    0000    8000    0000
IPATCH> / POSITION 3<CR>
0000123E: FS      123    0000    8003    0000
IPATCH> NEXT<CR>
00001243: AC Height (3) = 67.9  "This text belongs to a type 3 AC.."
IPATCH> / Contour 70 "This text belongs to a type 2 AC.."<CR>
00001243: AC Contour (2) = 70  "This text belongs to a type 2 AC.."
IPATCH>
```

DELETE

Deletes the current entry

FORMAT: **DELETE**

DESCRIPTION:

The DELETE command deletes the current entry, replacing it by a void (VO) entry of the appropriate size. This command does not do anything if the file is not opened for write.

Note that if there is a current mark (see the MARK command), the DELETE command will have removed it.

Warning messages:

The following warning messages are specific to the DELETE command:

GOTOVO, error returning to just deleted entry (void)
NEXTDEL, error moving to next entry, to delete this one
NOCURDEL, no current entry to delete
NOTWRITE, IFF file has not been opened for write

Examples:

1. an attempt to delete an entry in a file that has been opened for read:

```
0000123B: NF      170      1
IPATCH> DELETE<CR>
%IPATCH-W-NOTWRITE, IFF file has not been opened for write
0000123B: NF      170      1
IPATCH>
```

2. deleting the same entry when the file has been opened for write:

```
0000123B: NF      170      1
IPATCH> DELETE<CR>
0000123B: VO        2
IPATCH>
```

DELETE MARK

Delete entries from the marked entry to the previous entry.

FORMAT: DELETE MARK

DESCRIPTION:

The DELETE MARK command deletes all entries from (and including) the marked entry to the previous entry. The deleted entries are replaced by a void (VO) entry of the appropriate size. The current entry is not deleted.

The marked entry (at which deletion starts) is indicated with the MARK command. Note that after DELETE MARK, there is no longer any mark on the (now voided) entry.

Warning messages:

The following warning messages are specific to the DELETE MARK command:

DEORDER, current entry is before the marked entry - can't delete to the mark
MARKDEL, no marked entry to delete from
NOTWRITE, IFF file has not been opened for write

Examples:

```
0000123B: NF      170      1
IPATCH> DELETE MARK<CR>
0000123B: NF      170      1
IPATCH> PREVIOUS<CR>
00001224: VO       22
IPATCH>
```

DISPLAY OFF

Stop NEXT command displaying intermediate entries

FORMAT: DISPLAY OFF

DESCRIPTION:

After this command, NEXT will only print out the destination entry. Thus the command NEXT 5 would not print out the intermediate entries (entries 1 to 4), but only the 5th entry, which becomes current.

This is the default state.

Information messages:

The following information messages are specific to the DISPLAY OFF command:

DISLAYOFF, NEXT will now only display the destination entry

Examples:

IPATCH> **DISPLAY OFF**<CR>
%IPATCH-I-DISLAYOFF, NEXT will now only display the destination entry
IPATCH> **NEXT 5**<CR>
0000126B: EF
IPATCH>

DISPLAY ON

Enable the display of intermediate entries by NEXT

FORMAT: **DISPLAY ON**

DESCRIPTION:

After this command, NEXT will print out all entries that it passes over. Thus the command NEXT 5 would print out the intermediate entries (entries 1 to 4), as well as the 5th entry, which becomes current.

Information messages:

The following information messages are specific to the DISPLAY ON command:

DISPLAYON, NEXT will now display intermediate entries

Examples:

```
IPATCH> DISPLAY ON<CR>
%IPATCH-I-DISPLAYON, NEXT will now display intermediate entries
IPATCH> NEXT 5<CR>
0000126C: NF      170      2
0000126F: FS      123      0000      0000      0000
00001274: AC Height (3) = 67.9  "This text belongs to a type 3 AC.."
00001289: TH       30
0000128B: ST       4      0000
IPATCH>
```

EOF

Move current position to the end of file

FORMAT: **EOF**

DESCRIPTION:

The EOF command moves the current position to the end of the IFF file, after the last entry. Note that the first attempt to PREVIOUS from the end of file will cause IPATCH to scan up to the end, which can take a while for a large file.

Examples:

0000128B: ST 4 0000
IPATCH> **EOF<CR>**
00006199: End of file
IPATCH>

EXIT

Exit from the IPATCH program

FORMAT: **EXIT**

DESCRIPTION:

The EXIT command causes IPATCH to exit. Note that any editing performed by IPATCH was done at the time it was requested, not at the time of EXIT.

<CTRL/Z> (pressing the Ctrl and Z keys together) may also be used to exit the program.

Examples:

IPATCH> **EXIT<CR>**

ELAPSED: 00:05:25.84 CPU: 0:00:05.71 BUFIO: 281 DIRIO: 46 FAULTS: 263
\$

EXPLAIN

Explain the current entry

FORMAT: **E**
 EXPLAIN

DESCRIPTION:

The EXPLAIN command causes IPATCH to explain the meaning of the current entry. The fields of the entry will be named, and any bit-set fields will be explained.

Examples:

```
0000128B: ST      4    0000
IPATCH> EXPLAIN<CR>
0000128B: ST      4    0000
2-dimensional point string entry
- size of entry = 16 words
- number of points = 4
- flag bits = 0000 - ie pen    UP to first point
IPATCH> FS<CR>
000012A0: FS     123    0000    0000    0000
IPATCH> EXPLAIN<CR>
000012A0: FS     123    0000    0000    0000
Feature Status entry
- size of entry = 4 words
- FC (feature code) 123
- word 2 = 0000 hex
    CLOSED   = 0      (feature is open)
    NOTEDGE  = 0      (feature is an edge, not a line)
    REVERSE  = 0      (feature is not reversed)
    TWOPART  = 0      (feature is not two-part)
    PAINTOUT = 0      (feature is not paintout - keep it)
    SQUARE   = 0      (squaring flag clear)
    INVERSE  = 0      (normal polarity)
    SUPPRESS = 0      (no paintout suppress)
- word 3 = 0000 hex
    text/symbol bits = 0  (linear feature)
    PC = 0           (process code)
- word 4 = 0000 hex - user specific data
IPATCH>
```

FIND

Move to a given address in the file

FORMAT: **F** **address**
 FIND **address**

Command parameters:

address

The hexadecimal address within the IFF file which is to be made the new current position

DESCRIPTION:

The FIND command changes the current position to the given hexadecimal address in the IFF file, and takes in hand any entry there.

If the program has scanned past that address (for instance by using NEXT to move past it), then IPATCH will report if it does not believe that the entry that you have positioned to is 'real' - that is, reachable by a sequence of NEXT commands from the start of file.

The FIND command is mainly intended to allow inspection of an IFF file after a corrupted area, which may stop the NEXT command from working over that area. It can also be useful for finding a problem reported by a utility program, when the error message contained an IFF file address.

You should realize, however, that using a series of NEXT entries from a random position in the file can sometimes provide a superficially sensible set of entries, even though they are only due to the accidental bit patterns caused by the real entries in that area.

Warning messages:

The following warning messages are specific to the FIND command:

BADFIND, cannot find before 00000001 or after 'end-of-file' - nothing there
NOADDRESS, no (hexadecimal) address given after FIND
NOENTRY, no entry at that position
NOTENTRY, entry 'entry' at 'hex-address' is not simply reachable

Examples:

```
IPATCH> FIND 128B<CR>
0000128B: ST      4    0000
IPATCH> FIND 123<CR>
%IFF-E-IFFERR, IFF error BINC on LUN 1
%IFF-E-IFFERR, IFF error BINC on LUN 1
%IPATCH-W-NOTENTRY, entry XX at 00000123 is not simply reachable
00000123: XX      249
IPATCH>
```

FRT

Read in an FRT file

FORMAT: **FRT file-spec**

Command parameters:

file-spec

The name of the FRT file to be read. Missing parts of the file-spec are taken from 'LSL\$FRT:.FRT'.

DESCRIPTION:

The FRT command causes an FRT file to be read, possibly as a replacement for one specified on the IPATCH command line. The FRT library may produce messages concerning any errors in the FRT file.

Only the ACD (Attribute Code Definition) part of the FRT is relevant to IPATCH.

HELP

Give help on a subject

FORMAT: H subject
 HELP subject

Command parameters:

subject

The subject on which help is required

DESCRIPTION:

The HELP command looks the rest of the line up in the IPATCH HELP library. This library contains a brief summary of the operation of each command, and examples of the operation of each.

The information is looked up in the IPATCH section of the IMP help library, LSL\$HELP:IMP.HLB.

LINE

Look for the next FS with text and symbol bits unset in word 3

FORMAT: **LINE**

DESCRIPTION:

The LINE command performs a search for the next FS entry with the text and symbol bits both unset in the third word. See the FS entry documentation for further details.

Warning messages:

The following warning messages are specific to the LINE command:

FSBITS, cannot find linear FS

Examples:

IPATCH> **LINE**<CR>
000001F1: FS 1 0000 0000 0000
IPATCH>

LIST

List the points in the current point string (ST,ZS,CB) entry

FORMAT: L [from [to]]
 LIST [from [to]]

Command parameters:

from

The lower limit of the sequence of points to be listed.

to

The upper limit of the sequence of points to be listed

DESCRIPTION:

The LIST command types the specified points from the current ST, ZS, or CB entry.

If no parameters are specified, then all of the points will be listed. If only one parameter is specified, then all the points starting with the from'th will be listed. If both parameters are specified, then the from'th to to'th points will be listed.

Warning messages:

The following warning messages are specific to the LIST command:

INTRANGE, 'integer' is out of range - should be between 1 and 'npts'
NOTSTRING, LIST will only work for an ST, ZS, or CB
ORDER, cannot work from 'from' to 'to' - try again in the other order

Examples:

```
0000125A: ST      4      0000
IPATCH> LIST 2 3<CR>
0000125A: ST      4      0000
      2:      0.8506,      16.8394
      3:      2.2290,      18.8454
IPATCH>
```

MARK

Mark the current entry

FORMAT: **M**
 MARK

DESCRIPTION:

The MARK command is used to mark the current entry.

The RETURN command can then be used at any time to reposition to the marked entry, and the DELETE MARK command can be used to delete from it.

The DELETE and DELETE MARK commands both remove the mark from the marked entry.

Warning messages:

The following warning messages are specific to the MARK command:

NOCURMARK, no current entry to MARK

Information messages:

The following information messages are specific to the MARK command:

MARKED, current entry now marked

Examples:

```
0000126C: NF      170      2
IPATCH> MARK<CR>
%IPATCH-I-MARKED, current entry now marked
0000126C: NF      170      2
IPATCH>
```

NEXT

Move to the next entry

FORMAT: N [count]
 NEXT [count]
 NEXT *

Command parameters:

count

The number of entries by which to move forwards. This defaults to 1.

DESCRIPTION:

The NEXT command is used to move to the next entry in the IFF file, and make it current. If an integer argument is given, then the effect is as of repeating the NEXT command that number of times. If NEXT * is used, then the effect is as of repeating the NEXT command until the end of file is reached (and made current).

If the DISPLAY ON command has been given, then the intermediate entries will be displayed. Otherwise only the final entry (the one which is made current) will be displayed.

Note that the NEXT * command is only obeyed when DISPLAY is ON - otherwise an error message is given and the command ignored. If you just want to move to the end of file, use the EOF command.

See also the discussion of IFF file scanning.

Warning messages:

The following warning messages are specific to the NEXT command:

BADNEXT, argument to NEXT must be greater than zero
NEXTEOF, you can't NEXT off the end of the file
NEXTERR, NEXT command abandoned after 'integer' entries
NEXTONE, NEXT command abandoned after 1 entry
WILDNEXT, DISPLAY is OFF - attempt to do "NEXT *" ignored

Examples:

```
IPATCH> NEXT 5<CR>
00001243: AC Height (3) = 67.9  "This text belongs to a type 3 AC.."
00001258: TH          30
0000125A: ST          4    0000
0000126B: EF
0000126C: NF        170      2
IPATCH>
```

PREVIOUS

Move to the previous entry

FORMAT: P [count]
 PREVIOUS [count]

Command parameters:

count

The number of entries to move backwards by. This defaults to 1.

DESCRIPTION:

The PREVIOUS command is used to move to the previous entry in the IFF file, and make it current. If an argument is given, then the effect is as of repeating the PREVIOUS command that number of times, although without displaying the intermediate entries.

See also the discussion of IFF file scanning.

Warning messages:

The following warning messages are specific to the PREVIOUS command:

DISABOVE, PREVIOUS is disabled above 'address'
PREVBETWEEN, PREVIOUS will not work between 'address' and 'address'
PREVBOF, you can't PREVIOUS off the start of the file
PREVERR, PREVIOUS command abandoned
PREVNF, error looking for NF entry, during PREVIOUS scan
PREVXX, error looking for start-of-file entry, during PREVIOUS scan

Examples:

```
IPATCH> NEXT 5<CR>
00001243: AC Height (3) = 67.9 "This text belongs to a type 3 AC.."
00001258: TH      30
0000125A: ST      4    0000
0000126B: EF
0000126C: NF    170      2
IPATCH> PREVIOUS<CR>
0000126B: EF
IPATCH> PREVIOUS 2<CR>
00001258: TH      30
IPATCH>
```

RETURN

Return to the marked entry

FORMAT: RETURN

DESCRIPTION:

The RETURN command makes the marked entry the current entry again.

The MARK command is used to mark an entry.

Warning messages:

The following warning messages are specific to the RETURN command:

MARKERR, error in positioning back to MARKed entry
NOMARK, no marked entry to RETURN to

Examples:

00001307: AC Height (3) = 67.9 "This text belongs to a type 3 AC.."
IPATCH> **RETURN<CR>**
0000126C: NF 170 2
IPATCH>

REVISION_LEVEL

Alter the IFF input revision level.

FORMAT: **REVISION_LEVEL level**

Command parameters:

level

The new IFF input revision level to use. This must be -1, 0, or 1.

DESCRIPTION:

The REVISION_LEVEL command is used to alter the IFF input revision level used for reading the file. The original level used defaults to -1, but may be set using the /REVISION_LEVEL command qualifier. The default of -1 causes all entries to be displayed exactly as they are in the file, but 0 may be used to force CB entries to be displayed as ST or ZS entries, while 1 may be used to force ST or ZS entries to be displayed as CB entries.

After a REVISION_LEVEL command, the current entry is displayed.

Warning messages:

The following warning messages are specific to the REVISION_LEVEL command:

BADREV, unknown revision level 'integer', must be -1, 0, or 1

Examples:

```
IPATCH> NEXT<CR>
0000029E: ZS      5  0001
IPATCH> REVISION_LEVEL 1<CR>
0000029E: CB      5    01    1    7    5
IPATCH>
```

STATUS

Print current status information

FORMAT: **STATUS**

DESCRIPTION:

The STATUS command prints a summary of the current file status. This is the same information as would be output in the status area if the STATUS ON command had been given.

Examples:

```
IPATCH> STATUS<CR>
Reading file DUA0:[LSL.IFF]WINNIE_THE_POOH.IFF;2
current 0000126C  NF   170,      2 at 0000126C  mark at 0000126C
eof      00006199  revision levels: input -1 output 1
      5 features scanned (below 00001330)
NEXT will not display intermediate entries
IPATCH> DISPLAY ON<CR>
%IPATCH-I-DISPLAYON, NEXT will now display intermediate entries
IPATCH> FS 123<CR>
00001368: FS   123   0000   0000   0000
IPATCH> STATUS<CR>
Reading file DUA0:[LSL.IFF]WINNIE_THE_POOH.IFF;2
current 00001368  no previous known NF          mark at 0000126C
eof      00006199  revision levels: input -1 output 1
      5 features scanned (below 00001330) - PREVIOUS will have to scan
NEXT will display intermediate entries
IPATCH>
```

STATUS OFF

Remove the status area from the screen

FORMAT: **STATUS OFF**

DESCRIPTION:

The STATUS OFF command switches off the status area at the top of the screen, and clears the screen. The default is to have a status area.

STATUS ON

Request the display of a status area

FORMAT: **STATUS ON**

DESCRIPTION:

The STATUS ON command causes IPATCH to write and maintain a status area at the top of the screen.

The status area is only supported on ANSI terminals (ie VT100 and VT200 series).

Warning messages:

The following warning messages are specific to the STATUS ON command:

CHKTERM, error checking to see if SYS\$OUTPUT: is an ANSI terminal
NOTANSI, SYS\$OUTPUT: is not an ANSI terminal

SYMBOL

Look for the next FS with symbol bit set in word 3

FORMAT: **SYMBOL**

DESCRIPTION:

The SYMBOL command performs a search for the next FS entry with the symbol bit set and the text bit unset in the third word. See the FS entry documentation for further details.

Warning messages:

The following warning messages are specific to the SYMBOL command:

FSBITS, cannot find symbol FS

Examples:

IPATCH> **SYMBOL**<CR>
00000221: FS 54 0000 4000 0000
IPATCH>

TEXT

Look for the next FS with text bit set in word 3

FORMAT: **TEXT**

DESCRIPTION:

The TEXT command performs a search for the next FS entry with the text bit set and the symbol bit unset in the third word. See the FS entry documentation for further details.

Warning messages:

The following warning messages are specific to the TEXT command:

FSBITS, cannot find text FS

Examples:

IPATCH> **TEXT**<CR>
00000209: FS 28 0000 8083 0000
IPATCH>

TO DECIMAL

Convert a hexadecimal number to decimal

FORMAT: **TO DECIMAL hex-number**

Command parameters:

hex-number

The hexadecimal number which is to be converted to decimal.

DESCRIPTION:

The TO DECIMAL command translates the given integer value from hexadecimal to decimal, and prints it out.

Examples:

IPATCH> **TO DECIMAL 123B<CR>**
123B hexadecimal is 4667 decimal
IPATCH>

TO DEGREES

Convert an angle from radians to degrees

FORMAT: **TO DEGREES radians**

Command parameters:

radians

The real number representing the desired angle in radians.

DESCRIPTION:

The TO DEGREES command translates the given angle from radians to degrees, and prints it out.

Examples:

IPATCH> **TO DEGREES 3.1415<CR>**
3.141 radians is 179.995 degrees
IPATCH>

TO HEXADECIMAL

Converts a decimal number to hexadecimal

FORMAT: **TO HEXADECIMAL integer**

Command parameters:

integer

The decimal integer which is to be converted to hexadecimal.

DESCRIPTION:

The TO HEXADECIMAL command translates the given integer value from decimal to hexadecimal, and prints it out.

Examples:

IPATCH> **TO HEX 1234**<CR>
1234 decimal is 4D2 hexadecimal
IPATCH>

TO RADIANS

Converts an angle from degrees to radians

FORMAT: **TO RADIANS degrees**

Command parameters:

degrees

The real number representing the desired angle in degrees.

DESCRIPTION:

The TO RADIANS command translates the given angle from degrees to radians, and prints it out.

Examples:

IPATCH> **TO RADIANS 89.0<CR>**
89.000 degrees is 1.553 radians
IPATCH>

TYPE

Types the current entry, or a sequence of entries

FORMAT: T [count]
 TYPE [count]
 TYPE *

Command parameters:

count

The number of entries to type out. This value defaults to 0, which means type the current entry only. If the value is greater than zero, then the current entry is not typed, but the next count entries are.

DESCRIPTION:

The TYPE command types out the contents of an entry.

If no argument is given, then the effect is as of TYPE 0, and the current entry is typed out. If an integer argument greater than zero is given, then each of the next count entries (but not the current entry) is typed out, and the last one is left current. If TYPE * is used, then the effect is as of specifying an argument great enough to TYPE to the end of file, which is left current.

The typed information is as follows:

- o RA - as for EXAMINE
- o HI - as for EXAMINE
- o MH - the map header in hexadecimal words (for type 2 MH entries) or ASCII (for type 3 or 4 MH entries)
- o MD - the map descriptor in hexadecimal words
- o CC - as for EXAMINE
- o CP - as for EXAMINE
- o ST - as for LIST
- o ZS - as for LIST
- o CB - as for LIST
- o all other entries are as for NEXT

Warning messages:

The following warning messages are specific to the TYPE command:

BADTYPE, argument to TYPE must be zero or greater
NEXTEOF, you can't NEXT off the end of the file
TYPEERR, TYPE command abandoned after 'integer' entries
TYPEONE, TYPE command abandoned after 1 entry

IFF ENTRIES

Descriptions, searching and editing

Display of values

The type of a numeric value is indicated by the way in which it is displayed. Thus an integer will be displayed in the minimum width possible, a hexadecimal number will always be 2, 4, or 8 digits, padded on the left with zeroes, and a real number will have a decimal point or be in engineering format.

Texts are displayed within double quotation marks (except within an EXPLAIN).

AC - Ancillary code entry

address: AC name (type) = value ["text"]

DESCRIPTION

AC entries are used to hold miscellaneous information. They are composed of a word (integer*2) type, a longword (integer*4, real, date, time, or character) value and an optional text.

The type of information held in an AC is determined by the AC type. Names and data types are associated with the numerical AC type by means of an FRT file (though some default types are always present). For a list of currently default AC types, and the method for adding user defined types via an FRT file, see the IFF and FRT User guides.

AC types may take values in the range 0-32767. Negative ACs are not allowed in normal use of IFF files, although IPATCH does not enforce this. The name of an AC (if defined in the FRT file) may be used in place of the numerical AC type. If the type is not present in the FRT file, then the name is displayed as ?, and the datatype is assumed to be integer.

AC values are printed in the format defined by the FRT file. A value may be 'absent'. The absent value is printed as ? (or "" for a text value). Values should be entered in the same format as that in which they are printed (including the absent value).

AC texts may be up to 255 characters in length. However, note that IPATCH itself can never change the length of a text.

SEARCHING

AC [values required]

If the search is to be qualified by the contents of the AC, then the relevant values may be specified. Any values not required may be wildcarded if necessary. The AC type may be specified either as an integer, or by using the associated name.

If the AC type is wildcarded, and the AC value is specified, then IPATCH has to decide the data type of the desired AC. Only real or integer values are allowed, and if a real is intended, then it must be unambiguous - it should contain a decimal point or be in E notation. For instance, AC * 100 will search for an integer value, but AC * 100.0 will search for a real. If a date, time, or character value is required, then the AC type may not be wildcarded.

If a specific text is being searched for, IPATCH will stop at the first AC entry which contains the requested text as a sub-string. Note that the case of the letters must match exactly. For instance, AC * * "UP & down" would match the AC

AC ? (1234) = 27 "This text goes UP & down"

EDITING

The AC type, value and text (if present) may all be changed independently or together. If the AC type is changed such that the data-type of the value (integer, real, date, time, character) should also change, but a new value is not requested, then in the case of conversion between integer and real, IPATCH will convert the value. In other cases, IPATCH leaves the value unchanged (i.e. keeps the same bit pattern), but checks whether the value is in the valid range for the new AC type. If not the value is changed to 'absent'. For instance:

```
00001243: AC Height (3) = 67.9 "This text belongs to a type 3 AC.."
IPATCH> / TYPE 2<CR>
%IPATCH-I-ACNOWINT, value has been converted from real to integer
00001243: AC Contour (2) = 68 "This text belongs to a type 3 AC.."
IPATCH> / TYPE 3<CR>
%IPATCH-I-ACNOWREAL, value has been converted from integer to real
```

```
00001243: AC Height (3) = 68.0 "This text belongs to a type 3 AC.."
```

Since IPATCH cannot add new data to the file, it is not possible to add text data to an AC that does not already contain it.

If a new text is specified, then if it is too long IPATCH will truncate it before using it, and if it is too short IPATCH will pad it to the right with spaces. For instance:

```
00001243: AC Height (3) = 68.0 "This text belongs to a type 3 AC.."
IPATCH> / TEXT "This text has been changed to a longer string"<CR>
%LSLLIB-E-STRTOOLONG, string is too long - truncated

00001243: AC Height (3) = 68.0 "This text has been changed to a lo"
IPATCH> / TEXT "This text is too short"<CR>

00001243: AC Height (3) = 68.0 "This text is too short          "
```

Prompt mode

The following prompts will be made:

```
AC type:
AC value (<data-type>):
AC text:
```

where <data-type> is either "integer", "real", "date", "time", or "character", depending on the AC type.

If a text is specified, it should not be enclosed in quotation marks. Note that this means that it is not possible to define a text of all spaces in prompt mode, as the 'trailing' spaces on a response are ignored. Use line mode or command mode for this purpose.

A reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the AC. Wildcards may be used as placeholders - for instance

```
IPATCH> / 2<CR>
```

will change the AC type, and

```
IPATCH> / * 80.0 "This is a new text"<CR>
```

changes the value and text field.

Command mode

/ TYPE integer

Sets the AC type.

/ VALUE value

Sets the AC value. The value must be of a datatype appropriate to the AC type.

/ TEXT "text"

Changes the AC text. It is not possible to insert a text longer than the text already present. The text inserted will be padded with spaces on the right if necessary.

CB - Coordinate block entry

address: CB nrow flags gtype ncol natt

DESCRIPTION

The CB entry contains the point data defining the feature. There may be more than one CB entry in a feature. The CB entry supercedes the ST and ZS entries. In addition to X and Y coordinates, it may contain other attributes (ncol in total) on a per-point basis, and also attributes (natt of them) considered fixed for the entire CB. Attributes may contain a value indicating 'absent' (equivalent to not present at all) and this is represented by a question mark '?', or for a character value "".

1. Each CB should contain no more than 200 points, so a new CB with the flags set to 1 (ie keep pen down when drawing to the start of this new CB) will be started for the remaining points.
2. If an invisible line is to be coded in the feature, then this is represented by starting a new CB with the flags set to 0 (ie keep pen up when moving to the start of this new CB).

Note that the pen is always considered to be up for the first CB in a feature - that is the pen is always kept up to move to the start of a new feature.

SEARCHING

CB [values required]

If the search is to be qualified by the contents of the CB, then the relevant values may be specified (in the order in which they are printed above). Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

Prompt mode is not supported for the CB entry.

Line mode

Line mode is not supported for the CB entry.

Command mode

- / CHANGE from to
Change the specified points. If only one number is given, then that one point will be changed. IPATCH will prompt for the new values. If there is only one point, then no argument is necessary.
- / DELETE from to
Deletes the specified points (rows). If only one number is given, then that one point will be deleted. If there is only one point, then no argument is needed.
- / FLAGS hex-value
Specifies a new value for the flags field of the CB. Currently only the bottom bit (bit 0) is significant, and is used to indicate pen up or down.
- / GTYPE value
Specifies a new value for the gtype field of the CB. This should be in the range 1-12 (see FRT User's Guide).
- / ATTRIBUTE name or code
Change the fixed attribute with the specified name or attribute code. IPATCH will prompt for a new name/code and value. The same comments as for AC entries apply if the datatype is changed but a new value is not specified.

CC - Cubic coefficients entry

address: CC

DESCRIPTION

The CC entry occurs once per section, and is a matrix of size (10,2) real numbers (indexed as for Fortran). It defines a transformation between two coordinate systems to be used by a post-processing program to transform all points in the file into the same coordinate space. This is necessary when (for instance) data has been digitised on a Lasertrak.

The matrix may be represented as

a	k
b	l
c	m
d	n
e	o
f	p
g	q
h	r
i	s
j	t

which represents the transformation

$$\begin{aligned} X' &= a + bX + cY + dXX + eXY + fYY + gXXX + hXXY + iXYY + jYYY \\ Y' &= k + lX + mY + nXX + oXY + pYY + qXXX + rXXY + sXYY + tYYY \end{aligned}$$

The unit matrix specifies a unit transformation - ie no transformation is necessary. The unit matrix has all terms 0.0, except for those at (2,1) and (3,2) which are 1.0. In the convention adopted by IPATCH these are the b and m terms.

SEARCHING

CC

It is not possible to modify the search by the CC contents.

EDITING

Prompt mode

Prompt mode is not supported for the CC entry.

Line mode

Line mode is not supported for the CC entry.

Command mode

/ letter value

'letter' specifies one of the 20 matrix values, from A to T. 'value' then specifies a real number to replace that position in the CC matrix.

/ UNIT

The UNIT command resets the CC matrix to the unit transformation. All entries are set to 0.0, except B and M which are set to 1.0. The unit transformation is the norm for files which have been digitised by non-distorting hardware (for instance a digitising table via DIGSYS).

CH - Literal character entry

address: CH "text"

DESCRIPTION

This entry is obsolete, and should not be used in new IFF files.

Its only current use is for storing the data in Laseraid patch files.

Note that CH entries must be outside features, and may also be outside layers (Laseraid patch files do not contain layers or features).

SEARCHING

CH ["text"]

If a specific text is being searched for, IPATCH will stop at the first CH entry which contains the requested text as a sub-string. Note that the case of the letters must match exactly. For instance, CH "UP & down" would match the CH

CH "This text goes UP & down"

EDITING

If a new text is specified, then if it is too long IPATCH will truncate it before using it, and if it is too short IPATCH will pad it to the right with spaces. For instance:

```
00001243: CH "This text belongs to a CH entry"
IPATCH> / TEXT "This text has been changed to a longer string"<CR>
%LSLLIB-E-STRTOOLONG, string is too long - truncated
00001243: CH "This text has been changed to a"
IPATCH> / TEXT "This text is too short"<CR>
00001243: CH "This text is too short"
```

Prompt mode

The following prompt will be made:

Text:

If a text is specified, it should not be enclosed in quotation marks. Note that this means that it is not possible to define a text of all spaces in prompt mode, as the 'trailing' spaces on a response are ignored. Use line mode or command mode for this purpose.

A reply of <CR> will leave the text unaltered.

Line mode

The / may also be followed immediately by the new text to be inserted into the CH - for instance

```
IPATCH> / "This is a new text"<CR>
```

Command mode

/ TEXT "text"

Replaces the text for the entry with the specified string. If the string is longer than that already in the file, it will be truncated. If the string is shorter than that already in the file, it will be padded with spaces on the right.

CP - Control points entry

address: CP

DESCRIPTION

The CP entry occurs once for each section. It defines the control points for the section, in both original (input) and destination (output) space.

SEARCHING

CP

It is not possible to modify the search by the CP contents.

EDITING

Prompt mode

Prompt mode is not supported for the CP entry.

Line mode

Line mode is not supported for the CP entry.

Command mode

/ BOTH

The user is prompted for replacement values for each of the original control points (the 'left hand side' control points), and then for replacement values for the target control points (the 'right hand side' control points). A response of <CR> to a prompt tells the program to retain the original value.

/ REPEAT

If the previous CP edit operation was /ROTATE or /TARGET then it will be repeated on the current CP entry. Care should be taken not to unintentionally repeat the /ROTATE command on the same CP entry.

/ ROTATE corner

The control points (both original and target) are rotated so that the named corner is at the north-west position. The corners (in standard Laser-Scan order) are NW,SW,SE,NE.

/ TARGET

The user is prompted for replacement values for each of the target control points (the 'right hand side' control points). A response of <CR> to a prompt tells the program to retain the original value.

CS - Character size entry

address: CS height spacing

DESCRIPTION

This entry is obsolete, and should not be used in new IFF files.

SEARCHING

CS [values required]

If the search is to be qualified by the contents of the CS, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

The following prompts will be made:

Height:
Spacing

Note that a reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the CS. Wildcards may be used as placeholders - thus

IPATCH> / 2 4<CR>

will change both the height and the spacing, but

IPATCH> / * 8<CR>

changes only the spacing.

Command mode

Command mode is not supported for the CS entry.

EF - End of feature entry

address: EF

DESCRIPTION

The EF entry flags the end of a feature, and balances the NF entry at the start of the feature.

SEARCHING

EF

The EF entry does not have any contents, so has no search modifiers.

EDITING

It is not possible to edit the EF entry.

EJ - End of job (data) entry

address: EJ

DESCRIPTION

The EJ entry should always be the last entry in the IFF file. It signifies the end of data, and is used to provide a tidy end to the file, rather than requiring programs to detect the end-of-file explicitly.

SEARCHING

EJ

The EJ entry does not have any contents, so has no search modifiers.

EDITING

It is not possible to edit the EJ entry.

EM - End of map entry

address: EM

DESCRIPTION

The EM entry is the last entry of a map or sheet, and flags its end. It should be the last entry before the EJ entry.

Historically, IFF files were permitted to contain several maps, and the EM entry was then required to mark the end of each.

SEARCHING

EM

The EM entry does not have any contents, so has no search modifiers.

EDITING

It is not possible to edit the EM entry.

EO - End of layer (overlay) entry

address: EO

DESCRIPTION

The EO entry flags the end of a layer, and balances the matching NO entry at the start of the layer. That NO entry should contain a pointer field which holds the address of this EO - this allows fast chaining through a file when particular layers are being ignored.

Note that some entries are legal between an EO and an NO - particularly the new section entries (NS,CC,CP) and also the (obsolete) CH entry.

SEARCHING

EO

The EO entry does not have any contents, so has no search modifiers.

EDITING

It is not possible to edit the EO entry.

FS - Feature status entry

address: FS fc status pc/text user-word

DESCRIPTION

The FS entry contains data which describes the feature containing it. It should be the first entry after the NF entry.

The FS entry includes the feature code in word 1, which is the primary descriptive code for a feature. The second word contains flag data used by Laseraid and its post-processing programs. The third word states whether this feature is text, symbol or line, and also holds either a process code or a description of the type of text. The fourth word is reserved for use by users, although beware that not all processing programs will preserve this word from input to output.

SEARCHING

FS [values required]

If the search is to be qualified by the contents of the FS, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

See also the TEXT, SYMBOL and LINE commands, which can be used to search for an FS entry with the relevant bit set in word 3.

EDITING

Prompt mode

The following prompts will be made:

```
FC:
status flags      (hex):
text/symbol/PC word (hex):
user dependent word (hex):
```

A reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the FS. Wildcards may be used as placeholders - for instance

IPATCH> / 2<CR>

will change the feature code, and

```
IPATCH> / * * 8012 1234<CR>
```

changes the text/symbol/PC word and the user specific word.

Command mode

editing word 1:

```
/ FC integer
    Sets the feature code to an integer in the range 0-32767
```

editing word 2:

```
/ STATUS hex-value
    Sets the whole of word 2 to the given value.
```

The individual bits of the status word may be set to 0 or 1 using the following commands. Consult the LAPROCESS documentation for the meaning of the bits.

```
/ CLOSED bit
/ NOTEDGE bit
/ REVERSE bit
/ TWOPART bit
/ PAINTOUT bit
/ SQUARE bit
/ INVERSE bit
/ SUPPRESS bit
```

editing word 3:

```
/ TEXT
    Sets the text bit, clears the symbol bit - the feature is thus
    flagged as a text.
```

```
/ SYMBOL
    Sets the symbol bit, clears the text bit - the feature is thus
    flagged as a symbol.
```

```
/ LINE
    Clears both the text and symbol bits - the feature is thus
    flagged as a linear feature.
```

```
/ POSITION integer
    Sets the text position to a value in the range 0-8. The number
    encodes the point at which the text's position was digitised,
    relative to a rectangle enclosing the text. This command is
    only allowed if the text bit is set.
```

```
/ TYPE_STYLE integer
    Sets the type style to a value in the range 0-3. This command
    is only allowed if the text bit is set.
```

/ CATEGORY integer

Sets the text's name category to a value in the range 0-63.
This command is only allowed if the text bit is set.

/ PC integer

Sets the process code. This command is only allowed if the text
bit is not set.

editing any word:

/ WORD word hex-value

Sets the the specified word to the given hexadecimal value.

HI - History entry

address: HI

DESCRIPTION

The HI record contains a history of which programs have been run on the current data, when they were run, and whether they succeeded. It thus provides a history of how the current file was produced, which can be very useful in trying to track down problems.

The HI entry occurs once at the beginning of the file, after the RA entry.

SEARCHING

HI

It is not possible to modify the search by the HI contents.

EDITING

It is not possible to edit the HI entry. The entry is intended as a record of a file's lifetime, so editing it to change the record of that lifetime is not allowed.

JB - Junction block entry

address: JB

DESCRIPTION

The JB entry defines a series of junctions.

Within each sector of the IFF file, a chain of JB entries is maintained to hold the details of all junctions within that sector.

Each JB entry contains the number of the sector that it is in, a pointer to the next JB entry in this JB chain, and details of each junction.

For each junction, the JB contains

- o the number of arms at that junction
- o the X,Y position of the junction
- o the address of each ST corresponding to a junction arm
- o the number of the vertex (either 1 or the index of the last point in the ST) corresponding to the junction in that ST

See also the JP (Junction Pointer) and SH (Sector Header) entries.

SEARCHING

JB

It is not currently possible to modify the search by the JB contents.

EDITING

Editing of junction entries is not currently supported.

JP - Junction pointer entry

address: JP

DESCRIPTION

The JP entry is a pointer back to a JB entry. A JP entry is inserted before or after (as appropriate) each ST that starts or ends at a junction.

The point in the ST corresponding to the junction is either the first (in which case the JP occurs before the ST) or the last (the JP occurs after the ST).

Each JP entry contains the address of the relevant JB entry, and the sequence number of the junction within that JB.

SEARCHING

JP

It is not currently possible to modify the search by the JP contents.

EDITING

Editing of junction entries is not currently supported.

MD - Map descriptor entry

address: MD version size

DESCRIPTION

The MD entry contains data describing the origin, projection and coordinate system of the IFF file. It occurs once at the beginning of the file, after the MH entry.

An unset map descriptor has its first word set to -1 (or FFFF in hex).

SEARCHING

MD

It is not possible to modify the search by the MD contents.

EDITING

It is not possible to edit the MD entry using IPATCH. The map descriptor describes the current origin, projection and units of the IFF file, and a change to the map descriptor must thus be accompanied by the appropriate changes to all point data in the file. The IMP utility ITRANS is provided to perform such transformations, and it will modify the map descriptor correctly.

MH - Map header entry

address: MH customer size

DESCRIPTION

The MH entry contains customer specific data about the IFF file. The customer number determines which format of data is present.

SEARCHING

MH

It is not possible to modify the search by the MH contents.

EDITING

IPATCH supports editing of the map header's customer number. For editing the actual value within the map header itself, use the relevant customer specific map header editing program - that is:

- | | | |
|---|--|-----------------------|
| 0 | map header is unset | |
| 1 | MCE specific map header | - edit with MCEHED |
| 2 | OSGB specific type 2 map header | - edit with OMHED |
| 3 | OSGB specific type 3 (OSTF) map header | - edit with OS editor |
| 4 | OSGB specific type 4 (CITF) map header | - edit with OS editor |

MCEHED is a utility supplied with the customer-specific package MCE. OSMHED is a utility in the IFFOSTF module of the CONVERT package.

Type 3 and 4 OS map headers are intended for use by Ordnance Survey (Great Britain) only and will be edited using OS developed header editors.

The map header occurs once in the file, after the HI entry. Note that historically IFF files could contain more than one map, and such maps each started with a separate map header, and ended with an EM entry.

Prompt mode

Prompt mode is not supported for the MH entry.

Line mode

Line mode is not supported for the MH entry.

Command mode

/ CUSTOMER integer

Changes the customer number in the descriptor of the map header to be the given integer. Note that this will change the interpretation of any data in the map header by utility programs.

NF - New feature entry

address: NF fsn isn

DESCRIPTION

The NF entry starts a new feature. It contains two identifying numbers, both in the range 0-65535.

The feature serial number (FSN) is generally used as the 'name' of the feature. There are three main conventions about when or whether FSNs are unique:

1. FSNs unique within the IFF file. This mechanism has traditionally been used by OS, and may be used by other customers as well.
2. FSNs unique within a particular layer. Thus a feature could be identified by its layer and FSN. This mechanism has traditionally been used by MCE.
3. FSNs not unique. The main example of this now is in TRI and SRI files, where single symbols may be made up of multiple features, all with the same FSN. In these cases the FSN is the ASCII code of the text, or the identifying number of the symbol.

Note that historically FSN 0 was considered special by many programs. It was used to flag an empty feature used as a place-holder in the IFF file, generally to hold TCs which applied to the entire layer. Thus it was possible to come across multiple FSN 0s even when FSNs are notionally unique within a file. In modern IFF files, the trailing TCs are no longer used.

The internal sequence number (ISN) is unique within the IFF file, and can be used as a single unique identifier for a feature. It may, but need not, be the same as the FSN. The ISN is generally assigned starting at 1 and incremented as a file is digitised.

SEARCHING

NF [values required]

If the search is to be qualified by the contents of the NF, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

The following prompts will be made:

FSN:
ISN:

A reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the NF. A wildcard may be used as a placeholder - for instance

IPATCH> / 2 2<CR>

will change the feature serial number and the internal sequence number, but

IPATCH> / * 80<CR>

just changes the internal sequence number

Command mode

/ FSN integer

Sets the feature serial number. The value may be in the range 0-65535.

/ ISN integer

Sets the internal sequence number. The value may be in the range 0-65535. Note that many programs use the internal sequence number as a unique identifier for features, and therefore care should be taken when editing this field.

NO - New layer (overlay) entry

address: NO layer status EO-address

DESCRIPTION

The NO entry starts a layer. It contains the number identifying this layer, and a status word. It may also contain the address of the corresponding EO entry.

IFF files are generally divided up into multiple layers, where data of a common sort or source is grouped in the same layer. Note that a layer may be split into several parts, identified by all having the same layer number.

Layer 0 is conventionally reserved for 'non-essential' data - for instance a grid, symbols at control points, or MCE DFAD accuracy polygons. The assumption is that the IFF file would not be significantly degraded in terms of information content by throwing layer 0 away. If layer 0 is present, some programs expect it to be the first layer in the file.

Historically, layers 11 and 32 were also used for the same sort of purpose as layer 0.

The status flag is not currently used, and should be set to zero.

If the EO pointer field is present, it should contain the hexadecimal address of the EO entry which matches this NO. This is used by programs which must ignore layers to 'jump' from the NO to the EO. Thus take great care when editing the EO pointer field, as an incorrect value could cause a processing program to abort.

SEARCHING

NO [values required]

If the search is to be qualified by the contents of the NO, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

The following prompts will be made:

Layer:
Status:

and if the NO entry contains an EO address field, then

EO pointer:

A reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the NO. Wildcards may be used as placeholders - for instance

IPATCH> / 2<CR>

will change the layer, and

IPATCH> / 5 * 8013<CR>

changes the layer and the EO pointer field.

Command mode

/ LAYER integer

Sets the layer (overlay) to a new value. The number is in the range 0-32767.

/ STATUS hex-value

Sets the second word of the NO entry. Note that this entry is not currently used or transferred by Laser-Scan programs. The value is conventionally zero.

/ POINTER hex-address

Sets the pointer to the next EO entry, which terminates this layer. This field can only be changed if the NO entry contained it already - older IFF files contain NO entries without the EO pointer field.

NS - New section entry

address: NS "text"

DESCRIPTION

The NS entry is used to flag the start of a new digitising session. The text written in it conventionally contains the initials of the digitising operator, the program being used and the date and time at which digitising was started.

The entry is necessary because IFF files are often digitised in several sessions. The NS entry will be followed by a CC and a CP entry, defining the transformations and control points applying from now on. Post-processing programs used to transform all data into the same coordinate system will normally also reduce the whole IFF file down to one section.

The NS entry will follow either an MD entry (for the first section) or an EO entry (for later sections).

The maximum number of characters in the text is 255, but note that IPATCH cannot itself change the length of the text string.

SEARCHING

NS ["text"]

If a specific text is being searched for, IPATCH will stop at the first NS entry which contains the requested text as a sub-string. Note that the case of the letters must match exactly. For instance, NS "UP & down" would match the NS

NS "This text goes UP & down"

EDITING

If a new text is specified, then if it is too long IPATCH will truncate it before using it, and if it is too short IPATCH will pad it to the right with spaces. For instance:

```
00001243: NS      "This text belongs to an NS entry"
IPATCH> / TEXT "This text has been changed to a longer string"<CR>
%LSLLIB-E-STRTOOLONG, string is too long - truncated
00001243: NS      "This text has been changed to a "
IPATCH> / TEXT "This text is too short"<CR>
00001243: NS      "This text is too short          "
```

Prompt mode

The following prompt will be made:

Text:

If a text is specified, it should not be enclosed in quotation marks. Note that this means that it is not possible to define a text of all spaces in prompt mode, as the 'trailing' spaces on a response are ignored. Use line mode or command mode for this purpose.

A reply of <CR> will leave the text unaltered.

Line mode

The / may also be followed immediately by the new text to be inserted into the NS - for instance

IPATCH> / "This is a new text"<CR>

Command mode

/ TEXT "text"

Replaces the text for the entry with the specified string. If the string is longer than that already in the file, it will be truncated. If the string is shorter than that already in the file, it will be padded with spaces on the right.

RA - Range entry

address: RA minX maxX minY maxY

DESCRIPTION

The RA entry records the maximum extent of the data in the IFF file. It is used by plot and display programs to work out whether to clip the file, and what scale it can be displayed at.

The range entry is always the first entry in the IFF file.

SEARCHING

RA [values required]

If the search is to be qualified by the contents of the RA, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

The following prompts will be made:

Min X:
Max X:
Min Y:
Max Y:

A reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the RA. Wildcards may be used as placeholders - thus

IPATCH> / 200.0 0.0 300.0 0.0<CR>

will change the entire RA, but

IPATCH> / * 80.0<CR>

just changes the minimum X value.

Command mode

/ MAXX real-value
Changes the maximum X value in the range entry to the specified real.

/ MINX real-value
Changes the minimum X value in the range entry to the specified real.

/ MAXY real-value
Changes the maximum Y value in the range entry to the specified real.

/ MINY real-value
Changes the minimum Y value in the range entry to the specified real.

RO - Rotation entry

address: RO angle

DESCRIPTION

The RO entry defines the angle at which an oriented symbol or a text is to be drawn. The angle is in radians, measured counterclockwise from the positive horizontal axis.

SEARCHING

RO [angle]

If the search is to be qualified by the contents of the RO, then the angle may be specified.

EDITING

Prompt mode

The following prompt will be made:

Angle:

A reply of <CR> will leave the angle unaltered.

Line mode

The / may also be followed immediately by the new angle to be inserted into the RO - for instance

IPATCH> / 2.0<CR>

Command mode

/ ANGLE radians

Changes the angle stored in the RO entry to the given number of real radians.

SH - Sector header entry

address: SH

DESCRIPTION

The SH entry contains the addresses of the start of the JB chain for each sector in the IFF file.

When an IFF file contains junction structure, the map is divided into rectangular sectors. A chain of JB entries is maintained for each sector, containing the junction information. The SH entry provides the address of the start of each of these chains.

SEARCHING

SH

It is not currently possible to modify the search by the SH contents.

EDITING

Editing of junction entries is not currently supported.

SL - Select symbol library entry

address: SL symbol-library

DESCRIPTION

This entry is obsolete, and should not be used in new IFF files.

SEARCHING

SL [symbol-library]

If the search is to be qualified by the contents of the SL, then the symbol library may be specified.

EDITING

Prompt mode

The following prompt will be made:

Integer:

A reply of <CR> will leave the value unaltered.

Line mode

The / may also be followed immediately by the new value to be inserted into the SL - for instance

IPATCH> / 2<CR>

Command mode

Command mode is not supported for the SL entry.

SS - Symbol select entry

address: SS symbol

DESCRIPTION

This entry is obsolete, and should not be used in new IFF files.

SEARCHING

SS [values required]

If the search is to be qualified by the contents of the SS, then the symbol may be specified.

EDITING

Prompt mode

The following prompt will be made:

Integer:

A reply of <CR> will leave the value unaltered.

Line mode

The / may also be followed immediately by the new value to be inserted into the SS - for instance

IPATCH> / 2<CR>

Command mode

Command mode is not supported for the SS entry.

ST - 2-dimensional point string entry

address: ST npts pen

DESCRIPTION

The ST entry contains the point data defining the feature. There may be more than one ST entry in a feature.

1. Each ST should contain no more than 200 points, so a new ST with the pen flag set to 1 (ie keep pen down when drawing to the start of this new ST) will be started for the remaining points.
2. If an invisible line is to be coded in the feature, then this is represented by starting a new ST with the pen flag set to 0 (ie keep pen up when moving to the start of this new ST).

Note that the pen flag is always considered to be 0 for the first ST in a feature - that is the pen is always kept up to move to the start of a new feature.

SEARCHING

ST [values required]

If the search is to be qualified by the number of points or pen code of the ST, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

Prompt mode is not supported for the ST entry.

Line mode

Line mode is not supported for the ST entry.

Command mode

/ CHANGE from to

Change the specified points. If only one number is given, then that one point will be changed. IPATCH will prompt for the new values. If there is only one point, then no argument is necessary.

/ DELETE from to

Deletes the specified points. If only one number is given, then that one point will be deleted. If there is only one point, then no argument is needed.

/ PEN_POSITION hex-value

Specifies a new value for the pen up/down field of the ST. Note that IFFLIB will truncate this value to 3 bits (ie the range 0-7). Currently only the bottom bit (bit 0) is significant, and is used to indicate pen up or down.

TC - Transmitted comment entry

address: TC "text"

DESCRIPTION

A TC entry is used to label the following feature. It is used by MCE to hold specialised plotting instructions.

Interpretation of a TC requires the IFF file to be regarded as a sequential file, since the TC applies either to the following feature, or to the layer as a whole (in which case it appears after the last feature - in old format IFF files before a special empty feature with FSN 0).

The TC entry is not allowed inside a feature, but must be inside a layer.

The maximum number of characters in a TC is 255, but note that IPATCH cannot itself change the length of that text.

SEARCHING

TC ["text"]

If a specific text is being searched for, IPATCH will stop at the first TC entry which contains the requested text as a sub-string. Note that the case of the letters must match exactly. For instance, TC "UP & down" would match the TC

TC "This text goes UP & down"

EDITING

If a new text is specified, then if it is too long IPATCH will truncate it before using it, and if it is too short IPATCH will pad it to the right with spaces. For instance:

```
00001243: TC "This text belongs to a TC entry"
IPATCH> / TEXT "This text has been changed to a longer string"<CR>
%LSLLIB-E-STRTOOLONG, string is too long - truncated
00001243: TC "This text has been changed to a"
IPATCH> / TEXT "This text is too short"<CR>
00001243: TC "This text is too short"
```

Prompt mode

The following prompt will be made:

Text:

If a text is specified, it should not be enclosed in quotation marks. Note that this means that it is not possible to define a text of all spaces in prompt mode, as the 'trailing' spaces on a response are ignored. Use line mode or

command mode for this purpose.

A reply of <CR> will leave the text unaltered.

Line mode

The / may also be followed immediately by the new text to be inserted into the TC - for instance

```
IPATCH> / "This is a new text"<CR>
```

Command mode

/ TEXT "text"

Replaces the text for the entry with the specified string. If the string is longer than that already in the file, it will be truncated. If the string is shorter than that already in the file, it will be padded with spaces on the right.

TH - Text height / line thickness entry

address: TH integer

DESCRIPTION

The TH entry is used

1. to hold the line thickness for displays which are capable of displaying multiple thickness lines
2. to hold the size of text for a text feature. In this case the integer is interpreted as either:
 - o the height of the text in hundredths of a millimetre
 - o the size of the text as a point size

SEARCHING

TH [height/thickness]

If the search is to be qualified by the contents of the TH, then the relevant value may be specified.

EDITING

Prompt mode

The following prompt will be made:

Thickness/height:

A reply of <CR> will leave the value unaltered.

Line mode

The / may also be followed immediately by the new value to be inserted into the TH - for instance

IPATCH> / 20<CR>

Command mode

/ HEIGHT integer

Changes the value stored in the TH entry to the given integer.

/ THICKNESS integer

Changes the value stored in the TH entry to the given integer.

TS - Text status entry

address: TS tcc reserved textcode reserved

DESCRIPTION

The TS entry introduces a text component, and contains data which describes the text component following it.

Text features may contain one text string, with associated location and descriptive data, or they may be **composite** - that is composed of several sub-texts or text components, which may be manipulated independently or as a single entity.

Each text component starts with a TS entry, and ends with the next TS entry, or the final EF of the feature. The first TS entry occurs immediately after the FS entry and any AC entries. Text components may not include FS or AC entries, but may contain any other entries that are legal within a normal text feature.

Word 1 of the TS entry is the text component code (TCC), which is the primary descriptive code for a text component - it is effectively the feature code for this component of the composite text, and is used in the same manner.

Word 3 of the TS entry is the text word, and is identical in form to word 3 of a text's FS entry - it holds a description of the type of text component. Note that the top two bits (what would be the text/symbol bits in an FS) should always be set to '10' binary, as they would in an FS entry. These bits are referred to as redundancy bits, since they are not strictly necessary.

The second and fourth words are reserved for later definition, and should always be zero.

SEARCHING

TS [values required]

If the search is to be qualified by the contents of the TS, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

The following prompts will be made:

TCC :
word 2 (hex):
text word (hex):
word 4 (hex):

A reply of <CR> will leave the relevant value unaltered.

Line mode

The / may also be followed immediately by the new values to be inserted into the TS. Wildcards may be used as placeholders - for instance

```
IPATCH> / 2<CR>
```

will change the text component code, and

```
IPATCH> / * * 8012<CR>
```

changes the text word.

Command mode

editing word 1:

```
/ TCC integer
    Sets the text component code to an integer in the range 0-32767
```

editing word 3:

```
/ POSITION integer
    Sets the text position to a value in the range 0-8. The number
    encodes the point at which the text's position was digitised,
    relative to a rectangle enclosing the text.
```

```
/ TYPE_STYLE integer
    Sets the type style to a value in the range 0-3.
```

```
/ CATEGORY integer
    Sets the text's name category to a value in the range 0-63.
```

editing any word:

```
/ WORD word hex-value
    Sets the the specified word to the given hexadecimal value.
```

TX - Text entry

address: TX "text"

DESCRIPTION

The TX entry holds the text for a text feature. It should be the last entry before the EF entry.

The maximum number of characters in a TX is 255, but note that IPATCH cannot itself change the length of a text.

SEARCHING

If a specific text is being searched for, IPATCH will stop at the first TX entry which contains the requested text as a sub-string. Note that the case of the letters must match exactly. For instance, TX "UP & down" would match the TX

TX "This text goes UP & down"

EDITING

If a new text is specified, then if it is too long IPATCH will truncate it before using it, and if it is too short IPATCH will pad it to the right with spaces. For instance:

```
00001243: TX "This text belongs to a TX entry"
IPATCH> / TEXT "This text has been changed to a longer string"<CR>
%LSLLIB-E-STRTOOLONG, string is too long - truncated
00001243: TX "This text has been changed to a"
IPATCH> / TEXT "This text is too short"<CR>
00001243: TX "This text is too short"
```

Prompt mode

The following prompt will be made:

Text:

If a text is specified, it should not be enclosed in quotation marks. Note that this means that it is not possible to define a text of all spaces in prompt mode, as the 'trailing' spaces on a response are ignored. Use line mode or command mode for this purpose.

A reply of <CR> will leave the text unaltered.

Line mode

The / may also be followed immediately by the new text to be inserted into the TX - for instance

IPATCH> / "This is a new text"<CR>

Command mode

/ TEXT "text"

Replaces the text for the entry with the specified string. If the string is longer than that already in the file, it will be truncated. If the string is shorter than that already in the file, it will be padded with spaces on the right.

VO - Void entry

address: VO size

DESCRIPTION

The VO entry is used to replace a series of deleted IFF entries. Since it is not possible to 'compress' an IFF file, a deleted entry or series of entries is overwritten with a void area of the requisite size.

SEARCHING

VO [size]

The search may optionally be qualified by the size of the VO entry required, as it would be reported by IPATCH.

EDITING

It is not possible to edit a VO entry.

XX - Invalid entry

address: XX size

DESCRIPTION

IFFLIB returns the entry mnemonic XX when it has found an entry that it cannot identify as a valid IFF entry. This generally occurs because either the IFF file is corrupt in some manner, or because the FIND command has been used to position to an address which does not appear to contain an entry.

SEARCHING

It is not possible to search for an XX entry.

EDITING

It is not possible to edit an XX entry.

ZS - 3-dimensional point string entry

address: ZS npts pen

DESCRIPTION

The ZS entry contains the three-dimensional point data defining the feature. There may be more than one ZS entry in a feature:

1. Each ZS should contain no more than 200 points, so a new ZS with the pen flag set to 1 (ie keep pen down when drawing to the start of this new ZS) will be started for the remaining points.
2. If an invisible line is to be coded in the feature, then this is represented by starting a new ZS with the pen flag set to 0 (ie keep pen up when moving to the start of this new ZS).

Note that the pen flag is always considered to be 0 for the first ZS in a feature - that is the pen is always kept up to move to the start of a new feature.

SEARCHING

ZS [values required]

If the search is to be qualified by the number of points or pen code of the ZS, then the relevant values may be specified. Any values not required may be wildcarded if necessary.

EDITING

Prompt mode

Prompt mode is not supported for the ZS entry.

Line mode

Line mode is not supported for the ZS entry.

Command mode

/ CHANGE from to

Change the specified points. If only one number is given, then that one point will be changed. IPATCH will prompt for the new values. If there is only one point, then no argument is necessary.

/ DELETE from to

Deletes the specified points. If only one number is given, then that one point will be deleted. If there is only one point, then no argument is needed.

/ PEN_POSITION hex-value

Specifies a new value for the pen up/down field of the ZS. Note that IFFLIB will truncate this value to 3 bits (ie the range 0-7). Currently only the bottom bit (bit 0) is significant, and is used to indicate pen up or down.

EXAMPLE EDITING SESSION

```
$ ipatch winniethepooh/write/nostatus
00000000: Start of file
IPATCH> NEXT<CR>
00000001: RA    -225.114    9241.125    -0.145    9081.765
IPATCH> /MINX 200.01<CR>
00000001: RA    200.010    9241.125    -0.145    9081.765
IPATCH> TYPE 1<CR>
0000000A: HI
History entry
- size of entry = 4001 words
Date          Time  Username      Program  Function  Elapsed  CPU      STATUS
18-JUL-1986  17:16  TIM          IFROMTEXT Create  00:01:41 00:00:47 00000001
25-JUL-1986  17:28  TONY         IPATCH   Patch   00:09:45 00:00:11 00000001
IPATCH> NEXT 2<CR>
0000110D: MD      2      200
IPATCH> /<CR>
%IPATCH-W-NOEDIT, MD entries cannot be edited
0000110D: MD      2      200
IPATCH> EXPLAIN<CR>
0000110D: MD      2      200
Map descriptor entry
- size is 200 words
- map descriptor version 2
- Local origin:      0.0000,      90.0000
- Map scale:      1000.0000
- Projection:      100 (Geographic (ie Lat and Long))
- Spheroid:      0 (Clarke 1866)
- Units:      5 (radians)
* use the TYPE command to type the contents of the map descriptor in *
* hexadecimal, or use the ITRANS utility to edit the map descriptor *
IPATCH> NEXT<CR>
000011D6: NS  "created by IFROMTEXT at 15:39:49 on 30-OCT-85"
IPATCH> /<CR>
Text:  This NS has been edited<CR>
000011D6: NS  "This NS has been edited"
IPATCH> /TEXT "Fred"<CR>
000011D6: NS  "Fred"
IPATCH> NEXT<CR>
000011EC: CC
IPATCH> /<CR>
%IPATCH-W-EDITCC, use the form 'letter' 'value' to change CC entries
000011EC: CC
IPATCH> EXPLAIN<CR>
000011EC: CC
Cubic coefficients entry
- size of entry = 40 words
- matrix of the form:
      a = .000000E 000      k = .000000E 000
      b = .100000E 001      l = .000000E 000
      c = .000000E 000      m = .100000E 001
      d = .000000E 000      n = .000000E 000
      e = .000000E 000      o = .000000E 000
      f = .000000E 000      p = .000000E 000
```



```

        g = .000000E 000          q = .000000E 000
        h = .000000E 000          r = .000000E 000
        i = .000000E 000          s = .000000E 000
        j = .000000E 000          t = .000000E 000
IPATCH> /A 10.0<CR>
000011EC: CC
IPATCH> TYPE<CR>
000011EC: CC
Cubic coefficients entry
- size of entry = 40 words
- matrix of the form:
        a = .100000E 002          k = .000000E 000
        b = .100000E 001          l = .000000E 000
        c = .000000E 000          m = .100000E 001
        d = .000000E 000          n = .000000E 000
        e = .000000E 000          o = .000000E 000
        f = .000000E 000          p = .000000E 000
        g = .000000E 000          q = .000000E 000
        h = .000000E 000          r = .000000E 000
        i = .000000E 000          s = .000000E 000
        j = .000000E 000          t = .000000E 000
IPATCH> NEXT<CR>
00001215: CP
IPATCH> /<CR>
%IPATCH-W-CPEDIT, to edit a CP use /TARGET, /BOTH, /ROTATE or /REPEAT
00001215: CP
IPATCH> EXPLAIN<CR>
00001215: CP
Control point entry
- size of entry = 32 words
      corner    original          target
      NW        400.000,    0.000    400.000,    0.000
      SW        400.000,    400.000    400.000,    400.000
      SE         0.000,    400.000     0.000,    400.000
      NE         0.000,     0.000     0.000,     0.000
IPATCH> /ROTATE SE<CR>
00001215: CP
IPATCH> TYPE<CR>
00001215: CP
Control point entry
- size of entry = 32 words
      corner    original          target
      NW         0.000,    400.000     0.000,    400.000
      SW         0.000,     0.000     0.000,     0.000
      SE        400.000,     0.000    400.000,     0.000
      NE        400.000,    400.000    400.000,    400.000
IPATCH> /TARGET<CR>
Now give a new value for each of the target control points
NW corner (was      0.0000,    400.0000) : 800, 0 <CR>
SW corner (was      0.0000,     0.0000) : 800, 800<CR>
SE corner (was     400.0000,     0.0000) : 0, 800<CR>
NE corner (was     400.0000,    400.0000) : 0, 0 <CR>
IPATCH> TYPE<CR>
00001215: CP
Control point entry
- size of entry = 32 words

```

corner	original	target
NW	0.000, 400.000	800.000, 0.000
SW	0.000, 0.000	800.000, 800.000
SE	400.000, 0.000	0.000, 800.000
NE	400.000, 400.000	0.000, 0.000

```

IPATCH> NEXT<CR>
00001236: NO      1    0000  00FFFFFF
IPATCH> EO<CR>
00001714: EO
IPATCH> FIND 1236<CR>
00001236: NO      1    0000  00FFFFFF
IPATCH> / * * 1714<CR>
00001236: NO      1    0000  00001714
IPATCH> NEXT<CR>
0000123B: NF      170      1
IPATCH> /<CR>
FSN: 171<CR>
ISN: 171<CR>
0000123B: NF      171      171
IPATCH> NEXT<CR>
0000123E: FS      123    0001    4000    0000
IPATCH> /TEXT<CR>
0000123E: FS      123    0001    8000    0000
IPATCH> /CATEGORY 4<CR>
0000123E: FS      123    0001    8100    0000
IPATCH> /POSITION 3<CR>
0000123E: FS      123    0001    8103    0000
IPATCH> /STATUS 0<CR>
0000123E: FS      123    0000    8103    0000
IPATCH> /WORD 4 512<CR>
0000123E: FS      123    0000    8103    0512
IPATCH> NEXT<CR>
00001243: AC Height (3) = 67.9  "This text belongs to a type 3 AC.."
IPATCH> /TYPE 2<CR>
%IPATCH-I-ACNOWINT, value has been converted from real to integer
00001243: AC Contour (2) = 68  "This text belongs to a type 3 AC.."
IPATCH> /TYPE 3<CR>
%IPATCH-I-ACNOWREAL, value has been converted from integer to real
00001243: AC Height (3) = 68.0  "This text belongs to a type 3 AC.."
IPATCH> NEXT<CR>
00001258: TH      30
IPATCH> / 25<CR>
00001258: TH      25
IPATCH> NEXT<CR>
0000125A: ST      4    0000
IPATCH> TYPE<CR>
0000125A: ST      4    0000
      1:      -0.5344,      14.8308
      2:       0.8506,      16.8394
      3:       2.2290,      18.8454
      4:       3.4641,      23.9534
IPATCH> NEXT 2<CR>
0000126B: EF
IPATCH> /<CR>
%IPATCH-W-NOEDIT, EF entries cannot be edited
0000126B: EF

```

```
IPATCH> NEXT<CR>
0000126C: NF      170      2
IPATCH> RO<CR>
00001F38: RO      0.000
IPATCH> /ANGLE 0.8<CR>
00001F38: RO      0.800
IPATCH> TO DEGREES 12.0<CR>
      0.800 radians is      45.837 degrees
IPATCH> NEXT 3<CR>
00001F56: TC  "This TC is outside a feature.."
IPATCH> NEXT<CR>
00001F66: NF      330      44
IPATCH> PREVIOUS<CR>
Scanning file to establish where we are
00001F56: TC  "\"This TC is outside a feature\""
IPATCH> /"Hello there"<CR>
00001F56: TC  "Hello there"
IPATCH> DISPLAY ON<CR>
%IPATCH-I-DISPLAYON, NEXT will now display intermediate entries
IPATCH> NEXT 15<CR>
00001F66: NF      330      44
00001F69: FS      123      0000      8080      0000
00001F6E: TH      30
00001F70: ST      1      0000
00001F75: EF
00001F76: TC  "\"This is a pointless line of text ....."
00001F8B: NF      422      45
00001F8E: FS      0      0000      0000      0000
00001F93: ST      4      0000
00001FA4: EF
00001FA5: NF      190      46
00001FA8: FS      5      0000      0000      0000
00001FAD: ST      1      0000
00001FB2: EF
00001FB3: CH  "\"This text came from a CH entry\""
IPATCH> /<CR>
Text:  <CR>
00001FB3: CH  "\"This text came from a CH entry\""
IPATCH> /TEXT "freddy"<CR>
00001FB3: CH  "freddy"
IPATCH> EIT<CR>
%LSLLIB-E-UNEXPCMD, unexpected 'EIT' found instead of command
IPATCH> EXIT<CR>
ELAPSED:  00:05:25.84 CPU:   0:00:05.71 BUFIO:   281 DIRIO:   46 FAULTS:  263
$
```

IPATCH messages

MESSAGES (INFORMATIONAL)

These messages provide information on the current state of the program, or follow from a more serious message to help explain it.

ACNOWINT, value has been converted from real to integer

Explanation: The current AC entry or CB fixed attribute has had its type changed. The original type specified a real value, but the new type specifies an integer value. The real value has been converted to an integer.

User action: None.

ACNOWREAL, value has been converted from integer to real

Explanation: The current AC entry or CB fixed attribute has had its type changed. The original type specified an integer value, but the new type specifies a real value. The integer value has been converted to a real.

User action: None.

COMMAND, commands are being read from 'file-spec'

Explanation: Both the /LOG and the /COMMANDS switches have been used in starting IPATCH. This message confirms which file commands will be read from.

User action: None.

DISPLAYOFF, NEXT will now only display the destination entry

Explanation: The DISPLAY OFF command has been given. This message confirms its action.

User action: None.

DISPLAYON, NEXT will now display intermediate entries

Explanation: The DISPLAY ON command has been given. This message confirms its action.

User action: None.

JOURNAL, commands are being journalled to 'file-spec'

Explanation: Both the /LOG and the /JOURNAL switches have been used in starting IPATCH. This message confirms which file commands will be journalled to.

User action: None.

LOOKFOR, when looking for entry 'entry', at address 'hex-address'

Explanation: This message follows a WARNING message reporting some error in positioning back to a known entry within the IFF file. It is intended to provide extra help in diagnosing the error.

User action: As for the WARNING message.

MARKED, current entry now marked

Explanation: The MARK command has been given, and this message confirms that the current entry has been marked successfully.

User action: None.

NOCHANGE, no change made to the entry

Explanation: This message is given after an error has occurred in the routines which prompt for new values for an ST or ZS. It indicates that all changes for that ST or ZS in this sequence of prompts have been thrown away.

User action: As indicated by the original error messages.

NOPTS, no points read in for this entry

Explanation: This message indicates that no points were read in for the current ST, ZS, or CB entry. The message follows the WARNING message MAXPTS.

User action: As for the MAXPTS message.

NOROTATE, no rotation - NW corner is already at the NW

Explanation: An attempt has been made to rotate the current CP entry so that the NW corner is at the NW. The program need not do anything, but points this out in case it was a mistake.

User action: None.

PARTPREV, PREVIOUS will only work over the first part of the file

Explanation: This message follows an error in scanning the NF entries in the IFF file, to allow backwards movement with the PREVIOUS command. The error has been such that the scan has not been extended successfully, so the PREVIOUS command will only be available over part of the file.

User action: As for the preceding WARNING message. In general, report this error to Laser-Scan, together with a copy of the IFF file, and a log of the commands obeyed by IPATCH before the error.

READINGFRT, reading FRT file 'file-spec'

Explanation: This message if output if the /LOG switch has been used, and an FRT file is specified.

User action: None.

READONLY, the IFF file is not open for writing

Explanation: This message is output if the /LOG switch has been used, and the /WRITE switch is absent.

User action: None.

REPORTLSL, please report this problem to Laser-Scan

Explanation: The previous WARNING message reflects a bug or problem in the IPATCH program, and should be reported to Laser-Scan.

User action: Report the error to Laser-Scan.

REPROTATE, repeating rotation to put 'corner' corner at NW

Explanation: This message is output by the CP edit command /REPEAT if the operation being repeated is a /ROTATE command.

User action: None.

REPTARGET, repeating setting of target values

Explanation: This message is output by the CP edit command /REPEAT if the operation being repeated is a /TARGET command.

User action: None.

Scanning file to establish where we are

Explanation: This message (which is not prefixed by the normal %IPATCH-x-mnemonic string) is output before IPATCH starts to scan the file for a PREVIOUS command which is beyond the current NF list. It should be taken as a warning that the program may be occupied for a while, depending upon the size of the file and the distance to be scanned.

User action: Wait.

WRITING, the IFF file is open for WRITE

Explanation: This message is output if the /LOG switch has been used, and the file is opened with the /WRITE switch.

User action: None.

MESSAGES (WARNING)

These messages indicate that something has gone wrong, or that the operation requested is not possible.

ACDRANGE, 'value' is outside valid range for 'name' ('code')
valid range is 'value' : 'value'

Explanation: The value entered for an AC or CB attribute value is outside the valid range for the attribute code.

User action: Specify a value within the valid range.

ACNOTEXT, AC does not have a text entry

Explanation: An attempt has been made to edit the text of the current AC entry, but the entry does not contain a text field. The edit is ignored.

User action: If a text must be added to an AC entry, use a program that can add data to an IFF file - for instance LITES2.

ACNOWABSENT, value has been set to absent because of datatype change

Explanation: The current AC entry or CB fixed attribute has had its type changed. The old value is not within the valid range for the new type, so the value has been set to absent.

User action: Unless the absent value is desired, then set a new value appropriate to the new datatype.

BADCORNER, there is no corner 'corner' - it should be NW,SW,SE or NE

Explanation: The corner name used for the CP edit command /ROTATE is not one of the four recognised corners. The command is ignored.

User action: Try again with one of the valid corner names, as listed in the error message.

BADDATE, bad date format "'string'"

Explanation: Dates must be given in the system date format dd-mmm-yyyy.

User action: Specify date correctly.

BADFIND, cannot FIND before 00000001 or after 'end-of-file' - nothing there

Explanation: The FIND command will only move to an address within the IFF file (after the start of file, and up to the end of file). The address given was outside that range. The command is ignored.

User action: Check the address given. Note that it should be a hexadecimal value.

BADNEXT, argument to NEXT must be greater than zero

Explanation: It is not possible to use NEXT to move backwards, or to stay on the same entry. Thus NEXT 0 or NEXT -1 are illegal commands.

User action: Use PREVIOUS to move backwards, ? to show the current entry again, or specify a positive value for the distance to NEXT forwards.

BADPREV, argument to PREVIOUS must be greater than zero

Explanation: It is not possible to use PREVIOUS to move forwards, or to stay on the same entry. Thus PREVIOUS 0 or PREVIOUS -1 are illegal commands.

User action: Use NEXT to move forwards, ? to show the current entry again, or specify a positive value for the distance to PREVIOUS backwards.

BADPRJ, unknown projection code 'integer'

Explanation: The EXPLAIN command is trying to explain the contents of the current MD entry, but it does not recognise the projection code. Either the current version of IPATCH is not up to date with respect to legal values in the MD, or the MD is corrupt.

User action: The utility ITRANS can be used to report on the MD entry more fully, and its documentation lists legal projection codes and their meanings. ITRANS should also be used to repair a corrupt MD entry, although the cause of the error should also be established.

BADREV, unknown revision level 'integer', must be -1, 0, or 1

Explanation: An illegal IFF input revision level has been specified, either by the /REVISION_LEVEL qualifier, or by the REVISION_LEVEL command.

User action: Specify correct revision level.

BADSPH, unknown spheroid code 'integer'

Explanation: The EXPLAIN command is trying to explain the contents of the current MD entry, but it does not recognise the spheroid code. Either the current version of IPATCH is not up to date with respect to legal values in the MD, or the MD is corrupt.

User action: The utility ITRANS can be used to report on the MD entry more fully, and its documentation lists legal spheroid codes and their meanings. ITRANS should also be used to repair a corrupt MD entry, although the cause of the error should also be established.

BADTIME, bad time format "'string'"

Explanation: Times must be given in the system time format hh:mm:ss.cc.

User action: Specify time correctly.

BADTYPE, argument to TYPE must be zero or greater

Explanation: It is not possible to use TYPE to move backwards. Thus TYPE -1 is an illegal command.

User action: Use PREVIOUS to move backwards. Use TYPE or TYPE 0 to type the current entry, or a value larger than zero to type the next several entries.

BADUNT, unknown units code 'integer'

Explanation: The EXPLAIN command is trying to explain the contents of the current MD entry, but it does not recognise the units code. Either the current version of IPATCH is not up to date with respect to legal values in the MD, or the MD is corrupt.

User action: The utility ITRANS can be used to report on the MD entry more fully, and its documentation lists legal units codes and their meanings. ITRANS should also be used to repair a corrupt MD entry, although the cause of the error should also be established.

BINCHOPERR, internal error looking for start of current feature

Explanation: When IPATCH moves the current position to an entry within the list of addresses which it has scanned for the PREVIOUS list, it looks for the start of the feature containing this entry. This error means that the binary chop that it uses to perform that search has failed - the top and bottom limits have crossed over each other.

User action: Unless something obviously strange has already occurred earlier in the IPATCH session, which would quite sensibly have corrupted the internal list of NF positions, then report this error to Laser-Scan with as much detail as possible (eg the file and a log of the commands obeyed).

CANTEDIT, you cannot edit the 'string' of the file

Explanation: 'string' will be either "start" or "end". The start of the IFF file (address 00000000) and the end of the IFF file are not entries, so cannot be edited. Thus the preceding / command will be ignored.

User action: None.

CBEDIT, to edit a CB use /FLAGS /CHANGE /DELETE /GTYPE or /ATTRIBUTE

Explanation: Line mode and prompt mode edit commands are not provided for CB entries. The command is ignored.

User action: Use one of the command mode edit commands specified.

CHKTERM, error checking to see if SYS\$OUTPUT is an ANSI terminal

Explanation: When the STATUS ON command is given, IPATCH checks to see if the terminal is an ANSI terminal. If it is not, then the status area will not work. If an error occurs establishing this fact, then various system error messages may result, followed by this message. The status area will not be established.

User action: Dependent upon the system errors. If in doubt, report this error to the system manager or to Laser-Scan, with the results of the DCL SHOW TERMINAL command.

CONVWHAT, missing arguments to TO - convert what to what?

Explanation: The TO command was not followed by one of DECIMAL, DEGREES, HEXADECIMAL or RADIANS.

User action: Repeat the command with the rest of the line correct.

CPBOTH, it is not possible to repeat the /BOTH operation

Explanation: The CP edit /REPEAT command cannot be used to repeat the /BOTH command, as this would involve changing the original side of the CP without any checks.

User action: If the original side of the CP must be changed, use the /BOTH command on each occasion.

CPEDIT, to edit a CP use /TARGET, /BOTH, /ROTATE or /REPEAT

Explanation: This message is given if line mode or prompt mode editing is attempted on a CP entry. These edit modes are not supported for the CP.

User action: Use one of the command mode edit commands, as suggested.

CPNONE, no previous CP edit to repeat

Explanation: The /REPEAT command has been used on a CP, but there has not been a previous CP edit.

User action: Use the appropriate CP edit command.

CRESTATUS, error creating status area - status is now OFF

Explanation: In order to maintain the status area on the screen, IPATCH uses binary writes to the relevant lines. This error is given if the program fails to open the terminal for binary writing. The status area will not be written.

User action: If in doubt, report this error to the system manager or to Laser-Scan, with the results of the DCL SHOW TERMINAL command.

DEORDER, current entry is before the marked entry - can't delete to the mark

Explanation: To perform a DELETE MARK, the marked entry must precede the current entry. The command is ignored.

User action: Ensure that the entry at the start of the range to be deleted is the one marked, and that the current entry when you give the command is the one AFTER the range to be deleted. Then try again.

DISABOVE, PREVIOUS is disabled above 'hex-address'

Explanation: This message should not occur in the current version of IPATCH. It is a historical message from earlier development versions.

User action: Report the message to Laser-Scan, with as much data as possible - preferably the IFF file and a log of the commands used to produce the message.

EDITCC, use the form 'letter' 'value' to change CC entries

Explanation: Line mode and prompt mode are not supported for editing CC entries. The command is ignored.

User action: Use the /UNIT command to set the CC to the unit transformation, or one of the commands /A to /T to edit individual values.

EDITCS, there are no /commands to edit a CS - try prompt mode

Explanation: Command mode is not supported for editing the CS entry, as it is considered obsolete. The command is ignored.

User action: Use line or prompt mode.

EDITSSL, there are no /commands to edit an SS or SL - try prompt mode

Explanation: Command mode is not supported for editing the SS or SL entries, as they are considered obsolete. The command is ignored.

User action: Use line or prompt mode.

ENDOFCOM, unexpected end of command file - now prompting

Explanation: If the command file specified with the /COMMANDS switch does not end with an EXIT command, then IPATCH will give this message and look for commands from SYS\$INPUT:.

User action: Either give the EXIT command to the first prompt, or proceed to use the program interactively.

ENDSCAN, error moving back to current entry after scanning file

Explanation: After scanning the IFF file to update the PREVIOUS list (for instance, after a NEXT command which moves off the end of the list), an error has occurred in going back to the entry that should be current. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

FSBITS, cannot find 'type' FS

Explanation: This message is given if one of the LINE, SYMBOL or TEXT commands has been used to search for an FS entry with the appropriate bits set in word 3, and IPATCH has not been able to find such an entry after the current entry. 'type' is one of "linear", "symbol" or "text". The program will position back to the current entry - the effect is as if the command

had never been given.

User action: None.

GOBACK, error going back to current entry, after PREVIOUS scan

Explanation: When moving backwards during the PREVIOUS command, IPATCH has to take various other entries into hand before making the destination entry current. This message indicates that some error occurred in that final step. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

GOTOEOF, error moving to end of file

Explanation: The EOF command has been given, but the program is unable to reach the place that it considers end of file. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

GOTOVO, error returning to just deleted entry (void)

Explanation: In order to delete an entry (using DELETE), IPATCH must effectively perform the sequence

```
MARK
NEXT
DELETE MARK
PREVIOUS
```

This message indicates that an error occurred in the final stage, equivalent to a PREVIOUS command. Other error messages should indicate more about the problem. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

HANDST, error taking ST/ZS/CB into hand again, after editing it

Explanation: Editing an ST, ZS, or CB entry can involve deleting it and then overwriting it with the new values (specifically, the /PEN and /DELETE commands must follow this sequence). This error indicates that after the new entry has been written out to the file, IPATCH tried to take it back into hand, but failed. Other error messages should indicate more about the problem. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

HEXRANGE, 'hex-number' is out of range - should be between 'hex-number' and 'hex-number'

Explanation: The previous command contained a hexadecimal value which was outside the range permitted for the particular application. The permitted range is indicated in the message. The command is ignored.

User action: Specify the command again with a corrected value.

IFFPKE, error in positioning to entry

Explanation: This error indicates that IPATCH tried to position to a given address, but failed to find an entry there. The unrecognised entry type, XX, will be indicated. The normal cause of this message is using FIND to move to a wrong address, or trying to read a corrupt area of an IFF file.

User action: If the error was caused by FIND, then check the argument - the address should be a hexadecimal number. If the error does not seem to be caused by the commands given or by the file itself, then report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

INTRANGE, 'integer' is out of range - should be between 'integer' and 'integer'

Explanation: The previous command contained a number which was outside the range permitted for the particular application. The permitted range is indicated in the message. The command is ignored.

User action: Specify the command again with a corrected value.

MARKDEL , no marked entry to delete from

Explanation: The DELETE MARK command has been given, but there is no marked entry. This means that either the MARK command has not been used to mark an entry, or that a previous command has removed the MARK. The commands DELETE and DELETE MARK both remove any existing marks.

User action: Mark the required entry and try again.

MARKERR, error in positioning back to MARKed entry

Explanation: This message follows an error in the RETURN command. It probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

MAXENTRIES, more than 200 entries from the current NF to the next

Explanation: The list of entries in the current 'feature' that is maintained to allow the PREVIOUS command can only contain up to 200 addresses. Thus if there are more than that number of entries between the two NFs under consideration, this message will appear. The MAXENTRIES message is followed by the PREVBETWEEN message (qv), which explains over which area of the file PREVIOUS will be disabled.

User action: If this is a problem, request Laser-Scan to rebuild IPATCH with a larger limit for the number of entries between NFs.

MAXPTS, 'entry' contains 'integer' points, but maximum storable is 200

Explanation: Conventionally, no ST, ZS, or CB entry may contain more than 200 points, and this is the maximum that IPATCH can read in. This message is given when an ST, ZS, or CB entry with more than that number of points is made current. If the IFF file is corrupt, then it is possible that this is not a true ST, ZS, or CB entry, but just a bit pattern that looks like such.

User action: If applicable, correct the program producing the ST, ZS, or CB entries with too many points.

MD2LEN, version 2 MD entry should be 'integer' words long

Explanation: IPATCH knows how many words there should be in a version 2 MD entry. If the definition of an MD entry has been changed to make the entry longer, or if a corrupt MD entry which appears longer is found, then this message will be given.

User action: Report the error to Laser-Scan, with as much detail as possible, including the date of release of the IPATCH image, the exact text of the message(s), and the IFF file.

MDVERSION, unknown map descriptor version 'integer'

Explanation: A map descriptor with an unrecognised version number has been found. Either the version of IPATCH has not been updated to recognise a new form of descriptor, or the MD entry is corrupt.

User action: Report the error to Laser-Scan, with as much detail as possible, including the date of release of the IPATCH image, the exact text of the message(s), and the IFF file.

MHEDIT, use the /CUSTOMER command to edit the MH customer number

Explanation: Line and prompt modes are not available for editing the MH entry, and the only command available is to edit the customer number. The edit command is ignored.

User action: If the customer number is to be changed, use the /CUSTOMER command, otherwise use the appropriate utility to edit the map header.

NEXTDEL , error moving to next entry, to delete this one

Explanation: The DELETE command must effectively perform the sequence

MARK
NEXT
DELETE MARK
PREVIOUS

This message indicates that the 'NEXT' stage has failed. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

NEXTEOF, you can't NEXT off the end of the file

Explanation: The current position is at the end of file, and a NEXT command has been given. It is ignored. Note that a NEXT 'count' command which tries to move over the end of the file will give this error and stop at the end of file.

User action: None.

NEXTERR, NEXT command abandoned after 'integer' entries

Explanation: This follows an error message from the IFF library, reflecting a problem in taking the next entry into hand. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

NEXTONE, NEXT command abandoned after 1 entry

Explanation: This follows an error message from the IFF library, reflecting a problem in taking the next entry into hand. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

NFGONE, no NF where expected, when attempting PREVIOUS scan

Explanation: When an entry is made current, and it is within the area of the file that has been scanned to allow PREVIOUS to operate there, then a routine is called to determine which entry within the current 'feature' this one is. If the initial NF entry of the 'feature' is not found where it is expected, then this error is given.

User action: Unless a sequence of actions which might have corrupted the NF entry has been taken (for instance using FIND to move near the NF and editing the 'apparent' entry there), this is probably an error in either IPATCH or the IFF library. In that case, report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

NOADDRESS, no (hexadecimal) address given after FIND

Explanation: The FIND command must be followed by the address of the entry it is to find. The command is ignored.

User action: Give the FIND command with the required address.

NOATT, fixed attribute 'name' ('code') not present in CB

Explanation: The attribute specified in a /ATTRIBUTE command is not present in the current CB entry, and can not therefore be edited.

User action: Specify an attribute which is present in the CB.

NOCORNER, no corner given - say which corner should be at the NW

Explanation: The CP edit command /ROTATE has been given without following it by the name of a corner which should be placed at the NW corner.

User action: Give the command again, with one of the corners NW,SW,SE,NE.

NOCURDEL, no current entry to delete

Explanation: The DELETE command has been given when there is no current entry. It is ignored.

User action: Make sure that the entry that is to be deleted is current before trying to delete it.

NOCURMARK, no current entry to MARK

Explanation: The MARK command has been given when there is no current entry. It is ignored.

User action: Make sure that the entry that is to be marked is current before trying to mark it.

NOEDIT, 'entry' entries cannot be edited

Explanation: An attempt has been made to edit an entry that cannot be edited - for instance an EF. The command is ignored.

User action: None.

NOENTRY, no entry at that position

Explanation: The FIND command has been used to move to an address that does not contain a recognisable entry. Other messages will help to clarify the problem.

User action: Check that the address given to FIND was correct, and that it was hexadecimal.

NOEOPTR, this NO entry does not contain an EO pointer

Explanation: An attempt has been made to edit the pointer field of an NO entry which does not include one. The command is ignored.

User action: If an EO pointer field must be added, then use a utility which can add new data to the IFF file - for instance LITES2.

NOMARK, no marked entry to RETURN to

Explanation: The RETURN command has been given, but there is no marked entry to move back to. Either the MARK command has not been used, or a DELETE or DELETE MARK command has removed the mark.

User action: Find the required entry by some other method.

NOSEARCH, error returning to old entry after search failed

Explanation: If a search for a particular entry reaches the end of file without finding the entry, then IPATCH makes the entry that was current before the search be current again. This message indicates that that operation failed. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

NOTANSI, SYS\$OUTPUT: is not an ANSI terminal

Explanation: The STATUS ON command outputs this message if the terminal will not support a status area. The status area is created and maintained using ANSI escape sequences, so the terminal must support them for a status area to be possible.

User action: None.

NOTBIT, 'integer' is out of range - should be 0 or 1 (bit unset or set)

Explanation: The various bit setting commands used to edit the second word of the FS entry take an argument which is either 0 or 1. This message is given if some other value is found.

User action: Try the command again with the correct value.

NOTENTRY, entry 'entry' at 'hex-address' is not simply reachable

Explanation: If the FIND command is used to position to an entry which is within the area of the file scanned for PREVIOUS, but could not be reached by using successive NEXT commands from the start of file, then this warning is given. It is probable that the apparent current entry is really just a bit pattern that looks like an entry to the IFF library.

User action: Check that the correct argument was given to FIND - it should be a hexadecimal address. Take great care in using or amending the entry.

NOTIFTEXT, cannot do that for text features (text bit is set)

Explanation: An attempt has been made to use the /PC edit command on an FS with the text bit set in word 3. Text features do not have a process code. The command is ignored.

User action: If the command was correct, use the /SYMBOL or /LINE command to change the FS and then try the /PC command again. Otherwise, perform an edit appropriate to a text.

NOTSTRING, LIST will only work for an ST, ZS, or CB

Explanation: The LIST command was used on a current entry that is not an ST or a ZS. The command is ignored.

User action: To show the contents of an entry that is not an ST or ZS, use one of the EXPLAIN or TYPE commands.

NOTWRITE, IFF file has not been opened for write

Explanation: An edit or delete command was given on a file which was not opened /WRITE. The command is ignored.

User action: If the file is to be edited, specify the /WRITE switch on the initial command line.

OPENCOM, error opening command file 'file-spec'

Explanation: The /COMMANDS switch was used to designate a file from which commands should be taken. However, the file could not be opened - other messages should explain why not. This message repeats the name of the file that IPATCH thought was being opened.

The program will continue with prompts to SYS\$INPUT:.

User action: Either EXIT and try with a correct file specification, or continue editing interactively.

OPENJOU, error opening journal file 'file-spec'

Explanation: The /JOURNAL switch was used to designate a file to which commands should be journalled. However, the file could not be opened - other messages should explain why not. This message repeats the name of the file that IPATCH thought was being opened.

The program will continue without journalling commands.

User action: Either EXIT and try with a correct file specification, or continue editing without journalling the commands.

ORDER, cannot work from 'integer' to 'integer' - try again in the other order

Explanation: A range was specified to a command such as LIST or /DELETE, with the first number being higher than the second. The range must be specified from low end to high end.

User action: Repeat the command with the numbers in the other order.

OUTRANGE, 'number' is out of range - should be between 'integer' and 'integer'

Explanation: The previous command contained a number which was outside the range permitted for the particular application. The permitted range is indicated in the message. The command is ignored.

User action: Specify the command again with a corrected value.

PREVBETWEEN, PREVIOUS will not work between 'hex-address' and 'hex-address'

Explanation: This message follows the MAXENTRIES message (qv). Since only the first 200 entries between the previous and next NF can be remembered, PREVIOUS will not work for the 201st entry through to the entry before the next NF. This message specifies which addresses the PREVIOUS command will thus not work between.

User action: If this is a problem, request Laser-Scan to rebuild IPATCH with a larger limit for the number of entries between NFs.

PREVBOF, you can't PREVIOUS off the start of the file

Explanation: The current position is at the start of file, and a PREVIOUS command has been given. It is ignored. Note that a PREVIOUS 'count' command which tries to move over the start of the file will give this error and stop at the start of file.

User action: None.

PREVERR, PREVIOUS command abandoned

Explanation: This follows an error message from the program or IFF library, reflecting a problem in taking the previous entry into hand. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

PREVNF, error looking for NF entry, during PREVIOUS scan

Explanation: If moving backwards beyond the start of the current 'feature', PREVIOUS makes previous NF entries temporarily current whilst it is moving back. This message indicates that an NF entry was not where one was expected, which indicates corruption of the list of NF entries.

User action: Unless the previous actions are likely to have confused IPATCH about the positions of NF entries (such as using FIND to move somewhere strange and editing over an actual NF), then this indicates a problem in either the IFF library or the program. Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

PREVXX, error looking for start-of-file entry, during PREVIOUS scan

Explanation: This error is similar to the PREVNF error, but occurs when IPATCH is trying to move back between the start of file and the first NF. It indicates that the first entry is now what was expected.

User action: As for the PREVNF error.

READCOM, error reading from command file 'file-spec' - abandoning file

Explanation: This message follows LSLLIB messages describing the problem with reading from the file specified by the /COMMANDS switch. IPATCH will start to prompt for input from SYS\$INPUT:.

User action: Dependent upon the LSLLIB errors.

REMARK, error restoring the MARK after replacing an ST/ZS/CB

Explanation: Some ST, ZS, and CB edit operations have to delete and then rewrite the entry. This has the side-effect of removing the current mark. However, they then attempt to reinstate the mark so that the operation is transparent to the user. This message indicates some problem in restoring the mark, and represents a bug.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

RESCAN, error moving to last known NF to continue PREVIOUS scan

Explanation: When the PREVIOUS command is given at a position above the area of the file that was already scanned, IPATCH must scan up to the current entry. This message follows IFF messages about problems repositioning to the last NF that was in the scanned list - ie the position at which IPATCH wishes to start its new scan.

User action: Unless the previous actions are likely to have confused IPATCH about the positions of NF entries (such as using FIND to move somewhere strange and editing over an actual NF), then this indicates a problem in either the IFF library or the program. Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

RETST, error returning to ST/ZS/CB entry after restoring the MARK

Explanation: Some ST, ZS, and CB edit operations have to delete and then rewrite the entry. This has the side-effect of removing the current mark. However, they then attempt to reinstate the mark so that the operation is transparent to the user. This message indicates some problem in returning to the ST/ZS/CB after restoring the mark, and represents a bug.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

RUBBISH, unexpected "'character'" - line ignored

Explanation: An unexpected character was found at the end of a command line which otherwise made sense. Lines must either end with a <CR> or be terminated by a comment character (!). The line is ignored.

User action: Repeat the line without the extraneous character, or prefix the character(s) by a comment symbol.

SCANERR, error scanning file for PREVIOUS

Explanation: This message is given when the IFF library has an error finding the next NF entry during a scan of the IFF file to allow the PREVIOUS command. It should follow IFF error messages explaining the problem.

User action: If the errors are not sensible from context, then report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

SEARCH, cannot find 'entry' 'details'

Explanation: This message is given if an entry search command (eg AC) has been given, and IPATCH has been unable to find such an entry after the current entry. 'entry' is the type of entry being searched for, and 'details' is the rest of the entry search specification (for instance, what AC type was requested). IPATCH repositions to the current entry - the effect is as if the search command had not been given.

User action: None.

SEARCHEOF, you can't search off the end of the file

Explanation: This message is given if you are positioned at the end of the file and attempt to search for an entry - that is, using the TEXT, LINE or SYMBOL commands, or one of the entry search commands. The command is ignored.

User action: If you want to search from the start of the file, then use the * command to move there. Otherwise, reconsider what effect is wanted.

STCHANGE, which points are to be changed?

Explanation: If the /CHANGE edit command is used on an ST, ZS, or CB without arguments, and the entry has more than one point, then it is not obvious which points are to be changed. The command is ignored.

Note that if there is only one point in the ST, ZS, or CB, then IPATCH deduces correctly which point to change.

User action: Repeat the command specifying which point(s) to change.

STDELETE, which points are to be deleted?

Explanation: If the /DELETE edit command is used on an ST, ZS, or CB without arguments, and the entry has more than one point, then it is not obvious which points are to be deleted. The command is ignored.

Note that if there is only one point in the ST, ZS, or CB, then IPATCH deduces correctly which point to delete.

User action: Repeat the command specifying which point(s) to delete.

STEDIT, to edit an ST or ZS use /PEN, /CHANGE or /DELETE

Explanation: Line mode and prompt mode edit commands are not provided for ST and ZS entries. The command is ignored.

User action: Use one of the command mode edit commands specified.

TEXTONLY, can only do that for text features (text bit is not set)

Explanation: One of /CATEGORY, /TYPESTYLE or /POSITION edit commands was used on an FS which has not got the text bit set. These commands are not applicable to lines or symbols.

User action: If the command was correct, then use the /TEXT edit command to change the FS to have the text bit set in word 3. Otherwise, use the /PC edit command to edit the process code.

TOOMANYNF, error in PREVIOUS scan - more than 65535 NF entries found in file

Explanation: The list of NF entries maintained by IPATCH to allow the PREVIOUS command can only hold up to 65535 NF entries. This message is given if an attempt is made to scan more than that. In fact, a correct IFF file cannot have more than 65535 NF entries in it, as the internal sequence number must be between 1 and 65535, and unique within the file.

User action: None.

TYPEERR, TYPE command abandoned after 'integer' entries

Explanation: This follows an error message from the IFF library, reflecting a problem in taking the next entry into hand. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

TYPEONE, TYPE command abandoned after 1 entry

Explanation: This follows an error message from the IFF library, reflecting a problem in taking the next entry into hand. This probably reflects a bug or problem in either IPATCH or the IFF library.

User action: Report the problem to Laser-Scan, with as much detail as possible - preferably the IFF file and a log of the commands which provoked the message.

UNEXPENT, unexpected entry 'entry' found

Explanation: This message is given when the program has moved to an address expecting to find one type of entry, and has instead found the type mentioned. It will be followed by other messages which detail the circumstances.

User action: Dependent upon the accompanying messages.

UNKENTRY, unknown entry type 'entry' at address 'hex-address'

Explanation: This message is output whenever the program attempts to print out an entry type that it does not recognise. This indicates that the program is not up to date with respect to the possible entries within an IFF file.

User action: Report the problem to Laser-Scan, with a note of the unrecognised entry type. Laser-Scan will either update IPATCH to recognise the entry, or request further details if the entry really does not exist.

UNKSEARCH, attempt to search for unknown entry type 'entry'

Explanation: This error should not occur, as it indicates that the searching section of the program has been asked to look for an entry that it does not know about.

User action: Report the problem to Laser-Scan, with a note of the unrecognised entry type. Laser-Scan will provide a corrected version of IPATCH.

UNMARK, the MARK is now removed

Explanation: This message follows the REMARK message, and acts as a reminder that the marked entry is no longer marked.

User action: As for the REMARK message.

WILDNEXT, DISPLAY is OFF - attempt to do "NEXT *" ignored

Explanation: This message is given if the NEXT * command has been used when the NEXT command will not display intermediate entries. Since the EOF command is provided to move immediately to the end of the IFF file, this is faulted.

User action: Either switch DISPLAY ON and try again, or use the EOF command.

WRITEJOU, error writing to journal file 'file-spec' - stopping journal

Explanation: This message follows LSLLIB messages describing the problem with writing to the file specified by the /JOURNAL switch. Journalling of commands will cease.

User action: Dependent upon the LSLLIB errors.

XXGONE, no entry where expected, when attempting scan for PREVIOUS

Explanation: The error is identical to the NFGONE error, except that the program was looking for the first entry in the file, rather than an NF entry.

User action: As for NFGONE.

MESSAGES (FATAL)

These messages reflect a problem with the program itself, and should be reported to Laser-Scan.

BADREPEAT, unknown operation 'integer' to REPEAT

Explanation: None.

User action: Report problem to Laser-Scan, together with a note of which command was just given. A new version of IPATCH will be issued.

NOSUCHCMD, command number 'integer' not recognised

Explanation: None.

User action: Report problem to Laser-Scan, together with a note of which command was just given. A new version of IPATCH will be issued.

NOSUCHCMD2, secondary command number 'integer' not recognised

Explanation: None.

User action: Report problem to Laser-Scan, together with a note of which command was just given. A new version of IPATCH will be issued.

MESSAGES (OTHER)

In addition to those which are generated by IPATCH itself, messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries.

In particular, messages may be generated by the IFF library and by the Laser-Scan I/O library, LSLLIB.

IFF library messages are introduced by '%IFF' and are documented in the IFF library user's guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user, and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

LSLLIB messages are introduced by '%LSLLIB' and are generally self-explanatory. They are used to explain the details of program generated errors.

CHAPTER 12

MODULE IRECODE

MODULE **IRECODE**

REPLACES Nothing, IRECODE is a new IMP utility.

FUNCTION

IRECODE enables change of selected layer, feature code, text component code or AC numbers in an existing IFF file to specified new values. By default, IRECODE creates an output IFF file to contain the altered data. Alternatively, changes to the input file may be made in-situ.

FORMAT

\$ IRECODE input-file-spec [output-file-spec]

Command qualifiers

Defaults

/[NO]ALPHABETIC=filename	/NOALPHABETIC
/AC[=(integer>integer,...)]	None.
/FC[=(integer>integer,...)]	IRECODE prompts for feature codes.
/TCC[=(integer>integer,...)]	None.
/IN_SITU	Create a new file.
/LAYER[=(integer>integer,...)]	None.
/[NO]LOG	/NOLOG

NOTE

IRECODE (like ILAYFC) is primarily intended as an interactive program. By necessity its command line syntax is complex. Great care should be taken if using it non-interactively as it will **prompt** for input if the command line contains errors.

If, for example, the user issues this IRECODE command line within a command file and supplies the arguments for /LAYER as if they were to be prompted for:

```
$ IRECODE/AC=(1>g)/LAYER<CR>
2>3<CR>
4>6<CR>
5>8<CR>
<CR>
$
```

IRECODE will detect the erroneous "g" in the argument list for /AC and enter prompt mode. What had been intended as arguments for the /LAYER qualifier will now be read as arguments for /AC. Fortunately, in this example, IRECODE will detect end-of-input-file (the dollar of the next DCL command line) when trying

to get arguments for /LAYER. IRECODE will then abort execution. If, however, the user had mistakenly specified 2 blank lines in the input file before the dollar, /LAYER argument input would be considered complete and entirely erroneous recoding would have occurred. \$STATUS would be set to SS\$_NORMAL and the errors would continue through the command file.

PROMPTS

_Input-File: input-file-spec

If /IN_SITU is not specified, then:

_Output-File: output-file-spec

PARAMETERS

input-file-spec

- specifies the IFF file which is to be recoded. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF;0'.

output-file-spec

- specifies the IFF file which is to be created. By default, IRECODE creates an output IFF file to contain the altered data. Alternatively, changes to the input file may be made in-situ by specifying the /IN_SITU qualifier. No output file specification should be supplied on an IRECODE command line containing the /IN_SITU qualifier.

Any part of the output file specification which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

/ALPHABETIC=filename

This qualifier allows the optional text field contained in the Laser Scan default AC entry of type 1 (Secondary_FC), to be used to change the feature codes of features taken from the input IFF file.

The file which /ALPHABETIC takes as its argument contains a list of entries which specify the text (to be compared with that from the AC entry) and their corresponding feature codes. These entries must be of the form:

Definition_string Output_feature_code.

NOTE

Entries which have an exclamation mark '!' as their first non-space character will be assumed to be comments and will be ignored.

A Definition_string which consists of more than one word must be enclosed in quotes.

The Definition_string and Output_feature_code may be separated by an optional comma (',') or a greater-than sign ('>').

IRECODE will read the entries from the file and compare the Definition_strings with the optional text contained in the AC type 1 entries. If the strings match, the current feature code in the output file will be modified using the Output_feature_code associated with the string.

/AC[=(integer>integer[,...])]

- where the first integer of each pair of integers is the old AC type number, the second the new AC type number which is to replace the old one.

If a syntax error is made in the command string, or /AC is specified with no arguments then IRECODE will prompt for input:

Old AC type number, New AC type number ?

.
.
.

This prompt will be repeated for additional arguments until IRECODE detects a blank line.

W A R N I N G

-----+-----
| Beware that if the /AC qualifier is used on an AC type having
| a floating point value in the AC longword field then the longword
| will be rounded to the nearest integer (via FORTRAN NINT). If
| used to change an integer longword AC to an AC type having a
| floating point longword field the integer will be floated.
|-----+-----

| All AC types have an integer value stored in their longword field
| except type 3 and type 80 through 99 ACs) which have a floating
| point number.
|-----+-----

/FC[=(integer>integer[,...])]

- where the first integer of each pair of integers is the old feature code, the second the new feature code which is to replace the old code. This is the default mode of IRECODE. Failure to specify any qualifiers will result in IRECODE prompting for input. Similarly, if a syntax error is made in the /FC argument list, or /FC is specified with no arguments, then IRECODE will enter prompt mode:

Old Feature Code, New Feature Code ?

.
.
.

This prompt will be repeated for additional arguments until IRECODE detects a blank line.

/TCC[=(integer>integer[,...])]

- where the first integer of each pair of integers is the old TCC (Text Component Code), the second the new TCC which is to replace it. If a syntax error is made in the /TCC argument list, or /TCC is specified with no arguments, then IRECODE will enter prompt mode:

Old Text Component Code, New Text Component Code ?

.
.
.

This prompt will be repeated for additional arguments until IRECODE detects a blank line.

/IN_SITU
/NOIN_SITU

- the default mode of IRECODE operation is to create a new IFF file to receive the modified data. The /IN_SITU qualifier forces modification to occur in-situ, thus destroying the original data. /IN_SITU is not available if an output file is specified. Qualifiers on the command line preceding the /IN_SITU qualifier are ignored by the Command Line Interpreter (CLI) and a warning message is given.

/LAYER[=(integer>integer[,...])]

- where the first integer of each pair of integers is the old layer number, the second the new layer number which is to replace the old one. If a syntax error is made in the /LAYER argument string, or /LAYER is specified with no arguments, then IRECODE will prompt for input:

Old Layer Number, New Layer Number ?

.
.
.

This prompt will be repeated for additional arguments until IRECODE detects a blank line.

/LOG
/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened, and a reassuring message output each time a new IFF layer is encountered in the IFF file.

DESCRIPTION

Used on an existing IFF file, IRECODE enables the global change of the following IFF entry values within the file:

- o feature codes, (the first word of an FS (Feature Status) entry), and/or,
- o text component codes, (the first word of a TS (Text Status) entry), and/or,
- o layer numbers, and/or,
- o AC type numbers.

The IFF entry type which is to be changed is specified with the qualifier of the corresponding name. The 'old' and 'replacement' values for the specified IFF entry may be specified as arguments to the IFF entry qualifier. An argument pair, 'old' and 'replacement', is separated by a closing angle bracket. Multiple argument pairs are separated by commas and the whole list of argument pairs is enclosed in parentheses. In the event of a command decoding error or in the event of no arguments being supplied to an IFF entry qualifier IRECODE will enter prompt mode and explicitly request argument pairs from the user. Exit from prompt mode is achieved by specifying a blank line.

An option to change the input file in-situ is provided with the /IN_SITU qualifier. Only one file-spec parameter should be specified on the command line if the in-situ mode of operation is selected. It is very dangerous to work on an IFF file in situ as some changes are irrevocable. Such changes fall into two classes:

1. changes of AC types involving ACs with floating point numbers in the longword field. If truncated to an integer during an AC type number change there is no mechanism for the restoration of the floating point precision.
2. Changes which result in the concatenation of diverse feature types. Features may be differentiated by being assigned different feature codes or IFF layer numbers. Thus if multiple layers are assigned the same layer number, or diverse feature codes are made the same, feature segregation is lost. There is no mechanism available for re-segregating the combined data other than manually

reassigning values with reference to the digitising source using LITES2.

Implications of changing IFF entry values

Changing the value of IFF entries with IRECODE can have implications beyond the immediate value change. These implications are outlined below for each of the IRECODE IFF entry code qualifiers.

/AC

A number of AC types are reserved by Laser-Scan for special applications. IFF applications programs (such as those of IMP!) are written with awareness of the specialist nature of some of these AC types. AC entries are used to hold miscellaneous information. They are composed of a word (integer*2) type, a longword (integer*4 or real) code and an optional text.

The type of information held in an AC is determined by the AC type. Current type values are defined in the IFF User Guide.

AC types may take values in the range 0-32767. Negative types are not allowed.

The range of the AC code is implicitly specified by the use of the AC. Note that AC type 3 and types 80 to 99 (inclusive) are currently the only ACs to have a real value in the AC longword field.

If an AC type number is changed from one associated with a real (floating point number) longword field to an AC type number associated with an integer value in the longword field the floating point number will be rounded.

The length of the text field is limited to 255 characters.

/FC

The Feature Code of a feature is carried in the first word of the FS entry of each feature. IRECODE only changes the first (feature code) word of the FS entry. As the feature code is stored as a word (16 bit integer) its value must lie in the range -32768 to 32767. In practice negative feature codes are not allowed. The other 3 words of the IFF FS entry remain unaltered. Be very careful to ensure that by changing the feature code, conflicts with Words 2 to 4 of an IFF FS entry are defined in the IFF User Guide.

/LAYER

IFF files are generally divided up into multiple layers, where data of a common sort or source is grouped in the same layer. Note that a layer may be split into several parts, identified by all having the same layer number.

Layer 0 is conventionally reserved for 'non-essential' data - for instance a grid, symbols at control points, or MCE DFAD accuracy polygons. The assumption is that the IFF file would not be significantly degraded in terms of information content by throwing layer 0 away. If layer 0 is present, some programs expect it to be the first layer in the file.

Historically, layers 11 and 32 were also used for the same sort of purpose as layer 0. If working with an IFF file belonging to an historical database this should be taken into consideration.

/TCC

Text features may contain one text string, with associated location and descriptive data, or they may be **composite** - that is composed of several sub-texts or text components, which may be manipulated independently or as a single entity.

Each text component starts with a TS entry, and ends with the next TS entry, or the final EF of the feature. The first TS entry occurs immediately after the FS entry and any AC entries. Text components may not include FS or AC entries, but may contain any other entries that are legal within a normal text feature.

Word 1 of the TS entry is the text component code (TCC), which is the primary descriptive code for a text component - it is effectively the feature code for this component of the composite text, and is used in the same manner. IRECODE enables the user to alter only the first word of a TS entry, i.e. the TCC. Alteration of the TCC should, however, take account of the third word of the TS entry.

Word 3 of the TS entry is the text word, and is identical in form to word 3 of a text's FS entry - it holds a description of the type of text component.

The second and fourth words are reserved for later definition. For a detailed description of the IFF TS entry see the IFF User Guide.

EXAMPLES

```
$ IRECODE<CR>
Input-file: S320480SW.IFF/LOG<CR>
Output-File: TEST.IFF<CR>
Old Feature Code, New Feature Code ? 123,4<CR>
Old Feature Code, New Feature Code ? <CR>
%LSLLIB-I-IFFOPENED, DUA0:[LSL.ACCEPTANCE]S320480SW.IFF;4 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[LSL.ACCEPTANCE]TEST.IFF;1 opened for write
%IRECODE-I-LAYER, layer 1 found
%IRECODE-I-LAYER, layer 80 found
ELAPSED: 00:00:27.67 CPU: 0:00:05.50 BUFIO: 9 DIRIO: 155 FAULTS: 247
$
```

This example shows the default mode of IRECODE operation. No parameters are specified and so IRECODE prompts for input and output files. None of the /FC, /AC, /TCC or /LAYER qualifiers are specified so IRECODE prompts for feature codes. In this example all input file features with a feature code of 123 will be given a feature code of 4 in the output file LSL\$IF:TEST.IFF;1. The /LOG qualifier provides messages indicating the full file specifications of the files opened and each layer found in the input IFF file.

```
$ IRECODE S320480SW.IFF/AC=(3>J8)/LOG<CR>
Output File: TEST.IFF<CR>
%IRECODE-W-BADARGFMT, bad format for arguments
Old AC type number, New AC type number ? 5, D<CR>
%IRECODE-W-BADARGFMT, bad format for arguments
Old AC type number, New AC type number ? 3<CR>
%IRECODE-W-TWOARGS, two arguments required
Old AC type number, New AC type number ? 3,5<CR>
Old AC type number, New AC type number ? <CR>
%LSLLIB-I-IFFOPENED, DUA0:[LSL.ACCEPTANCE]S320480SW.IFF;4 opened for read
%LSLLIB-I-IFFOPENED, DUA0:[LSL.ACCEPTANCE]TEST.IFF;2 opened for write
%IRECODE-I-LAYER, layer 0 found
%IRECODE-I-LAYER, layer 1 found
%IRECODE-I-LAYER, layer 4 found
%IRECODE-I-LAYER, layer 5 found
%IRECODE-I-LAYER, layer 6 found
ELAPSED: 00:00:53.51 CPU: 0:00:03.68 BUFIO: 9 DIRIO: 112 FAULTS: 250
$
```

This example illustrates IRECODE behaviour when an error is detected while decoding qualifier code change pair arguments. A 'J' has been included in the /AC argument string. IRECODE abandons decoding of the argument list for the /AC qualifier and prompts for the correct input. The user should enter each code change pair on a new line. In the example the user make two further errors while specifying the code change pairs in response to the prompt for input. In each case IRECODE issues a warning message and re-prompts. When the whole list of code change pairs for the /AC qualifier is re-entered the user should enter a blank line. Processing will then begin. In the

example IRECODE detects no processing errors and the run completes normally. \$STATUS is set to SS\$_NORMAL.

```
$ IRECODE S320480SWF.IFF/AC=(3>5)/LAYER=(1>77,88>2)/IN_SITU TEST.IFF<CR>
%CLI-I-IGNQUAL, qualifiers appearing before this item were ignored
\IN_SITU\
%CLI-W-MAXPARAM, too many parameters - reenter command with fewer parameters
\TEST\
ELAPSED: 00:00:00.74 CPU: 0:00:00.13 BUFIO: 2 DIRIO: 0 FAULTS: 42
$
$ IRECODE S320480SWF.IFF/AC=(3>5)/LAYER=(1>77,88>2)/IN_SITU<CR>
%CLI-I-IGNQUAL, qualifiers appearing before this item were ignored
\IN_SITU\
%LSLLIB-I-IFFOPENED, DUA0:[LSL.ACCEPTANCE]S320480SWF.IFF;4 opened for
modification
Old Feature Code, New Feature Code ? <CTRL/Z>
ELAPSED: 00:00:04.86 CPU: 0:00:00.54 BUFIO: 2 DIRIO: 14 FAULTS: 87
$
```

The first example here shows an attempt to run IRECODE specifying both the /IN_SITU qualifier and an output file parameter. The Command Line Interpreter (CLI) issues error messages and IRECODE aborts execution. The command line is respecified without the output parameter. Unfortunately the user has specified qualifiers on the command line which occur **before** the /IN_SITU qualifier. The CLI ignores these. This results in the CLI interpreting the command line as if no code change qualifiers were specified. IRECODE therefore prompts for Feature Code change pairs by default. The user, realising the error, responds with <CTRL/Z> and IRECODE aborts.

```
$ IRECODE S320480SW.IFF/TCC/FC/LAYER<CR>
Output File: TEST.IFF<CR>
Old Feature Code, New Feature Code ? 3,5<CR>
Old Feature Code, New Feature Code ? 4,6<CR>
Old Feature Code, New Feature Code ? 3,6<CR>
%IRECODE-W-REPCHANGE, repeated change specified - latest will be applied
\3>5 and 3>6\
Old Feature Code, New Feature Code ? 7,9<CR>
Old Feature Code, New Feature Code ? 8,10<CR>
Old Feature Code, New Feature Code ? 12,11<CR>
Old Feature Code, New Feature Code ? <CR>
Old Text Component Code, New Text Component Code ? 2 4<CR>
Old Text Component Code, New Text Component Code ? 3 4<CR>
Old Text Component Code, New Text Component Code ? <CR>
Old Layer Number, New Layer Number ? 1,8<CR>
Old Layer Number, New Layer Number ? <CR>
ELAPSED: 00:00:35.64 CPU: 0:00:04.43 BUFIO: 9 DIRIO: 138 FAULTS: 169
$
```

This example shows IRECODE prompting for code change pairs for the /FC, /TCC and /LAYER qualifiers. The user specified 2 valid Feature Code change pairs. A third feature Code change pair was then specified. IRECODE detected that a change to feature code 3 had already been specified (change feature code 3 to 5 on prompt line 1).

A warning message is issued and the most recent feature code change for feature code 3 is substituted. Feature code 3 will now be changed to 4 not 5. A further three feature code changes were then specified followed by a blank line. IRECODE then prompted for TCC (Text Component Code) change pairs. The user specified TCC change pairs and then specified a blank line. Finally, IRECODE prompted for Layer change pairs. The user specified one Layer change pair and then specified a blank line. IRECODE then successfully processed the file, writing the modified feature codes and layer numbers to LSL\$IF:TEST.IFF;0.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). Most will only appear if /LOG has been specified on the command line, unless they are used to provide supplementary information after an error has occurred.

ARRANGE, IRECODE qualifier arguments must lie in the range 0 - 32767

Explanation: The specified argument value (or value typed in response to a prompt) is outside the specified range.

User action: Specify values that lie within the specified range.

LAYER, Layer 'integer' found

Explanation: The specified layer has been encountered in the IFF file.

User action: None.

LINE SKIP, skipping over line due to warning

Explanation: A warning condition occurred while reading a line from the file given with the /ALPHABETIC qualifier

User action: The warning message which preceded this message identifies what the problem was. Check the file given as the argument for the /ALPHABETIC qualifier.

CANALPHA, cancelling /ALPHABETIC qualifier

Explanation: An error occurred while reading from entries in the file specified with the /ALPHABETIC qualifier.

User action: This error suggests that the entries in the file were corrupt. Check that the file contained text only and that it was not corrupt.

NOMATCH, text from ACD1 "string" unmatched

Explanation: The "string" read from the optional text field an AC entry of type 1 (ACD1), was not matched with text entries read from the file specified with the /ALPHABETIC qualifier. As a result the feature code (FC) of the feature containing that AC entry will not be modified.

User action: If the user wished to modify the FC of the feature which had an ACD1 with the specified text entry, then an appropriate entry should be made in the file specified with the /ALPHABETIC qualifier.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

ARGSREQ, at least one argument pair required.

Explanation: The user has failed to provide any 'OLD', 'REPLACEMENT' code pairs and has pressed carriage return in response to one of the prompts:

Old Feature Code, New Feature Code?
or:
Old Text Component Code, New Text Component Code?
or:
Old AC type number, New AC type number?
or:
Old Layer Number, New Layer Number ?

User action: IRECODE expects at least one AC, Layer, Feature or TCC code pair to work with and will persist in reprompting until one is provided. Use <CTRL/Z> to abort the IRECODE run if you are unsure of the correct response.

BADARGFMT, bad format for arguments

Explanation: The parameters have been entered in an incorrect format (see command qualifiers section), the user must enter them all again.

User action: IRECODE cannot read the supplied arguments. Instead it will prompt for input.

The user should supply the arguments in the form requested. Further arguments can be supplied, if desired, and the argument list terminated by specifying a blank line.

BADARGRA, qualifier argument out of range

Explanation: The specified argument value (or value typed in response to a prompt) is outside the specified range.

User action: Specify values that lie within the specified range.

BADCHR, bad character in string

Explanation: A bad character is present in the argument list, IRECODE will prompt for input which should be supplied using the format:

'integer', 'integer'<CR>

User action: Supply the arguments in the form requested. Further arguments can be supplied, if desired, and the argument list terminated by specifying a blank line.

TOOMNYPAR, too many argument pairs - maximum 'integer'

Explanation: Too many code change argument pairs were supplied in prompt mode.

User action: The maximum number of code change argument pairs allowed is large. Ensure that you are quite certain that you really want to change so many code values. If you do, you will have to run IRECODE twice. The first time you should specify an output file and change up to the maximum number of codes only. The second time you may wish to modify the output file of the first IRECODE run using IRECODE in /IN_SITU mode. You can then change all remaining codes to specified new values. Should you wish to change such large numbers of IFF entry code values on a regular production basis please contact Laser-Scan.

TWOARGS, two integer arguments required.

Explanation: Only one integer argument was specified either as a qualifier argument or in response to a prompt for input.

User action: Supply the arguments in the form requested. Further arguments can be supplied, if desired, and the argument list terminated by specifying a blank line.

The following messages are self-explanatory and relate to entries being read from the file specified with the /ALPHABETIC qualifier:

- o UNEXPEOL, unexpected end of line while reading from file.
- o NOALPSUCC, no entries successfully read from file.
- o NOINTPOSS, error in reading string, no integer possible.
- o NOINT, error while attempting to read integer.
- o INTARGRNG, integer in string is out of range.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused IRECODE to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ERRSSNVIO, successive syntax violations - please read manual

Explanation: The user has failed to enter the argument pair in the correct format, in four successive attempts. IRECODE exits.

User action: Read the documentation before trying again.

UNEXPEOF, unexpected end of IFF file.

Explanation: The specified input IFF file ended before an EJ entry was encountered. The valid part of the input IFF file was processed, but upon detection of this error condition IRECODE deletes the output file and aborts execution.

User action: Use IMEND to correctly terminate the input file before re-specifying the IRECODE command line.

UNEXPERR, unexpected error while reading from file.

FILERR, unexpected file error.

Explanation: These errors relate to the file specified with the /ALPHABETIC qualifier.

User action: Check that the file contains only text entries and that it is not damaged.

MESSAGES (OTHER)

In addition to the above messages which are generated by IRECODE itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 13

MODULE IRENUMBER

MODULE **IRENUMBER**

REPLACES **DAMP module IFSN**

FUNCTION

IRENUMBER reallocates Feature Serial Numbers (FSNs) in monotonically increasing order. IRENUMBER does **not** reallocate Internal Sequence Numbers (ISNs). By default IRENUMBER creates a new output file but the /IN_SITU qualifier is available to modify an IFF file in-situ. IRENUMBER is intended mainly for recovery from error situations which have resulted in multiply allocated FSNs within an IFF file.

FORMAT

\$ IRENUMBER input-file-spec [output-file-spec]

Command qualifiers

/BASE=integer

/INCREMENT=integer

/IN_SITU

/[NO]LOG

/LOWER_LIMIT=integer

/UPPER_LIMIT=integer

Defaults

/BASE=1

/INCREMENT=1

Create a new output file.

/NOLOG

/LOWER_LIMIT=1

/UPPER_LIMIT=65535

PROMPTS

_Input-File: input-file-spec

 If /IN_SITU is not specified, then
_Output-File: output-file-spec

PARAMETERS

input-file-spec

- specifies the IFF file which is to be recoded. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

[output-file-spec]

- specifies the IFF file which is to be created. By default, IRENUMBER creates an output IFF file to contain the altered data. Alternatively, changes to the input file may be made in-situ by specifying the /IN_SITU qualifier. No output file specification should be supplied on an IRENUMBER command line containing the /IN_SITU qualifier.

Any part of the output file specification which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

/BASE=integer

- sets the base for the new number allocation. The integer argument to /BASE must lie in the range 1 to 65535. The default base is 1.

/INCREMENT=integer

- sets the increment for the new number allocation. The increment must be a positive, non-zero integer. The default increment is 1.

/IN_SITU

- the default mode of IRENUMBER operation is to create a new IFF file to receive the modified data. The /IN_SITU qualifier forces modification to occur in-situ, thus destroying the original FSNs. Qualifiers on the command line preceding /IN_SITU are ignored by the command line interpreter (CLI) and a warning message given.

/LOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened, and a reassuring message output each time a new IFF layer is encountered in the IFF file. The new and old FSNs are listed as they are changed. On large files the output produced can be voluminous.

/LOWER_LIMIT=integer

- sets the lower limit below which feature numbers in the input file will not be modified. The integer argument to /LOWER_LIMIT must be in the range 1 to 65535. The default lower limit is 1.

/UPPER_LIMIT=integer

- sets the upper limit above which feature numbers in the output file will not be modified. The integer argument to /UPPER_LIMIT must be in the range 1 to 65535. The default upper limit is 65535.

DESCRIPTION

IRENUMBER reallocates IFF Feature Serial Numbers (FSNs) in monotonically increasing order. IRENUMBER is intended mainly for recovery from error situations which have resulted in multiply allocated FSNs within an IFF file. IRENUMBER does not change the Internal Sequence Number (ISN) of the feature.

By default IRENUMBER will produce a new IFF file as output. An option to work on the input file in-situ is provided with the /IN_SITU qualifier. If /IN_SITU is specified and IRENUMBER is used incorrectly there is no route for the recovery of the original form of the data. Use /IN_SITU with great care.

In the absence of any command qualifiers IRENUMBER will reallocate all FSNs found in the IFF file using a replacement base value of 1. The default increment is 1.

Features having an FSN of 0 will not be renumbered. Such features historically have a special context in IFF files which have TC (Transmitted Comment) entries which lie **outside** of IFF features.

IRENUMBER may be used to reallocate a subset of the FSNs within the IFF file. Lower and upper FSN values may be specified with the /LOWER_LIMIT and /UPPER_LIMIT qualifiers respectively. If these qualifiers are specified IRENUMBER takes the following action:

- o /LOWER_LIMIT only - IRENUMBER renumbers all FSNs whose current values lie between the specified lower limit and 65535, (inclusive).
- o /UPPER_LIMIT only - IRENUMBER renumbers all FSNs whose current values lie between 1 and the specified upper limit, (inclusive).
- o /LOWER_LIMIT and /UPPER_LIMIT - IRENUMBER renumbers all FSNs whose current values lie between the specified lower limit and the specified upper limit, (inclusive).

The value for /BASE should be chosen with care. If the result of the formula:

$$\text{BASE} + (\text{'number of features'} * \text{INCREMENT})$$

is greater than 65536 IRENUMBER issues a warning message and ignores subsequent features. The output IFF file will be only partly renumbered.

EXAMPLES

```
$ IRENUMBER/IN_SITU/LOG/BASE=10 W153.IFF<CR>
```

```
%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.TEST]W153.IFF;4 opened for update
```

```
%IREN-I-FSNINCBAS, renumbering FSNs in increments of 1 from base 10
```

```
between LOWER_LIMIT 1 and UPPER_LIMIT 65535
```

```
%IREN-I-LAYER, layer 0 found
```

```
%IREN-I-LAYER, layer 10 found
```

```
%IREN-I-LAYER, layer 12 found
```

```
%IREN-I-RENFSN, renumbered FSN 1 to 10
```

```
%IREN-I-RENFSN, renumbered FSN 1 to 11
```

```
%IREN-I-RENFSN, renumbered FSN 1 to 12
```

```
%IREN-I-RENFSN, renumbered FSN 1 to 13
```

```
%IREN-I-LAYER, layer 101 found
```

```
%IREN-I-RENFSN, renumbered FSN 1 to 14
```

```
%IREN-I-RENFSN, renumbered FSN 2 to 15
```

```
%IREN-I-RENFSN, renumbered FSN 3 to 16
```

```
%IREN-I-RENFSN, renumbered FSN 4 to 17
```

```
%IREN-I-RENFSN, renumbered FSN 5 to 18
```

```
%IREN-I-RENFSN, renumbered FSN 6 to 19
```

```
%IREN-I-RENFSN, renumbered FSN 7 to 20
```

```
%IREN-I-RENFSN, renumbered FSN 8 to 21
```

```
%IREN-I-RENFSN, renumbered FSN 9 to 22
```

```
%IREN-I-RENFSN, renumbered FSN 10 to 23
```

```
%IREN-I-RENFSN, renumbered FSN 11 to 24
```

```
%IREN-I-RENFSN, renumbered FSN 12 to 25
```

```
%IREN-I-RENFSN, renumbered FSN 13 to 26
```

```
%IREN-I-RENFSN, renumbered FSN 14 to 27
```

```
%IREN-I-RENFSN, renumbered FSN 15 to 28
```

```
%IREN-I-RENFSN, renumbered FSN 16 to 29
```

```
%IREN-I-RENFSN, renumbered FSN 17 to 30
```

```
%IREN-I-RENFSN, renumbered FSN 18 to 31
```

```
%IREN-I-RENFSN, renumbered FSN 19 to 32
```

```
%IREN-I-RENFSN, renumbered FSN 20 to 33
```

```
%IREN-I-RENFSN, renumbered FSN 21 to 34
```

```
%IREN-I-RENFSN, renumbered FSN 22 to 35
```

```
%IREN-I-RENFSN, renumbered FSN 23 to 36
```

```
%IREN-I-RENFSN, renumbered FSN 24 to 37
```

```
%IREN-I-RENFSN, renumbered FSN 25 to 38
```

```
%IREN-I-RENFSN, renumbered FSN 26 to 39
```

```
%IREN-I-RENFSN, renumbered FSN 27 to 40
```

```
%IREN-I-RENFSN, renumbered FSN 28 to 41
```

```
%IREN-I-RENFSN, renumbered FSN 29 to 42
```

```
%IREN-I-RENFSN, renumbered FSN 30 to 43
```

```
%IREN-I-RENFSN, renumbered FSN 31 to 44
```

```
%IREN-I-RENFSN, renumbered FSN 32 to 45
```

```
%IREN-I-RENFSN, renumbered FSN 33 to 46
```

```
%IREN-I-RENFSN, renumbered FSN 34 to 47
```

```
%IREN-I-RENFSN, renumbered FSN 35 to 48
```

```
%IREN-I-LAYER, layer 111 found
```

```
%IREN-I-RENFSN, renumbered FSN 36 to 49
```

```
%IREN-I-RENFSN, renumbered FSN 37 to 50
```

```
%IREN-I-NUMFTREN, 41 features renumbered
```

```
%IREN-I-NUMFTIGN, 0 features ignored
```


ELAPSED: 00:00:06.26 CPU: 0:00:02.67 BUFIO: 2 DIRIO: 57 FAULTS: 79

\$

This example demonstrates the /LOG, /IN_SITU, and /BASE qualifiers. The /IN_SITU qualifier means that the file W153.IFF is modified in-situ rather than a new IFF file being created. As /LOG has been specified after /IN_SITU, information is output about the input file being opened, the parameters that the program is using, each layer as it is encountered, and each FSN as it is renumbered. Finally the total number of features renumbered and the total number ignored are output. The /BASE qualifier has the effect of renumbering the FSN's from 10. As no other qualifiers were given the default INCREMENT, LOWER_LIMIT and UPPER_LIMIT were used by the program.

\$ IRENUMBER/LOG NEBRASKA<CR>

Output file: TEST.IFF<CR>

%LSLLIB-E-IFFOPEN, IFF error opening file "LSL\$IF:NEBRASKA.IFF;" for read

%IFF-E-OPEN, error opening IFF file

-RMS-E-FNF, file not found

ELAPSED: 00:00:03.52 CPU: 0:00:00.15 BUFIO: 1 DIRIO: 0 FAULTS: 63

\$

In this example the user specified LSL\$IF:NEBRASKA.IFF as the input file-spec and the /LOG qualifier. As no output file was given on the command line, and the /IN_SITU qualifier was not given IRENUMBER prompts for an output filename. The user then specified LSL\$IF:TEST.IFF. However, the input file NEBRASKA.IFF could not be found in LSL\$IF: so program execution ends with suitable error messages. Either the filename has been mis-typed or the logical name LSL\$IF set to an inappropriate device and directory.

Had the input file been found the program would have created LSL\$IF:TEST.IFF as ouput.

\$ IRENUMBER TEST1/LOWER_LIMIT=10/INCREMENT=2<CR>

Output file: TEST1.REN<CR>

ELAPSED: 00:01:56.05 CPU: 0:00:50.85 BUFIO: 8 DIRIO: 436 FAULTS: 141

\$

In this example the user specified LSL\$IF:TEST1.IFF as input with a LOWER_LIMIT of 10 and INCREMENT of 2. As no output file was given on the command line, and the /IN_SITU qualifier was not given the program prompts for an ouput filename. The user typed TEST1.REN to specify the output file In the output file features with feature numbers less than 10 would not be modified. Features with feature numbers greater than 10 would have been renumbered in increments of 2 from the default BASE of 1 to an UPPER_LIMIT of 65535.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user. They are often used to supply explanatory information in support of a warning or error message.

FSNINCBAS, renumbering FSNs in increments of 'integer' from base 'integer' between LOWER_LIMIT 'integer' and UPPER_LIMIT 'integer'

Explanation: The specified increment, base and upper and lower limits are being used.

User action: None.

RENFSN, renumbered FSN 'integer' to 'integer'.

Explanation: The specified FSN 'integer' has been renumbered to the indicated value.

User action: None.

IGNFSN, ignoring FSN 'integer'.

Explanation: FSN 'integer' has been ignored.

User action: None.

LAYER, layer 'integer' found

Explanation: Layer 'integer' has been found in the input file.

User action: None.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user or that continued processing will result in a valid IFF output file.

RCHMAXFSN, reached maximum valid FSN number 'integer'

Explanation: The renumbering of features has reached the maximum valid FSN number 65535. No further renumbering will take place.

User action: If a large increment is the cause of this message and unique FSN numbers are required then a smaller increment should be specified. If the default increment has been used then the file is likely to be too large for most IMP utilities and other LSL software.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ERRCLOINP, error closing input file 'file-spec'.

ERRCLOUT, error closing output file 'file-spec'.

Explanation: An error has occurred in closing the specified IFF file.

User action: The accompanying messages should help you to interpret the cause of this error. Check that there is sufficient disk space for the file and that the disk is functioning properly. If the problem continues, notify the system manager.

INVBASE, invalid BASE 'integer' specified.

Explanation: An invalid base has been specified either less than 1 or greater than 65535.

User action: Re-run the program with a valid base specified.

INVINCRMT, invalid increment 'integer' specified.

Explanation: An invalid increment has been specified either less than 1 or greater than 65535.

User action: Re-run the program with a valid increment specified.

INVLOWLMT, invalid LOWER_LIMIT 'integer' specified.

Explanation: An invalid lower limit has been specified either less than 1 or greater than 65535.

User action: Re-run the program with a valid lower limit specified.

INVUPLMT, invalid UPPER_LIMIT 'integer' specified.

Explanation: An invalid upper limit has been specified either less than 1 or greater than 65535.

User action: Re-run the program with a valid upper limit specified.

LOWGTRUP, LOWER_LIMIT 'integer' greater than UPPER_LIMIT 'integer'.

Explanation: The specified lower limit is greater than the specified upper limit.

User action: Re-run the program with a lower limit less than the upper limit.

LOWEQUP, LOWER_LIMIT 'integer' equal to UPPER_LIMIT 'integer'.

Explanation: The specified lower limit is equal to the specified upper limit.

User action: Re-run the program with a lower limit less than the upper limit.

UNEXPEOF, Unexpected end of IFF file

Explanation: This message indicates there is something seriously wrong with the IFF file which has caused immediate termination of the program. IRENUMBER has detected the end of the IFF file, but has not detected an IFF 'integer' entry.

User action: Use IMEND on the file, which will correctly position the EOF marker and insert an EJ entry at the end of the file. Re-run IRENUMBER on the corrected file.

MESSAGES (OTHER)

In addition to the above messages which are generated by IRENUMBER itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 14

MODULE IREPORT

MODULE **IREPORT**

REPLACES Nothing, IREPORT is a new IMP utility.

FUNCTION

IREPORT gives information about the presence or absence of selected IFF entries within a file. Absence of the specified entry may be detected with the /ABSENT qualifier.

FORMAT

\$ IREPORT IFF-file-spec

Command qualifiers

/ABSENT
/[NO]ADDRESS
/FRT=file-spec
/IN_LAYER=(integer[,...])

/[NO]LITES2
/[NO]LOG
/OUTPUT[=file-spec]
/[NO]PRINTER
/REVISION_LEVEL=integer(-1,0,1)

/WITH_FC=(integer[,...])

Defaults

Report presence.
/NOADDRESS
No FRT file read.
IFF entries in all layers considered.
/NOLITES2
/NOLOG
Output to SYS\$OUTPUT:.
/NOPRINTER
-1, all CB, ST and ZS entries are treated as themselves.
Features with all feature codes considered.

IFF entry qualifiers

/AC[=(integer[,...])]
/CB
/CC
/CP
/FC[=(integer[,...])]
/LAYER[=(integer[,...])]

/NF
/NS
/RO
/ST
/TC
/TH
/TS[=(integer[,...])]
/TX
/ZS

All layers reported unless
/IN_LAYER specified

If IREPORT is run with no IFF entry qualifiers then nothing will be sent to SYS\$OUTPUT:.

PROMPT

_IFF-File: IFF-file-spec

PARAMETER

file-spec

- specifies the IFF file from which information is required. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

COMMAND QUALIFIERS

/ABSENT

- report the absence of the selected IFF entry(s) from any IFF feature. (Not available for use with the /NS, /CP or /CC qualifiers).

/ADDRESS

/NOADDRESS

- messages to include the hexadecimal addresses of IFF entries

/FRT=file-spec

- causes the specified FRT file to be read during the initialisation sequence. Any AC types defined in it which are subsequently encountered in the IFF file and are to be reported on, will have their natures described according to the definitions in the FRT file. An incomplete file specification will default using 'LSL\$FRT:.FRT'.

/IN_LAYER=integer[,...] or /IN_LAYER=(integer,integer[,...])

- report on the presence or absence of IFF entries only within the specified layers. Ranges may be specified. The specified layers are also reported when found. A maximum of 1024 layer numbers may be specified. Layer numbers must lie in the range 0 to 32767.

Not valid for use with /LAYER, /CC, /CP or /NS qualifiers.

/LITES2

/NOLITES2

- creates a LITES2 command file to take the user to potential errors. Messages generated to report a potential error are incorporated in the file. By default the LITES2 command file specification is parsed against that of the input IFF file but with the substitution of logical name LSL\$LITES2CMD: and the extension '.LCM'. Thus if the input IFF file is called LSL\$IF:TST.IFJ then the default LITES2 command file is LSL\$LITES2CMD:TST.LCM.

LITES2 command sequences produced by IREPORT take the following typical form:

```
%MESSAGE FSN 670 (55) has no type 3 AC
%POSITION      765.3      456.89
%TEST $CURSINWIN
%ELSE %ZOOM 1
%PING
%RESPOND
%ABANDON
%ABANDON
```

/LITES2 is not available with the /ST, /ZS, /CB, /NS, /CC, /CP, or /LAYER qualifiers.

The LITES2 command file contains a standard Laser-Scan LITES2 command file header which, in addition to run time information, contains commands for compatibility with the LITES2 command file sorting utility LCMORG:

```
%POSITION      -0.730      -0.157
%POSITION      400.183      401.083
%ABANDON
%MESSAGE
%MESSAGE          L I T E S 2      C O M M A N D      F I L E
%MESSAGE
%MESSAGE          created by
%MESSAGE
%MESSAGE ===== I R E P O R T =====
%MESSAGE
%MESSAGE IREPORT invoked by TIM using terminal LTA196: at 10-JUN-1987 12:09:47.1
%MESSAGE
%MESSAGE Command line:
%MESSAGE
%MESSAGE IREPORT/LITES2=MISS_CODED OEEPECON/FC=5
%MESSAGE
%MESSAGE =====
%MESSAGE
%ABANDON
%ABANDON

/LOG
/NOLOG
```

- produces supplementary messages to SYS\$OUTPUT.

/OUTPUT[=file-spec]

- by default output will be to SYS\$OUTPUT:. The /OUTPUT qualifier enables the user to redirect output to a specified file. If the optional /OUTPUT file-spec argument is omitted, output will be to SYS\$DISK:IREPORT.LIS;.

/PRINTER
/NOPRINTER

- queues the IREPORT output for printing under the name given by the /OUTPUT qualifier. If you specify /PRINTER without the /OUTPUT qualifier, the output is directed to a file named SYS\$DISK:IREPORT.LIS which is queued automatically for printing and then deleted.

/REVISION_LEVEL=integer (-1,0,1)

- sets the input revision level used when reading the input IFF file. The possible values are:
 - 1 IFF file is treated such that CB, ST and ZS entries will be read as themselves. This is the default setting.
 - 0 IFF file is treated such that ST and ZS entries are read as themselves, and CB entries are converted into ST and ZS entries.
 - 1 IFF file is treated such that ST and ZS entries are converted into CB entries, and CB entries are read as themselves.

/WITH_FC=integer or /WITH_FC=(integer,integer[,...])

- report on the presence/absence of IFF entries only within features with the specified feature codes. Ranges of feature codes may be specified. A maximum of 1024 feature codes may be specified. Feature codes must lie in the range 0 to 32767.

/WITH_FC is not valid for use with the /FC, /CC, /CP or /NS qualifiers.

IFF ENTRY QUALIFIERS

/AC[=integer or =(integer,integer[,...])]

- report on the presence of the specified types of AC (Ancillary Code). Ranges of AC types may be specified. A maximum of 1024 AC types may be specified. The nature of the ACs encountered will be described, possibly using definitions contained in any FRT file specified using /FRT.

If no arguments are supplied then the presence of all types of AC is reported. If arguments are supplied to the /AC qualifier then only the occurrence of the specified type of AC is reported. A maximum of 1024 AC types may be specified. AC types must lie in the range 0 to 32767.

```
(2)          AC found   :   Height (3), value : 101.123   in feature with FSN 2
          With text : "THIS WOULD BE CONTENTS OF A TYPE 3 AC ENTRY"
(7)          AC found   :   Secondary_FC (1), value : 0     in feature with FSN 7
          With text : "THIS WOULD BE CONTENTS OF A TYPE 1 AC ENTRY"
```

Feature with FSN 670 (55), feature code 4, has no type 3 AC

Feature with FSN 670 (55), feature code 6, has no AC entry

- report on the presence, summarise the header information of, and give the first point in all CB (Coordinate Block) entries. /CB provides the following typical output :

```
- graphical type = 1
- number of points = 77
- number of attributes per point = 3
- number of fixed attributes = 0
```

```
Z (93) : 15.466
X (91) : 25.22
Y (92) : 1043.0
```

- report on the presence, and list the contents of, all IFF CC (Cubic Coefficient) entries. /CC provides the following typical output:

[illegible]

/CP

- report on the presence, and list the contents of, all IFF CP (Control Point) entries. /CP provides the following typical output:

CP entry				
NW	0.0	200.0	0.0	200.0
SW	0.0	0.0	0.0	0.0
SE	200.0	0.0	200.0	0.0
NE	200.0	200.0	200.0	200.0

/FC[=integer or =(integer,integer[,...])]

- the /FC qualifier offers two functions:
 - o if no arguments are supplied then the presence of all FS (Feature Status) entries is reported,
 - o if arguments are supplied then only the occurrence of FS entries with the specified feature codes is reported. Ranges of feature codes may be specified. A maximum of 1024 feature codes may be specified. Feature codes must lie in the range 0 to 32767.

The following typical output is provided:

Feature code 7 found, in feature with FSN 3 (3)

If the /ABSENT qualifier is specified then the following typical output is generated if specific FC numbers are supplied:
If one feature code was specified:

Feature with FSN 670 (55) has no feature code 28

If more than one feature code was specified:

Feature with FSN 670 (55) has no feature code 1,6,28, or 78

If the /ABSENT qualifier is specified and /FC was specified with no arguments then the following typical output is generated:

Feature with FSN 670 (55) has no FS entry

/LAYER=integer[,...] or /LAYER=(integer,integer[,...])

- report on the presence of the specified layers. Ranges of layers may be specified. A maximum of 1024 layer numbers may be specified. Feature codes must lie in the range 0 to 32767.
- If /LAYER is not specified then all layers are reported unless IN_LAYER is present in which case only the relevant layers are reported. The following typical output is generated:

Layer 4 found

If the /ABSENT qualifier is specified then the following typical output is generated if specific layer numbers are supplied:

No Layer 4 found.

/NF

- report on the presence of every NF (New Feature) entry in the IFF file. IREPORT refers to NF entries by their FSN (Feature Serial Number) and ISN (Internal Sequence Number). /NF provides the following typical output:

Feature with FSN 23 (20), feature code 77 found

If the /ABSENT qualifier is specified then the following typical output is generated:

Layer 34 has no features - it is empty

/NS

- report on each NS (New Section) entry. This provides the following typical output:

NS entry: "Digitised by CLIVE, using LAJ Version 7.1"

/RO

- report on the presence, and list the contents of, all IFF RO (Rotation) entries. /RO provides the following typical output:

RO entry, value 45.8, found in feature with FSN 3 (3)

If the /ABSENT qualifier is specified then the following typical output is generated:

Feature with FSN 670 (55) has no RO entry

/ST

- report on the presence, and list the contents of, all IFF ST (String) entries. /ST provides the following typical output:

Feature with FSN 3 (3), feature code 3, has 2 STs containing 93 points
First point = (765.22, 876.818), Last point = (456.223, 887.66)

If a feature has only one string with a single point then the following typical output is generated:

Feature with FSN 9 (9), feature code 5, has a string with only one point
Point = (400.064, 29.397)

If the /ABSENT qualifier is specified then the following typical output is generated:

Feature with FSN 670 (55) has no ST entry

/TC

- report on the presence, and list the contents of, all IFF TC (Transmitted Comment) entries.

If the TC entry is outside the feature then the following typical output is generated:

TC found before feature with FSN 199 (199)
"THIS WOULD BE THE CONTENTS OF A TC ENTRY"

If a TC is found inside a feature then the following typical output is generated:

TC entry found inside feature with FSN 3 (3)
"THIS WOULD BE THE CONTENTS OF THE TC ENTRY"

If the /ABSENT qualifier is specified then the following typical output is generated:

Feature with FSN 670 (670), feature code 5, has no TC entry

Output of this type is produced if no TC entries are found either before or inside the feature.

/TH

- report on the presence, and list the contents of, all IFF TH ("line THickness" or "Text Height") entries. /TH provides the following typical output:

TH entry, value 20, found in feature with FSN 3 (3)

If the /ABSENT qualifier is specified then the following typical output is generated:

Feature with FSN 670 (675) has no TH entry

/TX

- report on the presence, and list the contents of, all IFF TX TeXt) entries. /TX provides the following typical output:

Text entry found in feature with FSN 3 (3), rotation 5.2
"THIS IS AN EXAMPLE LINE OF TEXT"

If the /ABSENT qualifier is specified then the following typical output is generated:

Feature with FSN 670 (675) has no TX entry

/TS[=integer or =(integer,integer[,...])]

- the /TS qualifier offers two functions:
 - o if no arguments are supplied then the presence of all TS (Text Status) entries is reported,
 - o if arguments are supplied then only the occurrence of TS entries with the specified TCC (Text Component Codes) is reported. Ranges of TCCs may be specified. A maximum of 1024 TCCs may be specified. TCCs must lie in the range 0 to 32767.

The following typical output is provided:

Text status entry, TCC value 7, found in feature with FSN 3 (3)

If the /ABSENT qualifier is specified then the following typical output is generated if specific TS numbers are supplied:

If no arguments supplied:

Feature with FSN 670 (55) has no TS entry

If more than one TCC numbers specified:

Feature with FSN 670 (55) has no TS entry with TCC value 28

.
.
.

/ZS

- report on the presence, and list the contents of, all IFF ZS (three dimensional string) entries. /ZS provides the following typical output:

Feature with FSN 3 (3), feature code 3 has 2 ZSs containing 93 points
First point = (75.22, 76.818, 3.0), Last point = (56.223, 87.66, 9.7)

If a feature has only one string with a single point then the following typical output is generated.

Feature with FSN 6 (6), feature code 5, has a ZS entry with only one

point

Point = (400.064, 29.397, 78.522)

If the /ABSENT qualifier is specified then the following typical output is generated:

Feature with FSN 670 (55) has no ZS entry

RESTRICTIONS

- o /ABSENT and /CP are mutually exclusive
- o /ABSENT and /CC are mutually exclusive
- o /IN_LAYER and /LAYER are mutually exclusive
- o /IN_LAYER and /CC are mutually exclusive
- o /IN_LAYER and /CP are mutually exclusive
- o /IN_LAYER and /NS are mutually exclusive
- o /NS and /ABSENT are mutually exclusive
- o /LITES2 and /CC are mutually exclusive
- o /LITES2 and /CP are mutually exclusive
- o /LITES2 and /NS are mutually exclusive
- o /LITES2 and /LAYER are mutually exclusive
- o /LITES2 and /ST are mutually exclusive
- o /LITES2 and /CB are mutually exclusive
- o /WITH_FC and /CC are mutually exclusive
- o /WITH_FC and /CP are mutually exclusive
- o /WITH_FC and /NS are mutually exclusive
- o /WITH_FC and /FC are mutually exclusive
- o /WITH_FC and /LAYER are mutually exclusive

DESCRIPTION

IREPORT gives information about every occurrence of IFF entries specified on the command line using IFF entry qualifiers. Absence of the specified entry may be detected at the layer and feature level with the /ABSENT qualifier.

Reports on the presence or absence of IFF entries may be restricted to those lying within specified layers (/IN_LAYER q.v.). Similarly, reports on the presence or absence of IFF entries may be restricted to those which lie within features with specified feature codes (/WITH_FC q.v.).

By default IREPORT output is to SYS\$OUTPUT:. Output may, however, be redirected to file or to printer with the /OUTPUT and /PRINTER qualifiers respectively.

If output is directed to file or printer IREPORT adds header information to the output to facilitate identification and interpretation of the output. This header output is of the form:

```
===== IREPORT =====
IREPORT invoked by TIM using terminal LTA61: at 3-DEC-1986 10:48:14.75

IFF file: DUA0:[IFF.IMP.IFF]TST3D.IFF;4

Command line:
IREPORT/OUTPUT TST3D.IFF/LAYER/ADDRESS
```

```
=====

... IREPORT IFF entry output begins here ...
```

IREPORT may be directed to produce a LITES2 command file to enable semi-automatic examination and edit of reported features. Messages generated to report the presence or absence of IFF features are incorporated in the file. LITES2 command files are not available for use with the /ST, /NS, /CC, /CP, or /LAYER qualifiers.

EXAMPLES

```
$ IREPORT TST.IFF/LOG/LAYER/ADDRESS<CR>
%LSLLIB-I-IFFOPENED, DUA0:[BUREAU]TST.IFF;8 opened for read
Layer 1 found at address 000001EA and ends at address 00000474
Layer 88 found at address 00000475 and ends at address 000004D5
Layer 66 found at address 000004D6 and ends at address 00000785
Layer 55 found at address 00000786 and ends at address 0000085B
Layer 44 found at address 0000085C and ends at address 00000954
Layer 11 found at address 00000955 and ends at address 0000095A
ELAPSED: 00:00:05.33 CPU: 0:00:00.76 BUFIO: 5 DIRIO: 21 FAULTS: 73
```

\$

In this example the /LAYER qualifier has been selected with no arguments and therefore all layers are reported. Because /OUTPUT and /PRINTER were not specified output defaults to SYS\$OUTPUT:. As the /ADDRESS qualifier was specified the start and end addresses of each layer are indicated. The /LOG qualifier has resulted in the output of a message informing the user of the full file specification of the input file. The run completed normally and \$STATUS is set to SS\$_NORMAL.

```
$ IREPORT TST.IFF/AC/OUTPUT<CR>
ELAPSED: 00:00:03.95 CPU: 0:00:00.83 BUFIO: 5 DIRIO: 20 FAULTS: 79
```

\$

In this example the /OUTPUT qualifier has been used to direct all output to a file having the specification SYS\$DISK:IREPORT.LIS;. As /AC was specified IREPORT will send to the output file information about every occurrence of AC entries. The run completed successfully. \$STATUS is set to SS\$_NORMAL.

```
$ IREPORT SEWER9.IFF/IN_LAYER=(2,6)/WITH_FC=12/TC/AC=(2:4)<CR>
Layer 2 found
Type 3 AC found, value 67.9, in feature with FSN 170 (170)
"Pumping station number 6"
Type 3 AC found, value 67.2, in feature with FSN 171 (171)
"Pumping station number 9"
Type 3 AC found, value 68.3, in feature with FSN 172 (172)
"Storm drain overflow"
Layer 6 found
ELAPSED: 00:00:04.09 CPU: 0:00:00.70 BUFIO: 5 DIRIO: 18 FAULTS: 102
```

\$

In this example IREPORT reports on AC and TC entries in the specified layers and within features with the specified feature codes. No TCs were present. \$STATUS is set to SS\$_NORMAL.

```
$ IREPORT SEWER9.IFF/IN_LAYER=(2,6)/WITH_FC=(123)/TC/AC=(2:4)/ABSENT<CR>
Layer 2 found
Feature with FSN 168 (168), feature code 73, has no Type 2 AC
Feature with FSN 169 (169), feature code 73, has no Type 2 AC
Feature with FSN 170 (115), feature code 73, has no Type 2 AC
Feature with FSN 171 (171), feature code 73, has no Type 2 AC
Feature with FSN 172 (117), feature code 73, has no Type 2 AC
Feature with FSN 172 (36) has no TC entry
Feature with FSN 173 (173), feature code 73, has no Type 2 AC
Feature with FSN 173 (173) has no TC entry
Feature with FSN 174 (174), feature code 73, has no Type 2 AC
Layer 6 found
ELAPSED: 00:00:05.49 CPU: 0:00:00.74 BUFIO: 5 DIRIO: 17 FAULTS: 104
$
```

The same qualifiers as the previous example have been specified. This time, however, /ABSENT has been specified. Note that the presence of the specified layers is still reported. Within those layers features having the feature codes specified using /WITH_FC are checked for absence of ACs in the range 2 to 4, and the absence of preceding TCs. A number of features lack these entries. \$STATUS is set to SS\$_NORMAL.

```
$ IREPORT TST.IFF/IN_LAYER=(1,66)/CC/LOG<CR>
%CLI-W-CONFLICT, illegal combination of command elements - check documentation
\IN_LAYER\
ELAPSED: 00:00:00.47 CPU: 0:00:00.13 BUFIO: 2 DIRIO: 0 FAULTS: 50
$
```

In this example both /CC and /IN_LAYER qualifiers have been specified. The /CC and /IN_LAYER qualifiers are mutually exclusive (see RESTRICTIONS). IREPORT aborts execution. \$STATUS is set to the indicated CLI (Command Line Interpreter) warning message condition code.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). They will only appear if /LOG has been specified on the command line.

OPOUT, "'file-spec'" opened for output

Explanation: Either /OUTPUT or /PRINTER have been selected. The specified file has been opened to receive IREPORT output.

User action: None.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation.

ERRCLOIFF, failed to close input IFF file "'file-spec'"

ERRCLOOUT, failed to close output file "'file-spec'"

Explanation: Both messages indicate that an error occurred while closing the specified file. They are often a result of insufficient space available on the output device.

User action: The accompanying messages will help to diagnose the problem.

ERROPOUT, error opening output file "'file-spec'"

Explanation: Although the IFF input file was opened without error the specified output file cannot be opened.

User action: Check that the indicated file-spec contains a valid device and directory specification. Check that you have write access to the specified directory. The accompanying LSLLIB and RMS error messages will help you to interpret the cause of the error.

FAILTRLG, failed to translate logical name LSL\$LITES2CMD

Explanation: IREPORT is unable to translate logical name LSL\$LITES2CMD which it needs to direct output to the LITES2 command file specified on the command line. The supplementary message issued in support of this message will indicate why the logical name could not be translated.

User action: Using the information given in the supplementary message ensure that logical name LSL\$LITES2CMD is correctly defined. If you are in any doubt about the purpose or correct definition of the logical name please consult your system manager. Do not arbitrarily define LSL\$LITES2CMD.

IFFERR, IFF error "'code'" occurred while closing output file

Explanation: An IFF internal error 'code' occurred while closing the output file. This should not usually occur.

User action: Please contact Laser-Scan.

INVALACT, 'integer' is an invalid /AC argument - value must lie in range
1 - 32767

Explanation: The specified AC type lies outside the indicated range for valid IFF AC types.

User action: Respecify the IREPORT command line ensuring that any AC type specifications lie within the range 1 - 32767.

INVALFC, 'integer' is an invalid feature code argument - value must lie in range
0 - 32767

Explanation: The specified feature code lies outside the indicated range for valid IFF feature codes.

User action: Respecify the IREPORT command line ensuring that any feature code specifications lie within the range 0 - 32767.

INVALLAY, 'integer' is an invalid layer argument - value must lie in range
1 - 32767

Explanation: The specified feature code lies outside the indicated range for valid IFF layer numbers.

User action: Respecify the IREPORT command line ensuring that any layer number specifications lie within the range 0 - 32767. (Remember that layer 0 is normally reserved for registration marks and grid features).

INVALREVLEVEL, 'integer' is an invalid revision level - value must be -1, 0 or 1

Explanation: The specified input revision level is not one of the three allowed.

User action: Respecify the IREPORT command line ensuring that any /REVISIONLEVEL qualifier has an allowable input revision level.

INVALTS, 'integer' is an invalid /TS argument - value must lie in range
0 - 32767

Explanation: The specified /TS argument lies outside the indicated range for valid IFF text component codes.

User action: Respecify the IREPORT command line ensuring that any text component code specifications lie within the range 0 - 32767.

TOOMNYFC, too many feature code selections for /ABSENT - maximum 'integer'

Explanation: If /TS is specified together with /ABSENT IRECODE has to buffer up all the specified feature codes not found in a feature. IRECODE has run out of internal feature code storage.

User action: Respecify the command line with fewer feature code selections.

TOOMNYLAY, too many layers for /ABSENT - maximum 'integer'

Explanation: If /LAYER is specified together with /ABSENT IRECODE has to buffer up all the specified layers not found in the IFF file. IRECODE has run out of internal layer number storage.

User action: Respecify the command line with fewer layer selections.

TOOMNYTS, too many TSs in feature for /ABSENT - maximum 'integer'

Explanation: If /TS is specified together with /ABSENT IRECODE has to buffer up all the specified TCC codes not found in a feature. IRECODE has run out of internal TCC code storage.

User action: Respecify the command line with fewer TCC code selections.

UNEXPEOF, unexpected end of IFF file

Explanation: The specified input IFF file ended before an EJ entry was encountered.

User action: Only the valid part of the input IFF file was processed. Use IMEND (or ITOTEXT followed by IFROMTEXT) to repair the file.

MESSAGES (OTHER)

In addition to the above messages which are generated by IREPORT itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 15

MODULE ISELAC

MODULE ISELAC

REPLACES DAMP module SELAC

FUNCTION

ISELAC is a utility to read an IFF file and write selected features to a new IFF file. Feature selection is based on AC (Ancillary Code) type or on one or more of the attributes associated with an AC entry. ISELAC utilises a series of dynamic command tables created from a user-defined 'skeleton' file, to offer AC selection flexibility.

FORMAT

\$ ISELAC

COMMAND QUALIFIERS

None, ISELAC is command driven.

DESCRIPTION**General**

AC (Ancillary Code) entries in an IFF file are used to hold miscellaneous information not held by the feature code (Feature Status entry). Common AC entries hold information on heights, left and right boundary codes, and secondary feature codes. ISELAC is the IMP utility that enables selection by AC type, numerical AC attribute, Feature Serial Number, and layer, or any combination of these. Layer 0 (zero) the registration layer is always copied to the output file.

If no features are found in the input IFF file which match the selection criteria, the output file IFF is deleted.

ISELAC is an interactive, command driven utility. Command input is expected when the following prompt is issued:

ISELAC>

In certain circumstances the user can also be prompted for values.

IFF AC (Ancillary Code) entries

The format of an IFF AC Entry is a word (integer*2) type, a longword (integer*4 or real) code, and an optional text. The information held in an AC is determined by the AC type. The current AC type values are defined in the IFF User Guide.

Skeleton files

To function successfully ISELAC requires a 'skeleton' file to be set up to define all relevant AC types and attributes. A 'skeleton' file is a text file which specifies for each AC the:

- | | | |
|--------------------------------------|------|----------------|
| 1. AC name (character identifier) | e.g. | ICONTOUR |
| 2. AC type (numerical identifier) | e.g. | 2 |
| 3. AC code format | e.g. | Integer |
| 4. Prompt string | e.g. | Contour height |
| 5. AC text decriptor (if applicable) | | |

An default 'skeleton' file ACSKEL.DAT is supplied with the IMP package and should be present in the directory pointed to by the logical name LSL\$LOOKUP. If this is not found on program initialisation a warning is given. A copy of this file is shown below. The 'skeleton' file can be redefined during program execution using the SKELETON command. Skeleton files can be edited and created using any text editor. They are compatable with skeleton files for the DAMP SELAC program and the LSL DIGSYS manual digitising software.

```

!      EXAMPLE AC SKELTON FILE FOR IMP UTILITY ISELAC
!
!      Copyright Laser-Scan Ltd, Cambridge, England
!
!      Author: A T Timms          14-Oct-1986
!
!      Format of the file is:
!
! <AC name> (up to 24 characters)
! <AC type> <AC code format> <Prompt string> <Text descriptor>
!
! if the text descriptor is present a record of the associated text and
! numeric field is required:
! <numeric format> <text entry> <attribute prompt> <attrib. identifier>
! <range>
!
! "!" is a comment. Text after a "!" is ignored.
!
! Secondary FC
!
! SECFC          ! AC name
! 1 integer 'feature code' notext ! AC type , AC format , prompt, notext
!
! Integer contour height
!
! ICONTOUR       ! AC name
! 2 integer 'Contour height (Int)' ! integer, notext by default
!
! Real contour height

```

```

!
  RCONTOUR                                ! AC name
  3 real 'Contour height (Real)'          ! real, notext by default
!
! Left code
!
  LCODE
  4 integer 'Left code' notext
!
! Right code
!
  RCODE
  5 integer 'Right code' notext
!
! MCE dfad_point
!
  DFAD_POINT 7 integer=0 text              ! preset code to 0
%I1 typ 'type' <TYPE> [0]
%I6 hgt 'height (ms)' <HEIGHT> [-1022,1022]
%I4 fid 'feature id' <FEATURE_ID> [1,980]
%I2 sec 'security ' <SECURITY> [0]
%I2 rel 'releasability' <RELEASABILITY>
%I2 smc 'smc' <SMC> [1,13]
%I3 wid 'width' <WIDTH> [0 254]
%I3 lth 'length (ms)' <LENGTH> [0:254]
%I2 ori 'orientation (0 to 63)' <ORIENTATION> [0,63]
!
! MCE dfad_linear
!
  DFAD_LINEAR
  7 integer=0 text
%I1 typ 'type' <TYPE> [1]
%I6 hgt 'height (ms)' <HEIGHT> [-1022:1022]
%I4 fid 'feature_id' <FEATURE_ID> [1,980]
%I2 sec 'security' <SECURITY> [0 0]
%I2 rel 'releasability' <RELEASABILITY>
%I2 smc 'smc' <SMC> [1:13]
%I3 wid 'width' <WIDTH> [0,254]
%I1 dir 'directivity' <DIRECTIVITY> [1 3]
!
! MCE dfad_areal
!
  DFAD_AREAL
  7 integer=0 text
%I1 typ 'type' <TYPE> [2]
%I5 hgt 'height (ms)' <HEIGHT> [-1022,1022]
%I4 fid 'feature id' <FEATURE_ID> [1,980]
%I2 sec 'security' <SECURITY> [0]
%I2 rel 'releasability' <RELEASABILITY>
%I2 smc 'smc' <SMC> [1 13]
%I2 den 'density ' <DENSITY> [0,15]
%I3 rof 'roof coverage (%)' [0,100] <ROOF>
%I3 tre 'tree coverage (%)' [0,100] <TREE>
!
! MCE dfad_accuracy
!

```

```

DFAD_ACC
  8 integer=0 text
%I4 absv 'absolute vertical'    <ABS_VERTICAL>
%I4 absh 'absolute horizontal'  <ABS_HORIZ>
%I4 relv 'relative vertical'    <REL_VERTICAL>
%I4 relh 'relative horizontal' <REL_HORIZ>
!
! Gaspipes
!
  GASPIPE
  999 integer text
%I1 typ 'type' <TYPE> [0,2]
%I4 year 'year of installation' <YEAR> [1865:2100]
%I6 pre 'pressure' <PRESSURE> [-1000,1000]
%F6.2 dep 'depth (ms)' <DEPTH> [0.00,1000.00]
!
```

An exclamation mark '!' indicates a comment or comment line, and is ignored by ISELAC when reading from the skeleton file.

The AC name (character identifier) is used to select AC's. The name must consist of alphabetic characters or underlines, and may not exceed 24 characters. In the example skeleton file the following AC names are identified: SECFC, ICONTOUR, RCONTOUR, LCODE, RCODE, DFAD_POINT, DFAD_LINEAR, DFAD_AREAL, DFAD_ACC, and GASPIPE. Up to 20 AC names can be defined in one skeleton file.

The line following the AC name should have the format:

```
AC type [AC code format] ['Prompt string'] [Text indicator]
```

The AC type consists of one of the integer values in the table above, which is used by the program to identify an IFF AC entry of the specified name. This field is obligatory, the following three are optional.

The AC code format specifies whether the numerical value associated with the AC type is real or integer. In the case of a type 3 AC, the format is 'real' and for type 1 and 2 AC's the format is 'integer'. The value may be preset (ie. fixed) by supplying a value in the format:

```

[integer] [= n]
[real]    [= n]
```

where n is an integer or real as appropriate.

If only 'integer' or 'real' is specified, then the value is free to vary. In the example above the type 1,2,3,4 and 5 AC's (SECFC, ICONTOUR, RCONTOUR, LCODE and RCODE) have values that are free to vary but the MCE DFAD AC's (AC types 7 and 8) have the fixed AC code 0 (zero).

An optional prompt string may be supplied which must be enclosed within single quotes. It may be up to 24 characters long and is used when the user requests prompting for AC values (see section below on PROMPT mode). In the example skeleton file the prompt for type 2 AC's (ICONTOUR) is 'Contour height (Int)'.

The final field indicates whether the AC has associated text. The default value of this field is 'notext'. 'Text' should be specified if the AC has associated text entries. In the example file the type 999 AC's (GASPIPE) have associated text whereas the type 2 and 3 AC's (integer and real contour heights, ICONTOUR and RCONTOUR) do not.

For an AC with text, there then follows a number of records each defining a portion of the text and associated numerical field. The form of this record is:

```
numeric format  text entry 'attribute prompt' "<attribute identifier>"
"[range]"
```

The numeric format specifies whether the field is integer or real, the field width, and (for real) the number of digits after the decimal point.

This is done using:

1. %In to signify an integer field of n digits (filled with spaces on the left)
2. %Fn.m to signify a real field of total width n having m digits after the decimal point

The text entry defines the characters that will precede the numeric field in the AC entry in the IFF file. There is no restriction on the number of characters, but spaces are not allowed.

The three remaining parts of the record are distinguished by having different delimiters. These may occur in any order:

The attribute prompt is enclosed in single quotes and may be up to 24 characters.

The attribute identifier is the name associated with the field, and is enclosed in angle brackets. It is also restricted to 24 characters although only 12 characters can be displayed by the HELP ATTRIBUTE command.

The range, which is optional, sets the lower and upper limits for the numeric field, enclosed in square brackets and separated by a space, comma or colon. A single number enclosed in square brackets is interpreted as a preset value. ISELAC checks that values supplied by the user lie within these limits.

In the example skeleton file the type 999 AC (GASPIPE) has four possible associated attributes:

1. Identified as TYPE - a one digit integer between 0 and 2 which will be prompted for with "type", and exist in the AC entry as "typ".
2. Identified as YEAR - a four digit integer between 1865 and 2100 which will be prompted for with "year of installation", and exist in the AC entry as "year".
3. Identified as PRESSURE - a six digit integer between -1000 and +1000 which will be prompted for with "pressure", and exist in the AC entry as "pre".
4. Identified as DEPTH - a real number with two decimal places between 0.00 and 1000.00 which will be prompted for with "depth (ms)", and exist in the AC entry as "dep".

Selecting AC names or attribute names

Specific AC names or AC attributes (as defined in the skeleton file) can be selected from the input file by simply typing the name of the AC or attribute. It is also possible to specify ranges of AC values, Feature Serial Numbers, and Layers for selection and deselection. A feature is copied to the output file if an AC for that feature matches the AC selections. All AC's within a feature are checked and if one matches the specified selections the feature is copied.

All AC names, attribute names, the FSN and LAYER commands take one or more optional numerical arguments in one of the following forms:

1. Single values:

AC NAME	[inequality]	[value]
ATTRIBUTE NAME	[inequality]	[value]
FSN	[inequality]	[value]
LAYER	[inequality]	[value]
2. Range of values:

AC NAME	[inequality]	value	:	value2
ATTRIBUTE NAME	[inequality]	value	:	value2
FSN	[inequality]	value	:	value2
LAYER	[inequality]	value	:	value2
3. Multiple values:

AC NAME	[inequality]	value1	:	value2	AND	[inequality]	value3
ATTRIBUTE	[inequality]	value1	:	value2	AND	[inequality]	value3
FSN	[inequality]	value1	:	value2	AND	[inequality]	value3
LAYER	[inequality]	value1	:	value2	AND	[inequality]	value3

A range is specified by the use of ":" . The inequality NOT is used to exclude a specific value or range of values. A number of separate values or separate ranges of values, may be specified by use of the AND command. Care should be taken when specifying ranges of REAL

values to ensure that all required values are specified in the range. The difficulties encountered by the comparison of real numbers could mean that specifying a range of 200 : 300 for Real contour heights may not result in all contours with heights of 200.0 and 300.0 being selected. If such a range was required 199.999 : 300.001 would be more correct.

Possible inequalities are:

Inequalities:	Meaning:
=	.EQ. Equals
>	.GT. Greater than
>=	.GE. Greater than or equal to
<	.LT. Less than
<=	.LE. Less than or equal to
NOT	Not equal to i.e. exclude

The omission of an inequality implies equals "=".

A value may be an integer or real depending on the format for the specified in the 'skeleton' file. If a real value is entered for an attribute requiring an integer value, the value is converted to integer.

The attribute arguments are validated against the upper and lower limits or fixed values defined in the 'skeleton' file. An error message is given (with the valid range) if any of the entered values lie outside the defined range.

If the attribute name is supplied without numerical arguments, a range of values defined by the upper and lower limits in the 'skeleton' file is set.

Up to 10 different values or ranges of values for inclusion or exclusion may be specified for each AC name, attribute name FSN or LAYER. No checks for consistency are made, and the use of the SHOW command is recommended to display the selections prior to issuing the GO command. Selections can be cancelled by the RESET command.

The following examples (based on the default skeleton file) illustrate some of the various possibilities:

```
ISELAC> GASPIPE
will select all GASPIPE AC's (AC type 999)
```

```
ISELAC> ICONTOUR 100
will select integer contours (AC type 2) with a value of 100
```

```
ISELAC> WIDTH
will select widths lying between the upper and lower limits defined
in the 'skeleton' file
```

```
ISELAC> WIDTH 20 : 30
will select widths between 20 and 30 if one of the DFAD AC's
```


has been previously selected.

ISELAC> WIDTH NOT 20 : 30
will deselect widths between 20 to 30 (i.e. select widths outside the range 20 to 30), if one of the DFAD AC's has been previously selected.

ISELAC> RCONTOUR 0.0 : 1000.0 AND NOT 499.99 : 700.9
will select real contour values between 0.0 and 1000.0 excluding 499.99 to 700.9

ISELAC> FSN > 100
will select all features with an FSN (Feature Serial Number) greater than 100 (101 to 65535)

ISELAC> LAYER 1 : 3 AND 7 : 20 AND NOT 18
will select layers 1 to 3 and 7 to 20 excluding layer 18

ISELAC> YEAR .LT. 1900 AND .GE. 1940
will select years between 1865 and 1899 and 1904 to 2100 if GASPIPE has previously been selected. (1865 and 2100 are the maxima and minima for the year range specified in the skeleton file).

The selection of an AC name automatically cancels previous AC selections. Layer and FSN selections are retained.

Abbreviations

Commands, AC and ATTRIBUTE names can be abbreviated to their minimum non ambiguous form. Checks for ambiguities are made between the different commands and AC and attribute names. Warnings are given if commands are ambiguous.

ISELAC COMMANDS

AND

Separates values or ranges of values

FORMAT: AC NAME 'value' AND 'value'
ATTRIBUTE NAME 'value' AND 'value'
LAYER 'value' AND 'value'
FSN 'value' AND 'value'

COMMAND PARAMETERS: None.

DESCRIPTION:

The AND command separates values or ranges of values following an AC or attribute name, Layer or FSN specification.

Warning messages: None.

Examples:

ISELAC> FSN 1:100 AND NOT 50:60<CR>

ISELAC> SHOW<CR>

Current AC is ICONTOUR

AC selection is by:

FSN	including:	1	to	100
FSN	excluding:	50	to	60

ISELAC> GAS ?<CR>

type? 2<CR>

year of installation? 1908:1990<CR>

pressure? 10:100 AND 110:120<CR>

depth (ms)? <CR>

ISELAC> SHOW<CR>

Current AC is GASPIPE

AC selection is by:

TYPE	including:	2	to	2
YEAR	including:	1908	to	1990
PRESSURE	including:	10	to	100
PRESSURE	including:	110	to	120

ISELAC> DEPTH 10:100 AND NOT 50:60<CR>

ISELAC> SHOW<CR>

Current AC is GASPIPE

AC selection is by:

DEPTH	including:	10.000	to	100.000
DEPTH	excluding:	50.000	to	60.000

EXIT

Exit from ISELAC.

FORMAT: EXIT

Command parameters: None.

DESCRIPTION:

The EXIT command causes ISELAC to exit closing all IFF files.

<CTRL/Z> (pressing the Ctrl and Z keys together) may also be used to exit the program.

Warning messages: None.

Examples:

ISELAC> **EXIT<CR>**

ELAPSED: 00:05:25.84 CPU: 0:00:05.71 BUFIO: 281 DIRIO: 46 FAULTS: 263
\$

FROM

Selects the input IFF file.

FORMAT: FROM file-spec

Command parameters:

file-spec

The input IFF file.

DESCRIPTION:

Any part of the file name that is not specified will be completed from the default of "LSL\$IF:IFF.IFF;0". Any file previously opened for input are closed.

Warning messages:

The following warning messages are specific to the FROM command:

ERRPARFIL, error parsing filename

NOFILNAM, no filename specified

Examples:

ISELAC> FROM<CR>

%ISELAC-W-NOFILNAM, no filename specified

ISELAC> FROM BUR\$DISK:[BUREAU.TEST]101NW.IFF<CR>

%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TEST]101NW.IFF opened for read

ISELAC> FROM 101-NW--;<CR>

%ISELAC-W-ERRPARFIL, error parsing filename "101-NW--;"

FSN

Selects specified Feature Serial Numbers.

FORMAT: **FSN [value] [inequality] [range]**

Command parameters:

value

The required FSN.

inequality

One of the valid inequalities.

range

The required range of FSN's.

DESCRIPTION:

Feature selection (or de-selection) can be made on the basis of Feature Serial Number as well as AC. Any single value or range of values (within the valid IFF limits of 0 to 65535) can be specified according to the format specified in the section on AC name and attribute selection. FSN selections can be cancelled using the RESET command.

Warning messages:

The following warning messages are specific to the FSN command:

 INVALFSN, invalid value for FSN specified

Examples:

```
ISELAC> FSN<CR>
%ISELAC-W-MISSVALCMD, missing value on command line
ISELAC> FSN 1:1000<CR>
ISELAC> SHOW<CR>
Current AC is ICONTOUR
AC selection is by:
FSN                      including:          1          to          1000
ISELAC> FSN RESET<CR>
FSN selection reset
ISELAC> FSN 1:100 AND NOT 50:60<CR>
ISELAC> SHOW<CR>
Current AC is ICONTOUR
AC selection is by:
FSN                      including:          1          to          100
```

FSN	excluding:	50	to	60
ISELAC> FSN RESET <CR>				
FSN selection reset				
ISELAC> FSN >10 <CR>				
ISELAC> SHOW <CR>				
Current AC is ICONTOUR				
AC selection is by:				
FSN	including:	11	to	65535

GO

Terminates command input and activates the selection process.

FORMAT: GO

Command parameters: None.

DESCRIPTION:

The GO command terminates command input, and activates the selection process. Input and output IFF files must have been previously defined. A message is output as each layer encountered and a summary of the features selected or rejected made at the end of each layer. Command is returned to the user once the input file has been processed, for possible further selections.

Warning messages:

The following warning messages are specific to the GO command:

OUTFILNOSPEC, output file not specified

INFILNOSPEC, input file not specified

Examples:

```
ISELAC> GO<CR>
%ISELAC-W-OUTFILNOSPEC, output file not specified
ISELAC>
ISELAC> GO<CR>
%ISELAC-W-INFILNOSPEC, input file not specified
```

```
ISELAC> GO<CR>
%ISELAC-I-LAYPROC, layer 1 will be processed
0 features selected in layer 1
3 features rejected in layer 1

%ISELAC-I-LAYPROC, layer 4 will be processed
3 features selected in layer 4
32 features rejected in layer 4

%ISELAC-I-LAYCOP, layer 5 will be copied
%ISELAC-I-LAYPROC, layer 10 will be processed
6 features selected in layer 10
2 features rejected in layer 10
```

```
ISELAC>
```

HELP

Give help on a subject

FORMAT: **HELP subject**

Command parameters:

subject

The subject on which help is required

DESCRIPTION:

The HELP command looks the rest of the line up in the ISELAC HELP library. This library contains a brief summary of the operation of each command.

The information is looked up in the ISELAC section of the IMP help library, LSL\$HELP:IMP.HLB.

In addition HELP AC gives a list of all available AC names defined in the current skeleton file, and HELP ATTRIBUTES gives a list of all possible attributes for the currently selected AC name.

Warning messages:

Where required, warning messages are output via the VMS LBR\$OUTPUT_HELP utility.

Examples:

ISELAC> **HELP AC<CR>**
Valid AC names are:

DFAD_ACC	DFAD_AREAL	DFAD_LINEAR	DFAD_POINT	GASPIPE
ICONTOUR	LCODE	RCODE	RCONTOUR	SECFC

ISELAC> **GASPIPE<CR>**
%ISELAC-I-ACNAMSEL, GASPIPE selected
ISELAC> **HELP ATTRIB<CR>**

Valid attribute names for defined AC are:

DEPTH	PRESSURE	TYPE	YEAR
-------	----------	------	------

LAYER

Selects specified Layers.

FORMAT: **LAYER** [value] [inequality] [range]

Command parameters:

value

The required LAYER.

inequality

One of the valid inequalities.

range

The required range of Layers.

DESCRIPTION:

Feature selection (or de-selection) can be made on the basis of Layer as well as Feature Serial Number or AC. Any single value or range of values (within the valid IFF limits of 0 to 32767) can be specified according to the format specified in the section on AC name and attribute selection. The following rules apply to layer selection:

1. The registration layer 0 (zero) is always copied to the output file.
2. If no layers are selected or deselected then all layers are processed.
3. Layers that are selected are processed and all others are ignored.
4. Layers deselected are ignored and all others are processed.

Layer selections can be cancelled using the RESET command.

Warning messages:

The following warning messages are specific to the LAYER command:

INVALLAY, invalid value for LAYER specified

Examples:

```
ISELAC> LAYER 1<CR>
ISELAC> SHOW<CR>
Current AC is GASPIPE
```

AC selection is by:

YEAR	including:	1908	to	1990
PRESSURE	including:	10	to	100
Layer	including:	1	to	1

ISELAC> **LAYER RESET<CR>**

LAYER selection reset

ISELAC> **LAYER 1:10 AND 15:20 AND NOT 17<CR>**

ISELAC> **SHOW<CR>**

Current AC is GASPIPE

AC selection is by:

Layer	including:	1	to	10
Layer	including:	15	to	20
Layer	excluding:	17	to	17

ISELAC> **LAYER<CR>** %ISELAC-W-MISSVALCMD, missing value on command line

RESET

Cancels specified AC NAME, AC ATTRIBUTE, LAYER or FSN selection.

FORMAT:	AC NAME	RESET
	AC ATTRIBUTE	RESET
	LAYER	RESET
	FSN	RESET

Command parameters: None.

DESCRIPTION:

The RESET command following an AC NAME, AC ATTRIBUTE, LAYER or FSN causes the selection for that to be cancelled.

Warning messages:

The following warning messages are specific to the RESET command:

INVALRESET, invalid use of RESET command

Examples:

```
ISELAC> SHOW<CR>
Current AC is GASPIPE
AC selection is by:
TYPE                including:          2      to          2
YEAR                including:        1908      to          1990
PRESSURE            including:          10      to          100
PRESSURE            including:        110      to          120
Layer               including:           1      to           1
ISELAC> GAS RESET<CR>
GASPIPE and all associated attributes reset
ISELAC> SHOW<CR>
Current AC is GASPIPE
AC selection is by:
Layer               including:           1      to           1
ISELAC> LAYER RESET<CR>
LAYER selection reset
ISELAC> SHOW<CR>
Current AC is GASPIPE
AC selection is by:
ISELAC>
```

SHOW

Displays current selections.

FORMAT: SHOW

Command parameters: None.

DESCRIPTION:

The SHOW command displays the current selections including AC name, AC attribute, Layer and Feature Serial Number.

Warning messages: None.

Examples:

ISELAC>

ISELAC> **SHOW<CR>**

Current AC is GASPIPE

AC selection is by:

TYPE	including:	2	to	2
YEAR	including:	1908	to	1990
PRESSURE	including:	10	to	100
PRESSURE	including:	110	to	120

SKELETON

Defines the AC skeleton file to be used.

FORMAT: **SKELETON skeleton-file-spec**

Command parameters:

skeleton-file-spec

The skeleton file-specification to be used.

DESCRIPTION:

Any part of the file name that is not specified will be completed from the default of "LSL\$LOOKUP:ACSKEL.DAT"

Warning messages:

The following warning messages are specific to the SKELETON command:

ERROPSKL, error opening skeleton file 'file-spec'

ERRRDSK, error reading AC skeleton file

NOSKFILNAM, no skeleton filename specified

ERRRDACNAM, error reading AC name in skeleton file, line 'integer'

IDNTABS, identifier absent in skeleton file, line 'integer'

IDTNTTCLO, identifier not closed in skeleton file, line 'integer'

IDNTOOLNG, identifier too long in skeleton file, line 'integer'

INCOMPFLD, incomplete field in skeleton file, line 'integer'

INVACTYP, invalid AC type in skeleton file, line 'integer'

INVALKYWD, invalid keyword in skeleton file, line 'integer'

INVALRNG, invalid range in skeleton file, line 'integer'

PRMTNTTCLO, prompt not closed in skeleton file, line 'integer'

PRMTOOLNG, prompt too long in skeleton file, line 'integer'

RNGNTTCLO, range not closed in skeleton file, line 'integer'

UNRECLIN, unrecognised line 'integer' in skeleton file

Examples:

```
ISELAC>
ISELAC> SKELETON<CR>
%ISELAC-W-NOSKFILNAM, no skeleton filename specified
ISELAC> SKELETON AIQWUERO<CR>
%ISELAC-W-ERROPSKL, error opening skeleton file "LSL$LOOKUP:AIQWUERO.DAT"
-LSLLIB-E-NOSUCHFILE, file cannot be found
ISELAC> SKELETON ME:LOGIN.COM<CR>
%ISELAC-I-OPNSKL, Skeleton file ME:LOGIN.COM opened for input
%ISELAC-E-UNRECLIN, unrecognised line 1 in skeleton file
$!  Personal LOGIN.COM file
```

```
ISELAC> SKELETON DUA0:[IMP.ISELAC]ACSKEL.DAT<CR>
%ISELAC-I-OPNSKL, Skeleton file dua0:[imp.iselac]acskel.dat opened for input
ISELAC>
```

TO

Specifies the output IFF file.

FORMAT: **TO file-spec**

Command parameters:

file-spec

The output IFF file.

DESCRIPTION:

Any part of the file name that is not specified will be completed from the default of "LSL\$IF:ISELAC_DEF.IFF"

Warning messages:

The following warning messages are specific to the TO command:

ERRPARFIL, error parsing filename

NOFILNAM, no filename supplied

Examples:

ISELAC> TO

%ISELAC-W-NOFILENAM, no filename specified

ISELAC> TO LSL\$DISK:[LSL.TEST]RCONTSEL

%LSLLIB-I-IFFOPENED, LSL\$DISK:[LSL.TEST]RCONTSEL.IFF opened for write

ISELAC> TO qwe-uu--;

%ISELAC-W-ERRPARFIL, error parsing filename "qwe-uu--;"

?

Prompts for values for the specified AC name or attribute.

FORMAT: AC NAME ?

Command parameters: None.

DESCRIPTION:

The ? command following an AC NAME name causes the program to prompt for attribute values specified in the skeleton file with the prompts specified in the skeleton file. Attributes that are not required for selection can be ignored by typing carriage return to an attribute prompt.

Warning messages: None.

Examples:

```
ISELAC> gas ?<CR>
type? 3<CR>
%ISELAC-W-INVVALSPEC, invalid value specified for attribute "TYPE"
Valid range for TYPE is 0 to 2
type? 2<CR>
year of installation? 1908:1990<CR>
pressure? 10:100 AND 110:120<CR>
depth (ms)? <CR>
ISELAC>
ISELAC> SHOW<CR>
Current AC is GASPIPE
AC selection is by:
TYPE                including:          2      to          2
YEAR                including:        1908      to          1990
PRESSURE            including:          10      to          100
PRESSURE            including:        110      to          120
ISELAC>
```

EXAMPLES

```
$ ISELAC<CR>
%ISELAC-I-OPNSKL, Skeleton file LSL$LOOKUP:ACSKEL.DAT opened for input
ISELAC> FROM 153NW.IFF<CR>
%LSLLIB-I-IFFOPENED, DUA0:[TOM.FAILED]153NW.IFF;2 opened for read
ISELAC> TO 153NW.SEL<CR>
%LSLLIB-I-IFFOPENED, DUA0:[TOM.FAILED]:153NW.SEL;1 opened for write
ISELAC> RCONT 99.999:200.001 AND 299.99:300.001<CR>
ISELAC> LAY 1:10<CR>
ISELAC> SHOW<CR>
Current AC is RCONTOUR
AC selection is by:
AC field value      including:      99.999      to      200.001
AC field value      including:      299.990     to      300.001
Layer               including:           1      to      10
ISELAC> GO<CR>
%ISELAC-I-LAYPROC, layer 1 will be processed
0 features selected in layer 1
3 features rejected in layer 1
%ISELAC-I-LAYPROC, layer 2 will be processed
4 features selected in layer 2
2 features rejected in layer 2
%ISELAC-I-LAYPROC, layer 3 will be processed
1 feature selected in layer 3
0 features rejected in layer 3
%ISELAC-I-LAYPROC, layer 4 will be processed
23 features selected in layer 4
12 features rejected in layer 4
%ISELAC-I-LAYPROC, layer 5 will be processed
9 features selected in layer 5
5 features rejected in layer 5
%ISELAC-I-LAYPROC, layer 10 will be processed
2 features selected in layer 10
0 features rejected in layer 10
ISELAC> ICONT 200:300<CR>
ISELAC> <CR>
ISELAC> SHOW<CR>
Current AC is ICONTOUR
AC selection is by:
AC field value      including:      200      to      300
Layer               including:           1      to      10
ISELAC> <CR>
ISELAC> TO 153NW.INT<CR>
%LSLLIB-I-IFFOPENED, DUA0:[TOM.FAILED]153NW.INT;1 opened for write
ISELAC> SHOW<CR>
Current AC is ICONTOUR
AC selection is by:
AC field value      including:      200      to      300
Layer               including:           1      to      10
ISELAC> GO<CR>
%ISELAC-I-LAYPROC, layer 1 will be processed
0 features selected in layer 1
6 features rejected in layer 1
```

```
%ISELAC-I-LAYPROC, layer 2 will be processed
0 features selected in layer 2
6 features rejected in layer 2
```

```
%ISELAC-I-LAYPROC, layer 3 will be processed
0 features selected in layer 3
1 feature rejected in layer 3
```

```
%ISELAC-I-LAYPROC, layer 4 will be processed
1 feature selected in layer 4
34 features rejected in layer 4
```

```
%ISELAC-I-LAYPROC, layer 5 will be processed
1 feature selected in layer 5
13 features rejected in layer 5
```

```
%ISELAC-I-LAYPROC, layer 10 will be processed
0 features selected in layer 10
2 features rejected in layer 10
```

```
ISELAC> EXIT<CR>
```

```
ELAPSED:      0 00:03:53.57  CPU: 0:00:08.34  BUFIO: 82  DIRIO: 220  FAULTS: 227
```

This example uses the default skeleton file LSL\$LOOKUP:ACSKEL.DAT and successfully selects a range of real contours (type 3 AC's) and a range of integer contours (type 2 AC's) to separate files from one input file. The default skeleton file is found and read in at program initialisation. The input file (LSL\$IF:153NW.IFF;2) is defined using the FROM command and the output file (LSL\$IF:153NW.SEL;1) is defined using the TO command. Real contours (type 3 AC's) in the range 99.999 to 200.001 and 299.99 to 300.001 are selected by typing the AC name RCONT(OUR) followed by the required ranges of heights (separated by AND). Layers 1 to 10 are selected using the LAYER command followed by the range. The SHOW command confirms the selections and the GO command starts the selection process. A message is output as each layer is encountered and a summary of the number of features selected and rejected is output at the end of each layer.

On completion of the selection process on the input file the user is prompted again with the ISELAC prompt. The user now selects integer contours (type 2 ACs) within the range 200 to 300 with the command ICONT with the specified range. The SHOW command confirms the selection. Note the LAYER selection is retained from the previous run, but all RCONTOUR selections are cancelled with the ICONTOUR command. A new output file (LSL\$IF:153NW.INT;1) is selected with the TO command and the GO command starts the selection process with selected features copied to the new output file. Again a message is output as each layer is encountered and a summary of the number of features selected and rejected is output at the end of each layer. Program execution is terminated with the exit command.

```
$ ISELAC<CR>
```

```
%ISELAC-I-OPNSKL, Skeleton file LSL$LOOKUP:ACSKEL.DAT;77 opened for input
```

```
ISELAC> <CR>
```

```
ISELAC> SKEL ROAD.SKL<CR>
```

```
%ISELAC-W-ERROPSKL, error opening skeleton file "LSL$LOOKUP:ROAD.SKL;"
```

```
-LSLLIB-E-NOSUCHFILE, file cannot be found
```

```
ISELAC> SKEL IF:ROAD.SKL<CR>
```

```
%ISELAC-I-OPNSKL, Skeleton file IF:ROAD.SKL;6 opened for input
```

ISELAC> **HELP AC**<CR>
Valid AC names are:

ROAD

ISELAC> **ROAD 1:2**<CR>
ISELAC> **SHOW**<CR>
Current AC is ROAD
AC selection is by:
AC field value including: 1 to 2
ISELAC> **FROM MAP1.LNK**<CR>
%LSLLIB-I-IFFOPENED, DUA0:[TOM.FAILED]MAP1.LNK;1 opened for read
ISELAC> **TO ONEWAYST.IFF**<CR>
%LSLLIB-I-IFFOPENED, DUA0:[TOM.FAILED]ONEWAYST.IFF;2 opened for write
ISELAC> **GO**<CR>
%ISELAC-I-LAYCOP, layer 0 will be copied
%ISELAC-I-LAYPROC, layer 1 will be processed
12 features selected in layer 1
248 features rejected in layer 1

ISELAC> **ROAD RESET**<CR>
ROAD and all associated attributes reset
ISELAC> **ROAD 0**<CR>
ISELAC> **SHOW**<CR>
Current AC is ROAD
AC selection is by:
AC field value including: 0 to 0
ISELAC> **TO SINGLERD.IFF**<CR>
%LSLLIB-I-IFFOPENED, DUA0:[TOM.FAILED]SINGLERD.IFF;9 opened for write
ISELAC> **GO**<CR>
%ISELAC-I-LAYCOP, layer 0 will be copied
%ISELAC-I-LAYPROC, layer 1 will be processed
241 features selected in layer 1
19 features rejected in layer 1

ISELAC> **<CTRL/Z>**
ELAPSED: 0 00:03:24.47 CPU: 0:00:30.63 BUFIO: 53 DIRIO: 587 FAULTS: 223

This example shows a successful run of ISELAC using a simple user defined skeleton file to select two different codes from features with AC type 160. This simple skeleton file is shown below.

```
!  
! Skeleton file to select type 160 AC's  
!  
  ROAD  
  160 integer notext  
!
```

The default skeleton file LSL\$LOOKUP:ACSKEL.DAT is found and read in at program initialisation. The user attempts to define the new skeleton file using the SKELETON command but the file could not be found in LSL\$LOOKUP, and a message is output to this effect. The SKELETON command is reissued with a correct file specification and the new skeleton file opened successfully. The HELP AC command verifies the possible AC selection is ROAD and the user selects AC codes 1 and 2 by issuing the ROAD AC name followed by the required range. The SHOW command verifies this selection. The input and output files are then

specified using the FROM and TO commands and both files are opened successfully. The GO command starts the selection process and messages are output as each layer is encountered. When the selection process is completed the AC code selections are cancelled using the RESET command and a new selection input using the ROAD AC name followed by the required value 0 (zero). The SHOW command again demonstrates the validity of the selections. A new output file is specified with the TO command and the selection process started with the GO command. On completion of the selection process <CTRL/Z> causes ISELAC to exit tidily.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user.

Explanation: The AC type specified by the name 'AC name' in the skeleton file has been selected without any arguments.

User action: None.

ATTRNAMSEL, 'ATTRIBUTE name' selected

Explanation: The AC attribute specified by the name 'ATTRIBUTE name' has been selected without any arguments.

User action: None.

LAYCOP, layer 'integer' will be copied

Explanation: Layer 'integer' has been found and will be copied

User action: None.

LAYPROC, layer 'integer' will be processed

Explanation: Layer 'integer' has been found and will be processed

User action: None.

LAYSKIP, layer 'integer' will be skipped

Explanation: Layer 'integer' has been found and will be skipped

User action: None.

OPNSKL, skeleton file 'file-spec' opened for input

Explanation: The specified skeleton file has been successfully opened.

User action: None.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

AMBIGCMD, ambiguous between 'cmd1' and 'cmd2'

Explanation: The command, AC or ATTRIBUTE name typed is ambiguous between 'cmd1' and 'cmd2'.

User action: Add sufficient letters to the typed command to make it non-ambiguous.

ERROPSKL, error opening skeleton file 'file-spec'

Explanation: The specified file cannot be opened for input. This message will usually be followed by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. Check the file specification for errors, check that logical name LSL\$LOOKUP: has been assigned to the correct device and directory, check that the file and directory are not protected against write access. If the problem continues, notify the system manager.

ERRPARFIL, error parsing filename 'file-spec'

Explanation: The LSLLIB file name parsing routine has failed to parse 'file-spec'.

User action: Check for the use of special characters and version numbers, and the length of the filename specified.

ERRRDSK, error reading AC skeleton file

Explanation: An error has occurred reading the AC skeleton file. This message will usually be followed by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. Check the file specification for errors, check that logical name LSL\$LOOKUP: has been assigned to the correct device and directory, check that the file and directory are not protected against read access. If the problem continues, notify the system manager.

ERRSPECARG, error specifying attribute argument

Explanation: A non numeric response to a prompt for a numeric AC attribute value has been given.

User action: Give a numeric response to the prompt.

MISSVALCMD, missing value on command line

Explanation: A numeric value is missing on the command line.

User action: Check that inequalities and ranges are correctly specified. Check the selected values with the SHOW command as erroneous values may now be present. RESET as necessary and retype the corrected command line.

INFILNOSPEC, input file not specified

Explanation: The GO command has been given but no input file has yet been specified.

User action: Use the FROM command to specify an input file then re-specify the GO command.

INVALFSN, invalid value for FSN specified

Explanation: A Feature Serial Number outside the valid IFF range of 0 (zero) to 65535 has been specified.

User action: Retype command line with valid FSNs.

INVALLAY, invalid value for LAYER specified

Explanation: A Layer outside the valid IFF range of 0 (zero) to 32767 has been specified.

User action: Retype command line with valid layers.

INVALRESET, invalid use of RESET command

Explanation: The RESET command has been given after an AND command. It is only valid after an AC or Attribute name.

User action: Check the selected values with the SHOW command as erroneous values may now be present. Use the RESET command with a valid AC or Attribute name.

INVVALSPEC, invalid value specified for attribute 'attrib'

Explanation: A value has been given for AC attribute 'attrib' that lies outside the range specified in the AC skeleton file. An additional message should give the valid maximum and minimum values for 'attrib'.

User action: Give a valid value or range of values. Check the selected values with the SHOW command as erroneous values may now be present.

NOFILNAM, no filename specified

Explanation: The FROM or TO command has been given with no file specification.

User action: Retype the command with a file specification

NOSKFILNAM, no skeleton filename specified

Explanation: The SKELETON command has been given with no file specification.

User action: Retype the command with a file specification

OUTFILNOSPEC, output file not specified

Explanation: The GO command has been given but no output file has yet been specified.

User action: Use the TO command to specify an output file then give the GO command.

TOOMANLIM, too many limits specified

Explanation: The limit of 10 selected values or ranges of values has been reached for an AC or Attribute name, Layer or FSN.

User action: Use the SHOW command to check that the selected limits are correct. Either repeat the selection process several times to allow for the number of limits required, or, check that inequalities are being used to their maximum effect.

UNRECCMDAC, unrecognised command or AC name 'cmd'

Explanation: An unrecognised command or AC or Attribute name 'cmd' has been given that cannot be found in any of the command tables. This message will also be given if there has been invalid use of an inequality after a valid AC or Attribute name.

User action: Check the validity and spelling of 'cmd' and any inequalities. Check the AC or Attribute name is in the current AC skeleton file (use HELP AC and HELP ATTRIBUTE).

UNEXPEOCL, unexpected end of command line

Explanation: A numeric value or inequality is missing at the end of the command line after an AND or NOT for instance.

User action: Check the selected values with the SHOW command as erroneous values may now be present. RESET as necessary and retype the corrected command line

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ADDNAMERR, error adding name to command tables

Explanation: An error has occurred adding an AC or Attribute name to the dynamic command tables used by the program.

User action: Check that there are no more than 20 AC names defined in the skeleton file. If the problem continues notify the system manager and report the circumstances to LSL.

ERROPCMD, error opening command tables

Explanation: An error has occurred creating a command table or adding a name to it.

User action: Check that there are no more than 20 AC names defined in the skeleton file. If the problem continues notify the system manager and report the circumstances to LSL.

ERRRDACNAM, error reading AC name in skeleton file, line 'integer'

Explanation: An error has occurred reading an AC name in the skeleton file at line 'integer'. The offending line should be output after the message.

User action: Check the validity of the line in the skeleton file, edit as necessary and rerun ISELAC.

FTOREPTONV, failed to reposition to non-void entry

Explanation: An error has occurred repositioning an IFF marker to a previously noted position. This is likely to be an internal error in the program.

User action: Report circumstances to LSL.

IDNTABS, identifier absent in skeleton file, line 'integer'

Explanation: The identifier for an AC attribute is missing in the AC skeleton file in line 'integer'. The offending line should be output after the message.

User action: Add the missing identifier and rerun ISELAC.

IDNTNTCLO, identifier not closed in skeleton file, line 'integer'

Explanation: The AC attribute identifier specified in line 'integer' of the skeleton file has no closing angle bracket '>' or is greater than 24 characters long. The offending line should be output after the message.

User action: Edit the skeleton file to correct the error and rerun ISELAC.

IDNTOOLNG, identifier too long in skeleton file, line 'integer'

Explanation: The AC attribute identifier specified in line 'integer' of the skeleton file is longer than 24 characters. The offending line should be output after the message.

User action: Edit the skeleton file to correct the error and rerun ISELAC.

INCOMPFLD, incomplete field in skeleton file, line 'integer'

Explanation: The numeric format field in the skeleton file, line 'integer', is incomplete.

User action: Edit the skeleton file to correct the error and rerun ISELAC.

INVACTYP, invalid AC type in skeleton file, line 'integer'

Explanation: An invalid AC type has been specified in the skeleton file at line 'integer'. Either no AC name has been specified or there is an error reading the AC type

User action: Edit the skeleton file to correct the error and rerun ISELAC.

INVALTXT, invalid text entry in skeleton file, line 'integer'

Explanation: An error has occurred reading the text entry for an AC attribute in line 'integer' of the skeleton file which should be output after the message.

User action: Edit the skeleton file to correct the error and rerun ISELAC.

INVALFRMT, invalid format in skeleton file, line 'integer'

Explanation: An invalid numeric format has been specified in line 'integer' of the skeleton file which should be output after the error message.

User action: Edit the skeleton file to correct the error and rerun ISELAC.

INVALKYWD, invalid keyword in skeleton file, line 'integer'

Explanation: An unrecognised entry has been found in line 'integer' of the skeleton file. The offending line should be output after the error message.

User action: Check the spelling of keywords in the specified line. Edit the skeleton file to correct the error and rerun ISELAC

INVALRNG, invalid range in skeleton file, line 'integer'

Explanation: An error has occurred reading the range entry of an AC Attribute in line 'integer' of the skeleton file. The line should be output after the message.

User action: Check the validity of the specified range, edit the skeleton file and rerun ISELAC.

NOSELFND, no selected items found in input file

Explanation: ISELAC found no matches between the selection criteria and the features within the input IFF file.

User action: Check that the selection criteria have been specified correctly. If so, then there really are no features with the selected characteristics in the input file.

PRMTNTCLO, prompt not closed in skeleton file, line 'integer'

Explanation: The closing single quote is missing from a prompt in line 'integer' of the skeleton file, or the prompt exceeds 24 characters.

User action: Add the closing single quote to the skeleton file and rerun ISELAC.

PRMTOOLNG, prompt too long in skeleton file, line 'integer'

Explanation: The prompt specified in line 'integer' of the skeleton file is greater than 24 characters in length.

User action: Edit the skeleton file to make the prompt less than 24 characters and rerun ISELAC.

RNGNTCLO, range not closed in skeleton file, line 'integer'

Explanation: The AC Attribute range specified in line 'integer' of the skeleton file is missing a closing square bracket "]".

User action: Edit the skeleton file to add the closing bracket and rerun ISELAC.

UNEXPEOF, unexpected end of input IFF file

Explanation: The specified input IFF file terminated before an IFF EJ entry was encountered.

User action: Use IPATCH to determine the position of the file truncation and then use IMEND to correctly terminate the file. Re-run ISELAC on the repaired IFF file.

UNRECLIN, unrecognised line 'integer' in skeleton file

Explanation: Line 'integer' of the skeleton file does not have any of the recognised records or characters at the beginning of the line. The offending line should be output after the error message.

User action: Check that the first character of the specified line is valid. Valid characters are alphabetic (for AC names), numeric (for AC types), "!" for a comment and "%" for a numeric format. Edit the skeleton file to correct the error and rerun ISELAC.

MESSAGES (OTHER)

In addition to the above messages which are generated by ISELAC itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 16

MODULE ISELECT

MODULE ISELECT

REPLACES None, ISELECT is a completely new IMP utility.

FUNCTION

ISELECT creates an IFF file to contain features selected from an existing IFF file. Features may be selected on the basis of:

- o IFF layer,
- o AC (Ancillary Code) type,
- o FSN (Feature Serial Number),
- o feature code, (the first word of an IFF FS (Feature Status) entry),
- o integer height (by default carried in a type 2 AC),
- o real height, (carried in a type 3 AC),
- o height modulus, (real or integer),
- o TCC (Text Component Code), (the first word of a TS (Text Status) entry),
- o any combination of the above.

If more than one criterion for selection is specified then all specified criteria have to match for a feature to be copied to the output file. In the event of no features matching the selection criteria ISELECT issues an error message and the empty output file is deleted.

FORMAT

\$ ISELECT input-file-spec output-file-spec

Command qualifiers

/[NO]AC[=integer,...]
/EXCLUDE_FC=integer[,...]
/FC=integer[,...]
/[NO]FSN=integer[,...]
/[NO]INTEGER_HEIGHT[=integer[,...]]
/[NO]LAYER=integer[,...]
/[NO]LOG
/[NO]MODULUS=integer[,...]
/[NO]REAL_HEIGHT[=real[,...]]
/[NO]TCC[=integer,...]

Defaults

/NOAC

/NOFSN
/NOINTEGER_HEIGHT
/NOLAYER
/NOLOG
/NOMODULUS
/NOREAL_HEIGHT
/NOTCC

PROMPTS

_From: input-file-spec
_To: output-file-spec

PARAMETERS

input-file-spec

- specifies the IFF file from which selections are to be made. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

output-file-spec

- specifies the IFF file which is to be created. Any part of the file specification which is not supplied will be taken from the parsed input file specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

/AC[=integer,...]
/NOAC

- enables selection of all features which contain an AC regardless of AC type. If qualifier arguments are supplied then only ACs of the specified type will be selected. Ranges of AC types may be specified. AC type numbers must lie in the range 1 to 32737. A maximum of 1024 AC type numbers may be specified.

/EXCLUDE_FC=integer[,...]

- excludes selection of all features which have a FC of the specified numbers. Ranges of FCs may be specified. IFF feature codes must lie in the range 0 to 32767. A maximum of 1024 feature codes may be specified.

/FC=integer[,...]

- enables selection of all features which have a FC of the specified numbers. Ranges of FCs may be specified. IFF feature codes must lie in the range 0 to 32767. A maximum of 1024 feature codes may be specified.

```
/FSN=integer[,...]  
/NOFSN
```

- enables selection of all features which have a FSN of the specified numbers. Ranges of FSNs may be specified. Feature serial numbers must lie in the range 1 to 65535. A maximum of 1024 feature serial numbers may be specified.

```
/INTEGER_HEIGHT[=int[,...]]  
/NOINTEGER_HEIGHT
```

- enables selection of all features which have an integer (type 2 AC) height of the specified values. Ranges of integer heights may be specified. A maximum of 1024 integer height values may be specified.

```
/LAYER=integer[,...]  
/NOLAYER
```

- enables selection of all features which lie in the layers with the specified values. Ranges of layers may be specified. IFF layer numbers must lie in the range 1 to 32767. A maximum of 1024 layer numbers may be specified.

```
/LOG  
/NOLOG
```

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened and a reassuring message output each time a new IFF layer is found in the IFF file.

```
/MODULUS=integer[,...]  
/NOMODULUS
```

- enables selection of all features which have a height (type 2 or type 3 AC) exactly divisible by the specified values. This qualifier is only valid with the /INTEGER_HEIGHT, /REAL_HEIGHT, or /AC qualifier to specify the AC type holding the height information. The arguments supplied for /MODULUS must be non-zero positive integer.

```
/REAL_HEIGHT[=real[,...]]  
/NOREAL_HEIGHT
```

- if specified without any arguments /REAL_HEIGHT enables selection of all features which have an real height (type 3) AC. If arguments are supplied to this qualifier then only features having a real height (in a type 3 AC) of the specified values will be selected. Ranges of real heights may be specified with starting and ending real heights that are separated by a colon, for example:

```
$ ISELECT/REAL_HEIGHT=230.7:290.8
```

When specifying more than one range of real heights, separate each range with a comma, for example:

```
$ ISELECT/REAL_HEIGHT=(240.0:240.9,895.7:899.0)
```

```
/TCC[=integer[,...]]  
/NOTCC
```

- enables selection of all features which contain a TS entry regardless of TCC (Text Component) Code. If qualifier arguments are supplied then only features that include TS entries with TCC's of the specified codes will be selected. Ranges of TCC codes may be specified. IFF TCC code values must lie in the range 0 to 32767. A maximum of 1024 TCC codes may be specified on the ISELECT command line.

RESTRICTIONS

- o /MODULUS requires /AC or /INTEGER_HEIGHT or /REAL_HEIGHT
- o /FC and /EXCLUDE_FC can not be used in combination.
- o /NOFC or /NOEXCLUDE_FC are not valid qualifiers.

DESCRIPTION

ISELECT creates a new IFF file containing features selected from an existing IFF file on the basis of attributes specified by qualifiers on the command line. Features may be selected on the basis of:

- o IFF layer,
- o AC (Ancillary Code) type,
- o FSN (Feature Serial Number),
- o feature code, (the first word of an IFF FS (Feature Status) entry),
- o integer height (by default carried in a type 2 AC),
- o real height, (carried in a type 3 AC),
- o height modulus, (real or integer),
- o TCC (Text Component Code), (the first word of a TS (Text Status) entry),
- o any combination of the above.

If more than one criterion for selection is specified then all specified criteria have to match for a feature to be copied to the output file. In the event of no features matching the selection

criteria ISELECT issues an error message and the empty output file is deleted.

An IFF HI (HIstory) entry is written to the output file only if the input IFF file contains an HI entry. Similarly, a type 2 MD entry is only written to the output file if the input file contains a type 2 MD entry.

ISELECT does **not** automatically copy the contents of layer 0 through to the output IFF file. Layer 0 is used to contain registration mark features and grids. The user must explicitly select layer 0 with the /LAYER qualifier if the contents of layer 0 is to be retained in the output file.

EXAMPLES

\$ ISELECT/FC=0/LAYER=123 W009SW W009SW_LAYER_123<CR>

ELAPSED: 00:00:17.54 CPU: 0:00:01.57 BUFIO: 16 DIRIO: 68 FAULTS: 123

\$

This is a typical run of ISELECT in a production environment. The qualifiers specify that selection is restricted to those features within layer 123 of file LSL\$IF:W009SW.IFF which have a feature code of 0. Output of the selected features is to file LSL\$IF:LSL\$IF:W009SE_LAYER_123.IFF. The run has completed successfully. \$STATUS is set to SS\$_NORMAL.

\$ ISELECT/EXCLUDEFC=0/LAYER=123 W009SW W009SW_LAYER_123<CR>

ELAPSED: 00:00:17.54 CPU: 0:00:01.57 BUFIO: 16 DIRIO: 68 FAULTS: 123

\$

This is a typical run of ISELECT in a production environment. The qualifiers specify that selection is restricted to those features within layer 123 of file LSL\$IF:W009SW.IFF which do not have a feature code of 0. Output of the selected features is to file LSL\$IF:LSL\$IF:W009SE_LAYER_123.IFF. The run has completed successfully. \$STATUS is set to SS\$_NORMAL.

\$ ISELECT/FC=(0:100)/LAYER=(1:99) LESLEY/LOG<CR>

_To: LESLEY.SEL

%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TRIALS]LESLEY.IFF;2 opened for read

%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TRIALS]LESLEY.SEL;1 opened for write

%ISELECT-I-LAYER, layer 24 found

%ISELECT-I-LAYER, layer 1 found

%ISELECT-I-LAYER, layer 66 found

%ISELECT-I-LAYER, layer 44 found

%ISELECT-I-LAYER, layer 151 found

ELAPSED: 00:00:21.20 CPU: 0:00:01.77 BUFIO: 16 DIRIO: 70 FAULTS: 95

\$

In this example the user has selected the /LOG qualifier. This has resulted in the output of supplementary messages. Because only one file was specified, LSL\$IF:LESLEY.IFF, the output file specification

was prompted for with the `_To:` prompt. The file specification `LSL$IF:LESLEY.SEL` was given in response to this. A message is output when each file is opened and when a new IFF layer is found. Only features in the input file which have a feature code in the range 0 to 100 and lie within layer 1 to 99 will be selected. The run completed successfully. `$STATUS` is set to `SS$_NORMAL`.

```
$ ISELECT/LAYER=9/INTEGER_HEIGHT/MODULUS=15<CR>
_From: BUR$DISK:[BUREAU.TRIALS.EDITING]CONTOURS7.IFF;1<CR>
_To: INDEX_CONTOURS<CR>
ELAPSED: 00:00:03.77 CPU: 0:00:00.49 BUFIO: 2 DIRIO: 7 FAULTS: 111
$
```

In this example the user has specified that only features which have an integer height (type 2) AC which is exactly divisible by fifteen and which lie within layer 9 will be selected from `BUR$DISK:[BUREAU.TRIALS.EDITING]CONTOURS7.IFF;1`. Output is to `LSL$IF:CONTOURS7.IFF;0`. The run completed successfully. `$STATUS` is set to `SS$_NORMAL`.

```
$ ISELECT/AC=(2,4)/LAYER=(1,3,11,80:200) SHEET_190.IFF;1 SELECTIONS/LOG<CR>
%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TRIALS]SHEET_190.IFF;1 opened for read
%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TRIALS]SELECTIONS.IFF;1 opened for write
%ISELECT-E-UNEXPEOF, unexpected end of IFF file
ELAPSED: 00:00:07.54 CPU: 0:00:01.81 BUFIO: 15 DIRIO: 69 FAULTS: 128
$
```

In this example only features which lie within the specified layers and which have type 2 and 4 ACs were to be selected from `LSL$IF:SHEET_190.IFF;1`. Output was to have been to `LSL$IF:SELECTIONS.IFF;1`. The run failed because the input file, `LSL$IF:SHEET_190.IFF;1` is corrupt. The output file `LSL$IF:SELECTIONS.IFF;1` will be deleted. `$STATUS` is set to `SS$_ABORT`.

The input file `LSL$IF:SHEET_190.IFF;1` may only lack an EJ entry, or it may be more seriously broken. Use `IPATCH` to determine whether the file lacks an EJ entry. If it does, use `IMEND` to try to correctly terminate the file. Re-run `ISELECT`. If the problem persists then the internal structure of the IFF file is probably corrupt. Try reading it into `LITES2` and then `EXIT` from `LITES2`. Re-run `ISELECT`.

```
$ ISELECT/AC=(2,4)/LAYER=(1,3,11,80:200)<CR>
_From: GROUND_RENT_ISOLINES/REAL_HEIGHT=(99.Y:800.0)<CR>
_To: GROUND_RENT_ISOLINES<CR>
%LSLLIB-E-INVALSPEC, invalid syntax for value specification
\9.Y\
ELAPSED: 00:00:01.65 CPU: 0:00:00.25 BUFIO: 2 DIRIO: 0 FAULTS: 100
$
```

This example shows the use of the /REAL_HEIGHT qualifier to specify selection based a range of height values within type 3 (real height value) ACs. The run failed because a 'Y' was typed instead of a numeric value as part of the range specification. \$STATUS is set to SS\$_ABORT. Re-enter the command line.

```
$ ISELECT/AC=(2,4)/LAYER=(1,3,11,80:200) FE/MODULUS=10<CR>
_To: FESEL<CR>
%LSLLIB-E-IFFOPEN, error opening input file "LSL$IF:FE.IFF;0" for read
-RMS-F-FNF, file not found
ELAPSED: 00:00:03.14 CPU: 0:00:00.37 BUFIO: 4 DIRIO: 2 FAULTS: 81
$
```

In this example the input file does not exist or is not in the directory to which LSL\$IF: has been assigned. \$STATUS is set to SS\$_ABORT. Check the file specification for errors, check that logical name LSL\$IF: has been assigned to the correct device and directory.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). Generally they will only appear if /LOG has been specified on the command line.

DELETE, deleting empty output file 'file-spec'

Explanation: ISELECT is to delete the specified output IFF file because none of the selections made from the input file matched. The output IFF file is therefore empty and useless.

User action: Determine why none of the selections matched data in the input IFF file. Re-run ISELECT with new selections or use a different input IFF file.

LAYER, layer 'integer' found

Explanation: The specified layer has been found in the input file and is being processed.

User action: None.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. With the exception of the IFF file closure errors, no output file will be produced if one of these error messages is output.

ERRCLINFL, error closing input IFF file 'file-spec'

Explanation: The specified file cannot be closed. This message will be accompanied by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. If the problem continues, notify the system manager.

ERRCLOUTFL, error closing output IFF file 'file-spec'

Explanation: The specified file cannot be closed. This message will be accompanied by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. If the problem continues, notify the system manager.

IHNOTTWO, /INTEGER_HEIGHT selected without /AC=2

Explanation: You have specified the /INTEGER_HEIGHT command qualifier and /AC=[n...]. Unfortunately you have excluded type 2 ACs from the argument list of /AC resulting in conflicting requirements from the two qualifiers. No matches for /INTEGER_HEIGHT will be found, although there may be features containing type 2 integer height ACs in the input file.

User action: Re-enter the command line, ensuring that the /AC qualifier contains type 2 ACs in its argument list. Alternatively, if you really only require those features which have type 2 height Acs omit the /AC qualifier altogether.

INVALFC, invalid /AC argument - "'integer'" is outside range 0 to 32767

Explanation: IFF AC type numbers lie in the range 0 to 32767.

User action: Respecify the ISELECT command line and ensure that the arguments for the /AC qualifier lies within the specified range.

INVALFC, invalid /FC argument - "'integer'" is outside range 0 to 32767

Explanation: IFF feature codes lie in the range 0 to 32767.

User action: Respecify the ISELECT command line and ensure that the arguments for the /FC qualifier lies within the specified range.

INVALFSN, invalid /FSN argument - "'integer'" is outside range 0 to 32767

Explanation: IFF Feature serial numbers lie in the range 0 to 65535.

User action: Respecify the ISELECT command line and ensure that the arguments for the /FSN qualifier lies within the specified range.

INVALLAY, invalid /LAYER argument - "'integer'" is outside range 0 to 32767

Explanation: IFF layer numbers lie in the range 0 to 32767.

User action: Respecify the ISELECT command line and ensure that the arguments for the /LAYER qualifier lies within the specified range.

INVALMOD, invalid /MODULUS argument - must be integer greater than 0

Explanation: IFF layer numbers lie in the range 0 to 32767.

User action: Respecify the ISELECT command line and ensure that the arguments for the /LAYER qualifier lies within the specified range.

NOHTACMOD, height ACs (AC 2 and 3) deselected for /MODULUS

Explanation: You have selected the /MODULUS qualifier but have excluded all type 2 and 3 height Acs by use of the /AC qualifier.

User action: Re-enter the command line, ensuring that the /AC qualifier contains type 2 and type 3 ACs in its argument list. Alternatively, if you really only require those features which have type 2 or 3 height Acs omit the /AC qualifier altogether.

RHNOTTHREE, /REAL_HEIGHT selected without /AC=3

Explanation: You have specified the /REAL_HEIGHT command qualifier and /AC=[n...]. Unfortunately you have excluded type 3 ACs from the argument list of /AC resulting in conflicting requirements from the two qualifiers. No matches for /REAL_HEIGHT will be found, although there may be features containing type 3 real height ACs in the input file.

User action: Re-enter the command line, ensuring that the /AC qualifier contains type 3 ACs in its argument list. Alternatively, if you really only require those features which have type 3 height Acs omit the /AC qualifier altogether.

UNEXPEOF, unexpected end of IFF file

Explanation: The input IFF file terminated before an IFF EJ entry was encountered.

User action: Use IPATCH to determine whether the file has an EJ entry. If the file lacks an EJ entry use IMEND to correctly terminate the file. Try re-running the IMP utility which failed. If the problem persists then the internal structure of the IFF file is probably corrupt. Try reading it into LITES2 and then EXIT from LITES2. Re-run the IMP utility.

MESSAGES (OTHER)

In addition to the above messages which are generated by ISELECT itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 17

MODULE ISORT

MODULE ISORT

REPLACES DAMP modules SIF and IRN

FUNCTION

ISORT sorts an IFF file on the basis of FSN (Feature Serial Number), ISN (Internal Sequence Number), or FC (Feature Code).

FORMAT

\$ ISORT file-spec file-spec

Command qualifiers

/DUPLICATES
/NODUPLICATES=FIRST

/NODUPLICATES=LAST

/FC

/INTERNAL

/[NO]LOG

/OUTPUT[=file-spec]

/[NO]PRINTER

Defaults

Retain all duplicate FSNs
Retain only the first duplicate FSN (See text).
Retain only the last duplicate FSN (See text).

Sort on the basis of FSN.

/NOLOG

/OUTPUT=SYS\$OUTPUT

/NOPRINTER

PROMPTS

Unsorted-file: unsorted-file-spec
Sorted-file: sorted-file-spec

PARAMETERS

unsorted-file-spec

- specifies the IFF file to be sorted. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF;0'.

sorted-file-spec

- specifies the IFF file which is to be created. Any part of the file name which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS**/DUPLICATES**

- forces all features with duplicate FSNs to be retained in the sorted file. The features are retained in the same order as in the unsorted file. This is the default action for ISORT. The **/NODUPLICATES** qualifier may be used to override this default, and discard features with duplicate FSNs.

/NODUPLICATES=FIRST

- forces all features with duplicate FSNs, with the exception of the **FIRST** duplicate feature, to be discarded from the sorted file.

/NODUPLICATES=LAST

- forces all features with duplicate FSNs, with the exception of the **LAST** duplicate feature, to be discarded from the sorted file.

/FC

- forces sorting within layers to be based on the FC (Feature Code) rather than the FSN. Within each layer, features with the same code, which will usually require plotting in the same colour, will be grouped together. Plotting with some colour plotters will thus be speeded up by minimising the number of pen changes required.

/INTERNAL

- forces sorting to be based on the ISN (Internal Sequence Number) rather than the FSN. This will usually result in the restoration of the original digitising order.

/LOG**/NOLOG**

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when a file is successfully opened, the contents of the HI entry are then output to the listing file if **/OUTPUT** is specified and a reassuring message output each time a new layer is encountered in the IFF file.

/OUTPUT[=file-spec]

- by default output will be to SYS\$OUTPUT. This option allows the user to redirect output to file. If **/OUTPUT** is specified ISORT can generate voluminous output.

/PRINTER**/NOPRINTER**

- queues the ISORT output for printing under the name given by the **/OUTPUT** qualifier. After printing the file is deleted. If **/PRINTER** is specified without the **/OUTPUT** qualifier, the output is directed to

a file named SYS\$DISK:ISORT.LIS which is queued automatically for printing and then deleted.

DESCRIPTION

ISORT is an IFF file sort program. Its primary purpose is to sort the features in an IFF file into ascending Feature Serial Number (FSN) order prior to output from IFF to a customer format, as the order will usually have been disturbed by editing with LITES2.

In order to preserve the layer structure found in the input IFF file sorting is conducted on a layer by layer basis. Features with a FSN of 0 will be included in the sort.

ISORT outputs a summary of feature serial number allocation including duplicated feature serial numbers and gaps. The FSNs will normally be unique within any one layer. If duplicated FSNs are present then the /NODUPLICATES qualifier with either the FIRST or LAST option, may be used to override the default mechanism of retaining all features with duplicate FSNs. When /NODUPLICATES=FIRST is selected only the first feature with a duplicated FSN in a given layer is kept. When /NODUPLICATES=LAST is selected only the last feature with a duplicated FSN is retained.

The /INTERNAL qualifier may be specified in order that sorting be based on the features' ISN (Internal Sequence Number) instead of the FSN. Each feature in a file should have a unique ISN, and therefore the use of the /NODUPLICATES qualifier is not allowed if a sort on ISN is specified.

The /FC qualifier may be specified so that sorting is based on the features' FC (Feature Code). All the features with the same code within a layer will be grouped together and hence will be plotted consecutively. Since features with the same feature code are usually plotted in the same colour, the action of some colour pen plotters will be speeded up by reducing the number of pen changes needed. The /NODUPLICATES qualifier is not allowed with a sort on FC, since many features may be expected to have the same code.

EXAMPLES

\$ ISORT LESLEY FE.IFF<CR>

ELAPSED: 00:02:03.15 CPU: 0:00:56.12 BUFIO: 8362 DIRIO: 3230 FAULTS: 1379
\$

This is a typical run of ISORT. No qualifiers have been specified and sorting by FSN has occurred by default. Minimal output is sent to SYS\$OUTPUT: as the /LOG qualifier has not been given and no text output has been sent to a list file or to the printer.

\$ ISORT LESLEY FE.IFF/LOG

%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.TEST]LESLEY.IFF;0 opened for read

%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.TEST]FE.IFF opened for write

%ISORT-I-LAYER, layer 4 found

%ISORT-I-LAYER, layer 7 found

%ISORT-I-LAYER, layer 1 found

%ISORT-I-LAYER, layer 24 found

%ISORT-I-LAYER, layer 31 found

%ISORT-I-LAYER, layer 32 found

ELAPSED: 00:00:11.18 CPU: 0:00:02.38 BUFIO: 7 DIRIO: 50 FAULTS: 1227

\$

As in the previous example sorting has been specified on FSN by default. This time /LOG was specified so messages are output when the input and output files are opened and each time a new IFF layer is found in the input IFF file.

\$ ISORT LESLEY/OUTPUT/NODUPLICATES=FIRST FE.IFF<CR>

===== ISORT =====

Input file: DUA0:[IFF.IMP.TEST]LESLEY.IFF;3

Output file: DUA0:[IFF.IMP.TEST]FE.IFF;4

Command line:

ISORT LESLEY/OUTPUT/NODUPLICATES=FIRST FE.IFF

SORT SUMMARY:

Layer 1

Duplicate Feature with FSN 1 discarded

Duplicate Feature with FSN 1 discarded

Duplicate Feature with FSN 98 discarded

Lowest Feature Serial Number 1

Missing Feature Serial Numbers 113 to 327

Highest Feature Serial Number 329

Layer 1 contains 116 sorted features

3 duplicates were omitted

Layer 4

Duplicate Feature with FSN 6 discarded

Duplicate Feature with FSN 74 discarded

Lowest Feature Serial Number 408

Missing Feature Serial Numbers 2117 to 9995

Highest Feature Serial Number 9999

Layer 4 contains 1713 sorted features

2 duplicates were omitted

DUA0:[IFF.IMP.TEST]LESLEY.IFF;3 contains 2028 unsorted features

DUA0:[IFF.IMP.TEST]FE.IFF;4 contains 2023 sorted features, 5 duplicates were omitted

ELAPSED: 0 00:00:19.15 CPU: 0:00:10.28 BUFIO: 26 DIRIO: 242 FAULTS: 1192

\$

In this example, /NODUPLICATES=FIRST is used so that all features with duplicate FSNs with the exception of the first feature, are discarded. A message is generated for each feature discarded. The /OUTPUT qualifier without a file specification causes supplementary messages to be sent to SYS\$OUTPUT:.

\$ ISORT/FC/LOG/OUT START FCSORTED<CR>

```
%LSLLIB-I-IFFOPENED, LSL$SOURCEROOT:[IMP.ISORT]START.IFF;2 opened for read
%LSLLIB-I-IFFOPENED, LSL$SOURCEROOT:[IMP.ISORT]FCSORTED.IFF;5 opened for write
===== ISORT =====
Input file:  LSL$SOURCEROOT:[IMP.ISORT]START.IFF;2
Output file: LSL$SOURCEROOT:[IMP.ISORT]FCSORTED.IFF;5
Command line:
ISORT/FC/LOG/OUT START FCSORTED
SORT SUMMARY:
%ISORT-I-LAYER, layer 1 found
Layer 1
Lowest Feature Code          1
Highest Feature Code         300
Layer 1 contains 16 sorted features
%ISORT-I-LAYER, layer 2 found
Layer 2
Lowest Feature Code          1
Highest Feature Code        1000
Layer 2 contains 8 sorted features
%ISORT-I-LAYER, layer 3 found
Layer 3 is empty
%ISORT-I-LAYER, layer 4 found
Layer 4
Lowest Feature Code          1
Highest Feature Code          1
Layer 4 contains 1 sorted features
LSL$SOURCEROOT:[IMP.ISORT]START.IFF;2 contains 25 unsorted features
LSL$SOURCEROOT:[IMP.ISORT]FCSORTED.IFF;5 contains 25 sorted features
ELAPSED:    0 00:00:14.66  CPU: 0:00:12.61  BUFIO: 49  DIRIO: 73  FAULTS: 1598
$
```

In this example, the /FC qualifier specified that sorting within layers was to be on feature code. Because the /LOG and /OUT options were also used, ISORT indicates the layers encountered, the number of features which they contain, and the range of feature codes present in each.

\$ ISORT UNCODE CODE<CR>

```
%LSLLIB-E-IFFOPEN, IFF error opening file "LSL$IF:UNCODE.IFF;0" for read
%IFF-E-OPEN, error opening IFF file
-RMS-E-FNF, file not found
ELAPSED: 00:00:01.11  CPU: 0:00:00.28  BUFIO: 2  DIRIO: 4  FAULTS: 60
$
```

In this example the input file does not exist or is not in the directory to which LSL\$IF: has been assigned. Check the file specification for errors and check that logical name LSL\$IF: has been assigned to the correct device and directory.

\$ ISORT LESLEY FE.IFF/DUP/OUT=LESLEY.LIS<CR>

```
ELAPSED: 00:00:14.80  CPU: 0:00:05.65  BUFIO: 10  DIRIO: 60  FAULTS: 1988
$
```

\$ ISORT LESLEY FE.IFF/OUT=LESLEY.LIS/PRINTER<CR>

```
ELAPSED: 00:00:02.54  CPU: 0:00:00.38  BUFIO: 3  DIRIO: 1  FAULTS: 155
$
```

\$ ISORT LESLEY FE.IFF/PRINTER/INTERNAL<CR>

```
ELAPSED: 00:00:07.78  CPU: 0:00:01.78  BUFIO: 6  DIRIO: 55  FAULTS: 1125
$
```

These examples show the various possibilities for specifying /PRINT and /OUTPUT. In each case the the input file is DUA0:[IFF.IMP.TEST]LESLEY.IFF and the output file is DUA0:[IFF.IMP.TEST]FE.IFF. In the first example an output file SYS\$DISK:LESLEY.LIS is created with program output directed to it with the /OUTPUT qualifier. The second example also creates an output file SYS\$DISK:LESLEY.LIS with the /OUTPUT qualifier but as the /PRINT qualifier is also given the file is printed and then deleted. In the final example the /PRINT qualifier is given without any filename specified by the /OUTPUT qualifier so a temporary file SYS\$DISK:ISORT.LIS is created with output information, printed then deleted.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user. They are often used to supply explanatory information in support of a warning or error message.

LSTOPN, listing file 'file-spec' opened

Explanation: The specified listing file has been opened successfully.

User action: None.

LAYER, layer 'integer'

Explanation: The specified Layer has been encountered in the input file and is being sorted.

User action: None.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

LSTNSPLDEL, listing file not spooled or deleted.

Explanation: The program has been unable to queue for printing or delete the listing file specified by the /OUTPUT or /PRINT qualifier.

User action: Check that any listing file specification is correct. Check that the print queues are functioning correctly. If the problem continues notify the system manager.

LSTNSPL, listing file not spooled.

Explanation: The program has been unable to place the listing file specified by the /OUTPUT or /PRINT qualifier onto the printer queue.

User action: Check that any listing file specification is correct. Check that the print queues are functioning correctly. If the problem continues notify the system manager.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ERRCLINFL, error closing input IFF file
ERRCLOUTFL, error closing output IFF file

Explanation: An error has occurred in closing the specified IFF file.

User action: This should not normally occur. Check that there is sufficient disk space for the file and that the disk is functioning properly. If the problem continues, notify the system manager.

ERROPLST, error opening list file 'file-spec'

Explanation: The supplied file specification is erroneous, the device or directory specified does not exist, or you do not have write access to the directory.

User action: The accompanying messages will help you to interpret the cause of the error. Check the file specification for errors, check that you have read access to the device and directory. If the problem continues, notify the system manager.

NOFEATURES, input file contains no features - deleting output file

Explanation: The input IFF file does not contain any features. Since ISORT sorts IFF features there is no point in retaining any IFF output file produced. An IFF output file will have been created by the time ISORT detects that there are no features. ISORT closes and deletes this output file before terminating execution. \$STATUS will be set to SS\$ABORT.

User action: Use IINFO on the file to determine whether it contains an HI (HISTORY) entry. This should reveal what has caused the file to contain no features. Without features an IFF file is useless unless it is destined as template file to receive new digitising.

NOLAYERS, input file contains no layers - deleting output file

Explanation: The input IFF file does not contain any layers. ISORT sorts on a layer by layer basis and cannot work outside of a layer structure. An IFF output file will have been created by the time ISORT detects that there are no layers. ISORT closes and deletes this output file before terminating execution. \$STATUS will be set to SS\$ABORT.

User action: Use IINFO on the file to determine whether it contains an HI (HISTORY) entry. This should reveal what has caused the file to contain no layers. Many IFF utilities will behave unpredictably when presented with such a file. It is probably best to use ITOTEXT followed by IFROMTEXT to create a default layer structure in the file.

UNEXPEOF, unexpected end of IFF file

Explanation: This message indicates there is something seriously wrong with the IFF file which has caused immediate termination of the program. ISORT has detected the end of the IFF file, but has not detected an IFF 'EJ' entry.

User action: Use IMEND on the file, which will correctly position the EOF marker and insert an EJ entry at the end of the file. Re-run ISORT on the corrected file.

MESSAGES (OTHER)

In addition to the above messages which are generated by ISORT itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 18

MODULE ISTART

MODULE **ISTART**

REPLACES Customer specific utilities. ISTART is a general purpose utility.

FUNCTION

ISTART is designed to create an IFF file as a pre-digitising template. The file is then filled with data captured during the digitising process. ISTART enables the user to specify the control points and origin offset information which are written to the template CP (Control Point) and MD (Map Descriptor) entries respectively.

ISTART offers three modes of operation:

1. Template mode, where prompts are issued for the user to type the control points,
2. Ordnance Survey mode, where prompts are issued for Ordnance Survey (UK) specific information,
3. digitising table mode, where the control point values are collected from:
 - o a Laser-Scan MUART (Multi-UART), or,
 - o a Laser-Scan TABLE_MONITOR driven digitising table, or,
 - o via ISTART direct table reads.

FORMAT

\$ ISTART file-spec

Command qualifiers

/[NO]CHECK

/FC=integer

/[NO]LASERAID

/LAYER=integer

/[NO]LEARNER

/[NO]LOG

/MCE

/MH_TYPE=integer

/MUART_TABLE

/OS

/[NO]SYMBOLS

/TABLE

/[NO]TICKS

/USER_ORIGIN

Digitising table qualifiers

/ADDRESSABILITY=real

Defaults

Control points are checked.

Feature code 0 used for registration ticks or crosses.

See text.

Empty layer 1 created.

Minimal prompting.

/NOLOG

Standard Laser-Scan file.

/MH_TYPE=2

Template mode.

Standard Laser-Scan file.

Registration ticks.

Template mode.

Registration ticks.

Origin offset automatically derived.

Defaults

See text.

```
/BUTTONS=(NUMBER:integer,BASE:integer)      16 button cursor, base value 1
/CENTRE                                     Map origin at bottom left corner.
/COUNTS                                     Digitiser output in millimetres.
/REPEAT=integer                             Points to be digitised 3 times.
/[NO]ROTATION                               /ROTATION
/THOU                                       Digitiser output in millimetres.
```

PROMPT

```
_IFF file for output:      output-file-spec
```

PARAMETER

output-file-spec

- specifies the IFF file to be created. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

COMMAND QUALIFIERS

```
/CHECK
/NOCHECK
```

- by default ISTART checks the control point values supplied by the user. These checks are to ensure that the control points are cyclic (ie no "bow tie" deformation of digitising space is implied) and that the values supplied form a viable set of controls for Laseraid digitising. /NOCHECK may be used to switch off ISTART control point checking.

```
/FC=integer
```

- by default registration marks (see /SYMBOLS) will be given a feature code of 0. The /FC qualifier enables the user to specify an alternative feature code to be used for registration mark features. The graphical code of the two types of registration marks is, however, fixed. Registration crosses will always have word 3 of the feature status entry set to indicate a symbol. Registration ticks will always have word 3 of the feature status entry set to indicate a line. The FRT (Feature Representation Table) file which will be used later in conjunction with the output file should reflect these characteristics. The argument supplied for /FC must be in the range 0 to 32767.

```
/LASERAID
/NOLASERAID
```

- by default ISTART creates an IFF output file which is unsuitable for use within the Laser-Scan Laseraid digitising system. The /LASERAID qualifier causes ISTART to generate an IFF file which is compatible with Laseraid. For a detailed description of the effects of the /LASERAID qualifier see the Description Section.

/LAYER=integer

- by default ISTART creates an empty layer 1 in the output file. The /LAYER qualifier enables the user to specify the layer number which is to be created in the output file. Layers must be in the range 1 to 32767. Layer 0 will always be present in the file and contains the registration marks (see /SYMBOLS and /TICKS) and the grid (if in Ordnance Survey mode).

/LEARNER

/NOLEARNER

- by default ISTART issues only minimal prompts for input from the user. The /LEARNER qualifier causes longer, explanatory, prompts to be issued and its use is recommended for the inexperienced user, especially if digitising table mode is invoked.

/LOG

/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. A supplementary message is generated when the output file is successfully opened and the specified control points and origin offset values are displayed before program exit.

/MCE

- by default ISTART creates a standard type 2 MH (Map Header) entry in the output IFF file. Laser-Scan have for many years provided some customers with special map header entries to suit their particular mapping requirements. Although this practice has now been discontinued ISTART provides the mechanism to generate a customer specific Map Header entry for reasons of compatibility with existing customer specific software and file archives. The /MCE qualifier facilitates the creation of an MCE (Mapping and Charting Establishment R.E.) format MH entry.

/MH_TYPE=integer

- this is valid only when used in conjunction with the /OS qualifier and is used to specify the type of the map header to be created in the output file. The integer argument to /MH_TYPE can be either 2, 3 or 4, and the default is /MH_TYPE=2. Laser-Scan recommend that type 3 and 4 headers are only used by Ordnance Survey (Great Britain) and that other users of OS data use type 2 map headers which can be edited using the CONVERT utility OSMHED.

/MUART_TABLE

- ISTART enters template mode by default. In this mode no input from a digitising table is required. In template mode the user types in the control point values in response to prompts. If the /MUART_TABLE qualifier is specified ISTART enters digitising table input mode, in which the control point values are collected from a Laser-Scan MUART (Multi-UART) driven digitising table. The user is prompted to digitise each control point position the number of times specified

with the /REPEAT qualifier (default 3) and the results are written into the CP (Control Point) entry of the output file.

/OS

- like the /MCE qualifier (q.v.) this qualifier is provided to maintain compatibility with an existing customer specific MH (Map Header) entry format. The /OS qualifier facilitates the creation of an O.S. (Ordnance Survey, U.K.) format MH entry. The map scale, sheet origin, capture scale and O.S. "basic grid interval" (BGI) are written into the MH entry.

/TABLE

- ISTART enters template mode by default. In this mode no input from a digitising table is required. In template mode the user types in the control point values in response to prompts. If the /TABLE qualifier is specified ISTART enters digitising table input mode, in which the control point values are collected from a Laser-Scan TABLE_MONITOR driven digitising table. If the TABLE_MONITOR process is not running, or TABLE_MONITOR is not available at your site, ISTART will attempt to read coordinate positions from the digitising table using direct table reads. In either case the user is prompted to digitise each control point position the number of times specified with the /REPEAT qualifier (default 3) and the results are written into the CP (Control Point) entry of the output file.

/SYMBOLS

/NOSYMBOLS (default)

- by default ISTART generates registration marks in layer 0 as "L" shaped ticks, each defined by three points.

The /SYMBOLS qualifier forces output of single point registration symbols. By default registration marks of either sort will be given a feature code of 0. The /FC qualifier (q.v.) enables the user to specify an alternative feature code to be used for registration mark features. Registration symbol features will always have word 3 of the feature status entry set to indicate a symbol. Registration ticks will always have word 3 of the feature status entry set to indicate a line. The FRT (Feature Representation Table) file which will be used later in conjunction with the output file should reflect these characteristics.

If both /NOSYMBOLS and /NOTICKS are specified, an empty layer zero is generated, into which the user may subsequently digitise his own registration/graticule information.

/TICKS (default)

/NOTICKS

- by default ISTART generates registration marks in layer 0 as "L" shaped ticks, each defined by three points.

By default registration ticks will be given a feature code of 0. The /FC qualifier (q.v.) enables the user to specify an alternative feature code to be used for registration mark features. Registration ticks will always have word 3 of the feature status entry set to indicate a line. The FRT (Feature Representation Table) file which will be used later in conjunction with the output file should reflect these characteristics.

If both /NOSYMBOLS and /NOTICKS are specified, an empty layer zero is generated, into which the user may subsequently digitise his own registration/graticule information.

/USER_ORIGIN

- in template mode (the default) ISTART prompts the user for control point values. When all four control point coordinates have been supplied, ISTART automatically removes an offset from the coordinates such that the bottom left control point becomes 0.0, 0.0. The extracted offset is placed in the real*8 (double precision) origin offset field of the type 2 MD (Map Descriptor) entry in the output IFF file. The /USER_ORIGIN qualifier enables the user to override the automatic offset calculation and instead the user is prompted for an appropriate origin offset coordinate. The supplied value is then placed in the real*8 (double precision) origin offset field of the type 2 MD (Map Descriptor) entry as in the automatic mode of operation.

/USER_ORIGIN is not valid for use with the /OS qualifier. O.S. IFF files contain an O.S. specific origin offset within their MH (Map Header) entry. This is set by ISTART when in O.S. mode.

Digitising table qualifiers

/ADDRESSABILITY=real

- ISTART assumes that the digitising table has an addressability of 0.02mm (Altek Datatab). That is to say that there are 50 table counts to the millimetre. Should the digitising table addressability differ from this figure then the /ADDRESSABILITY qualifier may be used to specify the addressability of the digitising table. This is clearly a qualifier that would have to be specified every time that ISTART is invoked, if the digitising table does not have the default 0.02mm addressability. It is therefore recommended that the user define the DCL symbol used to invoke ISTART in the following manner:

```
$ ISTART:==$LSL$EXE:ISTART
$ ISTART:=='ISTART'/ADDRESSABILITY=real
```

where 'real' is the desired table addressability in mm. This is best done in the IMPINI command file supplied as a component of the IMP sales package. If you have any doubt about defining this symbol please contact Laser-Scan.

/BUTTONS=(NUMBER:integer,BASE:integer)

- by default ISTART assumes that a 16 button cursor is available and that the buttons are numbered 0 to 15. With the NUMBER keyword argument the /BUTTONS qualifier enables the user to specify the number of buttons available on the cursor supplied with the digitising table. The BASE keyword argument enables the user to specify a base offset to be added to the button value returned to ISTART by the digitising table controller for one to one correspondence between the button position on the puck and its value. An Altek DATATAB (C) 16 button cursor, for example, returns button press values of 0 to 15! The BASE qualifier should thus be set to 1 for button number/value returned correspondence. It is important that the NUMBER and BASE keyword values are correctly set as these control which puck buttons are identified as the "abandon point" and "abandon session" buttons.

/BUTTONS is clearly a qualifier that would have to be specified every time that ISTART is invoked, if a 16 button cursor is not available. It is therefore recommended that the user define the DCL symbol used to invoke ISTART in the following manner:

```
$ ISTART:==$LSL$EXE:ISTART
$ ISTART:=='ISTART'/BUTTONS=(NUMBER:integer,BASE:integer)
```

where the appropriate values are substituted for 'integer'. This is best done in the IMPINI command file supplied as a component of the IMP sales package. If you have any doubt about defining this symbol please contact Laser-Scan.

/CENTRE

- by default ISTART removes an offset from the digitiser counts returned by the digitising table options to make the bottom left control point (0.0,0.0). The other 3 control points are then calculated as offsets relative to this false, but sheet relevant, Cartesian origin. The /CENTRE qualifier forces ISTART to calculate the centre of gravity of the four control point digitiser count values and to offset the control point values relative to this.

/COUNTS

- by default ISTART converts the digitiser table counts returned by the digitising table options into millimetres. The /COUNTS qualifier causes ISTART to output the raw digitiser counts to the output IFF file, as absolute table positions. /COUNTS is invalid for use with the /ROTATION and /CENTRE qualifiers.

/REPEAT=integer

- if one of the digitising table options is selected ISTART prompts the user to provide puck button presses at each control point position on the digitising table. By default the user will be prompted to digitise each control point 3 times and the mean coordinate position for that point is calculated from the 3 button presses. The /REPEAT qualifier enables the user to specify the number of times that a control point must be digitised. The mean coordinate position will

then be calculated on the basis of a larger (or smaller!) sample of puck button presses. The /REPEAT argument should be an integer in the range 1 to 16.

/ROTATION

/NOROTATION

- by default, after digitising, ISTART corrects the control point coordinates for any rotation error caused by taping the source document onto the digitising table out of alignment with the Cartesian axes of the table coordinate system. The /NOROTATION qualifier overrides this default rotation correction and the control points are left with any rotation errors relative to the Cartesian axes of the digitising table coordinate system.

/THOU

- by default ISTART converts the digitiser table counts returned by the digitising table into millimetres. The /THOU qualifier forces ISTART to convert the digitiser table counts into thousandths of an inch.

RESTRICTIONS

- o /LASERAID and /FC are mutually exclusive
- o /OS and /CHECKS are mutually exclusive
- o /OS and /LAYER are mutually exclusive
- o /OS and /MCE are mutually exclusive
- o /OS and /MUART_TABLE are mutually exclusive
- o /OS and /TABLE are mutually exclusive
- o /COUNTS and /CENTRE are mutually exclusive
- o /COUNTS and /ROTATION are mutually exclusive
- o /COUNTS and /THOU are mutually exclusive
- o /COUNTS and /USER_ORIGIN are mutually exclusive
- o /TABLE and /MUART_TABLE are mutually exclusive
- o /CENTRE requires either /MUART_TABLE or /TABLE
- o /COUNTS requires either /MUART_TABLE or /TABLE
- o /MH_TYPE requires /OS
- o /REPEAT requires either /MUART_TABLE or /TABLE

- o /THOU requires either /MUART_TABLE or /TABLE

DESCRIPTION

General

ISTART is designed to create an IFF file as a pre-digitising template. The file is then filled with data captured during the digitising process. IFF files generated by ISTART contain HI (History) entries and type 2 MD (Map Descriptor) entries. They are therefore unsuitable for use with some customer specific utilities which are unable to handle these new entry types. Please refer any such problems to Laser-Scan.

ISTART enables the user to specify the control points and origin offset information which are written to the template CP (Control Point) and MD (Map Descriptor) entries respectively. The values for the control points may either be typed at the user's terminal, or they may be coordinates captured from a manual digitising table. The LASERAID digitiser may then be started up in 'OLD' mode reducing the time spent by the user in subdued lighting conditions typing in control point values. Except in Ordnance Survey mode (see below), entered by specifying the /OS qualifier, or when taking input from a digitising table, ISTART will automatically remove an appropriate origin offset from the control point values supplied by the user. The X and Y origin offset values removed from the control points are stored in the origin offset field of the type 2 MD (Map Descriptor) as two double precision (real*8) numbers. This mechanism may be over-ridden and the user prompted explicitly for the origin offset by specifying the /USER_ORIGIN qualifier.

ISTART offers three modes of operation:

- (i) Template mode, where prompts are issued for the user to type the control points,
- (ii) Ordnance Survey (U.K.) mode, where prompts are issued for Ordnance Survey specific information,
- (iii) digitising table mode, where the control point values are collected from:
 - 1. a Laser-Scan MUART (Multi-UART), or,
 - 2. a Laser-Scan TABLE_MONITOR driven digitising table, or,
 - 3. via ISTART direct table reads.

By default ISTART enters the template mode of operation.

Template mode

In template mode the user is prompted for the following input from the keyboard:

- (i) the map scale,
- (ii) the control point values,
- (iii) the origin offset, (if the /USER_ORIGIN origin is specified).

Unless /USER_ORIGIN is specified the default action is to make the bottom left control point (0.0,0.0) and the origin offset is set to the former value of the bottom left control point.

Ordnance Survey mode

Ordnance Survey mode is entered with the /OS qualifier. In Ordnance Survey mode the user is prompted for the following input from the keyboard:

- (i) the map scale,
- (ii) the map extent,
- (iii) the grid step,
- (iv) the sheet origin offset.

In Ordnance Survey mode the control point values are calculated from the extent value(s) supplied. The grid step is used to calculate the coordinates of (axis extent/grid step)+1 grid lines for each axis. The grid lines are output as IFF features each having two coordinates in layer 0 and a feature code of 398. The bottom left control point is given the value (0.0,0.0).

In addition to the grid, O.S. format IFF files also receive registration marks to represent the locations of the control points. Output of registration marks is suppressed if the /LASERAID qualifier is specified.

ISTART creates customer specific MH (Map Header) entries if the appropriate customer command qualifiers are specified. In Ordnance Survey mode ISTART creates an Ordnance Survey format MH entry. The map scale, sheet origin and "basic grid interval" (BGI) are written into the MH entry. Any additional O.S. Map Header information must be generated using the Laser-Scan CONVERT utility, OSMHED, if required.

If the /MH_TYPE qualifier was used to create a type 3 header, the start sheet flag (-1), header flag (-20), Eastings, Northings, scale, capture scale and basic grid interval are inserted into the header using the positions and sizes specified in the translation table,

LSL\$OS_MH_TABLE. For more information on type 3 headers and on setting up the translation table, refer to the DATA PREPARATION section in the IFFOSTF chapter of the "Convert User Guide".

If the /MH_TYPE qualifier was used to create a type 4 header, a default CITF Section Header Record (SECHREC) is written to the header with the Easting, Northing and Scale written at the positions defined in the lookup table, LSL\$OS_MH_TABLE. The size of the header created will be taken from the lookup table and this must lie in the range 320 to 10000 bytes. The default header lines will be written thus:

```

0          10          20          30          40          50          60          70
123456789012345678901234567890123456789012345678901234567890123456789
07                                     X_ORIG      Y_ORIG                      1%
00                                     1%
00                                SCALE

```

Where the '07' denotes the start of the Section Header Record, the '00's are continuation records and each line is terminated by a continuation flag (1) and and End of Record character (%).

Warning

Type 3 and 4 map headers are not intended to be used outside OS, and Laser-Scan recommend that other users of OS data use type 2 map headers which can be edited using the CONVERT utility OSMHED.

ISTART produces only a limited number of fields can be inserted into the MH entry. It is intended that the header is manually edited with a suitable OS developed header editor to insert other fields such as entry flags and the job number before using CONVERT utilities such as OSPIF.

In O.S. mode ISTART sets the projection, units and spheroid fields of the MD (Map Descriptor) as follows:

```

Projection (101) - Transverse Mercator (U.K. National Grid)
Spheroid    (9)  - Airy
Units       (2)  - metres

```

Digitising table mode

Laser-Scan MUART (Multi-UART) and TABLE_MONITOR driven digitising tables may be entered with the /MUART_TABLE and /TABLE qualifiers respectively. If the TABLE_MONITOR process is not running or if TABLE_MONITOR is not available at your site then the /TABLE qualifier will cause ISTART to attempt to read coordinate positions from the digitising table using direct table reads. Whether controlled by MUART, TABLE_MONITOR or ISTART istelf digitising table mode allows the capture of arbitrary fiducial positions for which coordinate values are not available on the source document. The digitiser counts

returned by the digitising table mode are manipulated in various ways depending upon the selection of the following options:

- (i) correct raw digitiser counts for map rotation (/ROTATION) relative to the Cartesian axes of the table coordinate system. This is the default ISTART action.
- (ii) leave the digitiser output in raw digitiser counts (/COUNTS). (ISTART does not support the /CENTRE and /ROTATION qualifiers for use with /COUNTS).
- (iii) force conversion of digitiser counts to millimetres. Millimetres are the default ISTART digitiser output measurement units.
- (iv) force conversion of digitiser counts to thousandths of an inch (/THOU). Millimetres are the default ISTART digitiser output measurement units.
- (v) convert output coordinates to be relative to the centre of gravity (/CENTRE) of the map sheet (based on the four control point co-ordinate values). The default action is to adjust the co-ordinates relative to a (0.0, 0.0) origin at the position of the bottom left control point on the table.

Unless /USER_ORIGIN is specified the default action is to make the bottom left control point (0.0,0.0) and the origin offset is set to (0.0,0.0). If /USER_ORIGIN is specified, remember to specify the origin offset in the same measurement units as the possibly converted digitiser counts (default mm).

If the origin of the digitised control points is moved to the centre of gravity of the control points it is important that the user remember this if specifying an origin offset (with /USER_ORIGIN).

ISTART imposes an arbitrary restriction on the proximity of adjacent control points. No control point may be digitised if it lies within 2cm of an existing control point. This restriction is designed to prevent the user "double digitising" a control point. If your source document has control points which are closer than 2cm apart the ISTART digitising mode cannot be used.

Logical names required for ISTART digitising table mode.

ISTART requires that 2 logical names be assigned before digitising mode can be used:

- o LSL\$TK should be assigned to the serial line used by the Laser-Scan MUART device. ISTART will translate this logical name if the /MUART qualifier is specified.
- o LSL\$MONITOR_TABLE should be assigned to the serial line attached to the digitising table controlled by the TABLE_MONITOR or which ISTART is to probe with direct table reads. ISTART will translate this logical name if the /TABLE qualifier is specified.

Functions available in all ISTART modes.

The first IFF layer may be specified with the /LAYER qualifier, but defaults to 1. Either registration ticks (the default) or symbols (/SYMBOLS qualifier) will be placed in layer 0 at the location of the control point values.

ISTART offers the user the option of detailed explanatory prompts for user input. These are especially useful if digitising mode is invoked. It is recommended that the inexperienced user specified the /LEARNER qualifier on the ISTART command line to get these additional prompts.

By default ISTART checks the control point values supplied by the user. These checks are to ensure that the control points are cyclic (i.e. no "bow tie" deformation of digitising space is implied) and that the values supplied form a viable set of controls for Laseraid digitising. /NOCHECK may be used to switch off ISTART control point checking. It is strongly recommended that the checks are used at all times. No checks are performed in Ordnance Survey mode as the control point values are generated by ISTART itself using the map extent(s) information supplied by the user.

ISTART and LASERAID

By default ISTART creates an IFF output file which is unsuitable for use within the Laser-Scan Laseraid digitising system. The /LASERAID qualifier causes ISTART to generate an IFF file which is compatible with Laseraid. The Laseraid variant of the ISTART IFF file differs from the default ISTART IFF file in two important respects:

- (i) It contains an SH (Sector Header) entry which comes after the HI (HISTORY) entry. This contains a pointer to the first junction block within the file.
- (ii) No layer 0 is generated (except in Ordnance Survey mode). Laseraid itself generates registration tick features in layer 0. If Ordnance Survey mode is selected a grid will be placed in layer 0 but no registration ticks.

Files created using the /LASERAID qualifier should not be used as templates for other digitising systems (e.g. DIGSYS or LITES2) unless they have first been used by Laseraid in 'OLD' mode and have been processed using LAPROCESS. (See the separate Laseraid documentation for details of how to use these programs.)

The Laseraid compatible files may be selected as output in conjunction with any other valid combination of ISTART qualifiers, in template mode, digitising mode or in Ordnance Survey mode.

ISTART and digitising table characteristics

ISTART will operate in digitising table mode on any Laser-Scan supported digitising table. To provide the maximum flexibility ISTART permits the user to define the digitising table addressability for conversion of table counts into millimetres or thousandths of an inch. This is done with the /ADDRESSABILITY qualifier (q.v.). The default digitising table addressability is 0.02mm. (i.e. 50 table counts per millimetre).

Similarly, the user may redefine the number of digitising puck buttons recognised by ISTART by using the /BUTTONS qualifier (q.v.). ISTART, by default, expects a 16 button qualifier, with the buttons 0 through 13 available for digitising and button 14 and 15 set for point abandon and session abandon respectively. The minimum number of puck buttons that can be used with ISTART is 3; one for digitising, one for point abandon and one for session abandon.

EXAMPLES

\$ ISTART CONTOURS<CR>

Specify map scale, eg 10000 : 10000<CR>

Specify coordinates for the top left control point (X then Y) : 0 1000<CR>

Specify coordinates for the bottom left control point (X then Y) : 0 0<CR>

Specify coordinates for the bottom right control point (X then Y) : 1000 0<CR>

Specify coordinates for the top right control point (X then Y) : 1000 1000<CR>

ELAPSED: 0 00:00:24.71 CPU: 0:00:00.55 BUFIO: 12 DIRIO: 14 FAULTS: 77

\$

This example shows a successful run of ISTART in template (default) mode. Output is to LSL\$IF:CONTOURS.IFF;0. ISTART prompts for the map scale and the control point values only. The origin offset will be derived automatically from the value of the bottom left control point. All the the control point values will have the offset subtracted from them to make the bottom left control point (0.0,0.0). The control points will be checked for cyclic and distribution errors. The control points specified have passed the tests.

\$

\$ ISTART/OS SQ100450<CR>

Specify map scale, eg 10000 : 50000<CR>

Specify map extent, (X-extent [Y-extent]) : 40000 40000<CR>

Specify grid step, eg. 1000 : 1000<CR>

Specify origin offset X then Y : 100000 450000<CR>

ELAPSED: 0 00:00:41.71 CPU: 0:00:01.48 BUFIO: 10 DIRIO: 27 FAULTS: 104

\$

This example shows a successful run of ISTART in O.S. mode. Output is to LSL\$IF:SQ100450.IFF;0. ISTART prompts for the map scale and the parameters which determine the characteristics of an Ordnance Survey sheet; the sheet extent, the grid step and the sheet origin offset. The

control point values are calculated from the extent value(s) supplied. The grid step is used to calculate the coordinates of (axis extent/grid step)+1 grid lines for each axis. The grid lines are output as IFF features each having two coordinates in IFF layer 0. The bottom left control point is given the value (0.0,0.0) and the other three are offset relative to this. \$STATUS is set to SS\$_NORMAL.

\$

\$ ISTART/OS SQ100450/LEARNER<CR>

ISTART creates an IFF output file. This file will be sent to the directory pointed to by the logical name LSL\$IF:. Use SI to assign the logical name LSL\$IF: to the directory where you intend the output from ISTART to go. If LSL\$IF: does not point to the right directory it is probably best to quit now.

Do you wish to continue [YES/NO]? **YES<CR>**

You must now specify the map scale. The scale denominator only must be specified, ISTART cannot handle a representative fraction. To specify 1:50000 scale, for example, just type 50000

Specify map scale, eg 10000 : **10000<CR>**

You must now specify the map extent. It is important to get this right as it will determine the values used in the IFF CP entry, the IFF range and the characteristics of the grid. The map extent is expressed in metres and is usually tied to an O.S. sheet format. For example a 1:50000 scale sheet usually has an extent of 40 x 40 km ie an extent of 40000 metres in X and 40000 in Y. To specify this extent just type 40000. The Y extent will default to that of the X extent.

Typical scale/sheet extents are:

SCALE	X-EXTENT	Y-EXTENT
1:1250	500	500
1:2500	1000	1000
1:10000	5000	5000
1:25000	20000	10000
1:50000	40000	40000
1:250000	270000	220000

Specify map extent, (X-extent [Y-extent]) : **5000<CR>**

%ISTART-W-EQLEXT, X and Y extent values assumed equal

You must now specify the grid step in metres. The value that you specify determines the spacing between the grid lines. On a 1:50000 scale map the grid step is usually 1000 metres ie 1Km. To specify this grid step you would have to type 1000.

Specify grid step, eg. 1000 : **1000<CR>**

You have specified the /USER_ORIGIN qualifier and so you must now specify the map origin offset. The numbers that you supply will be written into the IFF MD (Map Descriptor). The origin offset will be used by IMP modules like IMERGE to determine the relationship of the data held in this file with data relating to neighbouring maps. The origin offset should be specified as two floating point numbers, X then Y, eg 11000.0 91000.0.

Specify origin offset, X then Y : **560000 270000<CR>**

ELAPSED: 0 00:02:45.95 CPU: 0:00:04.63 BUFIO: 56 DIRIO: 104 FAULTS: 97

\$

This is another example showing a successful run of ISTART in O.S. mode. This time, however, the /LEARNER qualifier was specified. The user is given help on the meaning of each of the prompts. \$STATUS is set to SS\$_NORMAL.

\$ ISTART

_IFF file for output: **TST.IFF/TABLE/LOG/MCE/SYMBOLS/LAYER=37<CR>**

%LSLLIB-I-IFFOPENED, LSL\$IF:TST.IFF opened for write

Specify map scale, eg 10000 : **50000<CR>**

Use any cursor button to digitise the TOP LEFT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to digitise the BOTTOM LEFT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to digitise the BOTTOM RIGHT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

This is more than 0.33 mm away from the last measurement!

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

%TSTART-control point abandoned - try again

Use any cursor button to digitise the BOTTOM RIGHT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to digitise the TOP RIGHT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Control points (in mm) minus offset:

NW (top left)	0.892	382.960	0.892	382.960
SW (bottom left)	0.000	0.000	0.000	0.000
SE (bottom right)	179.400	0.700	179.400	0.700
NE (top right)	176.360	385.093	176.360	385.093

Origin offset is 0.0 in X and 0.0 in Y

ELAPSED: 0 00:00:44.42 CPU: 0:00:00.53 BUFIO: 67 DIRIO: 13 FAULTS: 141
\$

In this example ISTART has been put into digitising mode using the /TABLE qualifier. Input will be taken from a TABLE_MONITOR driven digitising table. The user has selected the /SYMBOLS option which will result in single point registration mark symbols being generated as IFF features in layer 0 of the output file. (The default action is to produce "L" shaped registration ticks). ISTART will create an empty layer 37 in the output file (/LAYER=37). The user is prompted for digitising table cursor button presses for each control point in turn. Each control point is to be digitised 3 times (the default repeat value). Notice that the user badly digitised the top right control point. Successive button presses revealed a locational error greater than the ISTART tolerance for repeated digitising errors. The user subsequently pressed the abandon point button and completely redigitised the control point. As /LOG was selected the complete control point values are displayed at the end of the run. These control point values passed the ISTART control point cyclicity and distribution checks. An MCE type MH (Map Header) will be generated in the output file as /MCE was specified. The origin offset is set to (0.0,0.0). \$STATUS is set to SS\$_NORMAL.

\$

\$ ISTART<CR>

_IFF file for output: **TST/TABLE/THOU/CENTRE/LOG/LASERAID<CR>**

%LSLLIB-I-IFFOPENED, LSL\$IF:TST.IFF opened for write

Specify map scale, eg 10000 : **24000<CR>**

Use any cursor button to digitise the TOP LEFT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to digitise the BOTTOM LEFT control point, except:

- (i) the highest cursor button (which abandons the whole session),
- (ii) the highest but one button (which abandons just the current control point).

Use any cursor button to repeat the measurement, except:

- (i) the highest cursor button (which abandons the whole session),

(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to repeat the measurement, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to digitise the BOTTOM RIGHT control point, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to repeat the measurement, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to repeat the measurement, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to digitise the TOP RIGHT control point, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to repeat the measurement, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
Use any cursor button to repeat the measurement, except:
(i) the highest cursor button (which abandons the whole session),
(ii) the highest but one button (which abandons just the current control point).
%ISTART-E-CPNONCYC, ISTART/CHECK finds that the control points are non-cyclic -
output file deleted
Control points (in mm) minus offset:
NW (top left) 0.000 60000.000 0.000 60000.000
SW (bottom left) 0.000 0.000 0.000 0.000
SE (bottom right) -530000.0 -153000.0 -530000.0 -153000.0
NE (top right) -490000.0 -147000.0 -490000.0 -147000.0
Origin offset is 570000.0 in X and 170000.0 in Y
ELAPSED: 0 00:02:07.03 CPU: 0:00:03.82 BUFIO: 49 DIRIO: 93 FAULTS: 102
\$

This is another example of ISTART in digitising mode. In this case the user has specified that the origin should be moved to the centre of gravity of the control points and that the table counts should be converted into thousandths of an inch. Unfortunately, the user has digitised control points which have failed the checks for cyclicity. Had the checks been successful, a Laseraid compatible IFF file would have been generated as output. \$STATUS is set to SS\$_ABORT.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user. They are often used to supply explanatory information in support of a warning or error message.

DTABUSED, direct table read mechanism used

Explanation: ISTART will by default attempt to read from a digitising table using the TABLE_MONITOR process. If TABLE_MONITOR is not running, or is not available at your site, ISTART will instead use direct table reads and interpret the codes returned from the table itself. If this message appears direct table reads will be used.

User action: None.

MINEXT, minimum extent allowed is 10

Explanation: Ordnance Survey mode has been selected and a map extent value has been specified which is less than 10. The O.S. map header basic grid interval is calculated using the formula (map extent/10). Fractional basic grid intervals are not allowed.

User action: Re-specify the map extent value(s)..

PNTABA, control point abandoned - try again

Explanation: A digitising table option has been selected and the abandon point cursor button has been pressed. If you need help on the cursor button definitions re-run ISTART in digitising table mode with the /LEARNER qualifier.

User action: Re-digitise the control point.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

BIGGRD, grid step too big - grid step must be less than grid extent for the relevant axis

Explanation: Ordnance Survey mode has been selected and a map grid step value has been specified which is greater than the [minimum] map extent.

User action: Re-specify the grid step.

CMD, error reading command

Explanation: An error has occurred reading the command specified in response to a prompt for input.

User action: Re-specify the command, taking account of any accompanying message which should indicate the cause of the problem.

Explanation: ISTART allows the map extent to vary between the X and Y axes. If only one extent value is supplied in response to the prompt then the X and Y extents are assumed to be equal.

User action: None.

CPCLOSE, current control point too close to 'control point' control point

Explanation: A digitising table option has been selected and an attempt has been made to digitise a control point which is less than 2 cm away from an existing control point. This restriction is imposed to prevent a user from superimposing control points.

User action: Ensure that you are trying to digitise the correct control point. If you are, then the source document is too small to be digitised using ISTART.

DUFFQIO, bad QIO - attempting to continue

Explanation: A digitising table option has been selected and a bad QIO has been sent by ISTART while probing the digitising table serial line or reading from the table. A QIO is a Queued I/O (input/output) request to the system. This message should not normally occur.

User action: None, the problem should right itself. If however this happens frequently, please contact Laser-Scan.

INVBUT, invalid button press

Explanation: A digitising table option has been selected and an unknown cursor button has been pressed. If you need help on the cursor button definitions re-run ISTART in digitising table mode with the /LEARNER qualifier.

User action: Press either the highest but one cursor button to abandon the point or the highest cursor button to abandon the whole session. Press any other cursor button to digitise the point.

INTEXT, invalid extent - re-enter extent values

Explanation: Ordnance Survey mode has been selected and a invalid map extent value has been specified. The accompanying message should indicate the cause of the problem.

User action: Re-specify the map extent value(s).

MONNOTAV, TABLEMONITOR not available

Explanation: The TABLE_MONITOR process is not available. ISTART will attempt to continue using direct reads from the digitising table.

User action: Unless TABLE_MONITOR monitor availability was expected, none.

NOINPUT, no table input device initialised - attempting to continue

Explanation: None of the available digitising table options were initialised. This should never happen!

User action: Check that the digitising table is on-line and (if required) that the TABLE_MONITOR is running. Check that logical name LSL\$TK: is assigned to the correct device. Try running ISTART again. If the problem persists please contact Laser-Scan.

NOMOREBUTS, type-ahead buffer too full ('integer') for button presses - wait for a few seconds

Explanation: ISTART is using direct table reads to get data from the digitising table. There have been so many button presses that ISTART has not been able to keep up! There are currently 'integer' button presses in the type-ahead buffer. This may occur if the host computer is overloaded. A higher process priority may ease the problem.

User action: Wait for ISTART to catch up. If you believe that you have only pressed a few buttons check that the digitising table is not in 'stream' mode. If it is then switch the table controller to 'point' mode and re-run ISTART. If the problem persists please contact Laser-Scan.

SMALLGRD, grid step too small - minimum grid step allowed is 1

Explanation: Ordnance Survey mode has been selected and a map grid step value has been specified which is less than one.

User action: Re-specify the grid step.

TABERRREAD, error reading from digitising table - data returned is 'data-string'

Explanation: ISTART is reading from the digitising table using direct table reads. The coded data string 'data-string' returned by the table is not recognised by ISTART.

User action: If the problem persists please contact Laser-Scan.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. It is most unlikely that any output file produced will be correctly processed.

ABA, session abandoned

Explanation: The current ISTART session has been abandoned. The user has either entered <CTRL/Z> (holding down the key marked Ctrl and then pressing Z) in response to a prompt for input or else the highest numbered cursor button has been pressed while in table digitising mode.

User action: None, the session is irrevocably terminated. Re-run ISTART if the session was terminated in error!

CPNONCYC, ISTART/CHECK finds that the control points are non-cyclic - output file deleted

Explanation: By default ISTART checks that the control point values specified (either via the keyboard or the digitising table) are cyclic. If they are non-cyclic then a "bow-tie" distortion of map space is implied and this message is output. ISTART will not create an output IFF file.

User action: Re-run ISTART and re-specify the control points in a consistent rotational order.

CPNOTVIA, ISTART/CHECK finds that the control points are not viable for use by Laseraid

Explanation: By default ISTART checks that the control point values specified (either via the keyboard or the digitising table) are viable for use in Laseraid and other Laser-Scan digitising and processing programs. The control points output to the terminal following this message indicate the values that have been specified.

User action: Read the accompanying message text that is on the terminal screen and re-specify the control points. If you really do want these values then specify the /NOCHECK qualifier on the command line. You do this at your peril.

FLDTOTRANS, failed to translate logical name "logical name"

Explanation: The specified logical name could not be translated.

User action: Assign the logical name to the appropriate device. If in doubt please contact Laser-Scan.

INITAB, error initialising TABLE MONITOR

Explanation: The table monitor could not be initialised. The accompanying message will provide more detailed information.

User action: Use the TABLE_MONITOR documentation to determine the cause and remedial action for this error.

INVALFC, invalid /FC argument - "'integer'" is outside range 0 to 32767

Explanation: IFF feature codes lie in the range 0 to 32767.

User action: Respecify the ISTART command line and ensure that the argument for the /FC qualifier lies within the specified range.

INVALLAY, invalid /LAYER argument - "'integer'" is outside range 1 to 32767

Explanation: IFF layer numbers lie in the range 0 to 32767.

User action: Respecify the ISTART command line and ensure that the argument for the /LAYER qualifier lies within the specified range.

INVALREP, invalid /REPEAT argument "'integer'" - should be positive integer

Explanation: The /REPEAT qualifier is used to set the number of times that a control point is to be digitised before the mean position for that point is calculated. It is clearly nonsense to specify a /REPEAT argument which is less than 1.

User action: Respecify the ISTART command line and ensure that the argument for the /REPEAT - qualifier is greater than or equal to 1. The default count for /REPEAT is 3.

INVBNUM, invalid /BUTTONS=NUMBER argument 'integer' - should be greater than or equal to 3

Explanation: One role of the /BUTTONS qualifier is used to set the number of buttons available on a digitising table puck. ISTART requires a minimum of three 3 buttons:

- o a digitising button,
- o an "abandon-this-control-point" button, and,
- o an "abandon-this-whole-session" button.

User action: Respecify the ISTART command line and ensure that the argument for the /BUTTONS "NUMBER" keyword is greater than or equal to 3.

LINENOTTERM, direct table serial line "'line-id'" is not set up as a terminal

Explanation: ISTART attempted to read from the digitising table using direct table reads. The specified serial line is not, however, set up as a terminal. The digitising table must be on a serial line which is set up in this way.

User action: Ensure that the logical name LSL\$MONITOR_TABLE is assigned to the correct serial line. Ensure that this serial line is set up as a terminal with TYPEAHEAD and NOBROADCAST. Re-run ISTART.

STOPMON, error stopping table monitor

Explanation: This error is given if the table monitor is unable to hibernate after ISTART has finished with the digitising table. The table monitor will continue to be active. This should never happen.

User action: Use the VMS SHOW USERS command to ascertain whether the TABLE_MONITOR is still running. If it is, then stop it using the STOPMON command and then restart it using the STARTMON command. If this fails to cure the problem consult the TABLE_MONITOR documentation.

TABINUSE, TABLE_MONITOR already in use

Explanation: The TABLE_MONITOR process is already in use by another process.

User action: Check who is using the table monitor and use STOPMON to stop the table and STARTMON to restart the table monitor if appropriate.

TABREAD, error reading from digitising table

Explanation: Occasionally the digitising table returns a garbled coordinate string. This error message allows the user to detect such an event.

User action: If this happens frequently then it should be reported to Laser-Scan.

TABREP, TABLE MONITOR reports: 'report text'

Explanation: An error occurred reading from a TABLE_MONITOR driven digitising table. The TABLE_MONITOR passes the error report text to ISTART.

User action: The 'report text' included in this message will, in conjunction with the TABLE_MONITOR documentation, indicate what to do.

ZEROBASE, zero length baseline (BOTTOM_LEFT - BOTTOM_RIGHT) - superimposed control points?

Explanation: ISTART uses the base line defined by the bottom left and bottom right control points to calculate correction for rotation of the source document on the digitising table. This message will not normally occur as ISTART checks that no two control points lie within 2 cm of each other on the digitising table. The control points may, however, have been digitised in a peculiar order or all four along a vertical straight line!

User action: Re-run ISTART and re-digitise the control points. This time ensure that the bottom left and bottom right control points are not coincident. Also ensure that the map has not been taped down on the digitising table such that some of the control points lie within the "dead" area of the digitising table.

MESSAGES (OTHER)

In addition to the above messages which are generated by ISTART itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the output file should be preserved to facilitate diagnosis. Generally, however, ISTART will attempt to delete any corrupt output file before program termination.

CHAPTER 19

MODULE ITOTEXT

MODULE ITOTEXT

REPLACES DAMP module I2TEXT.

FUNCTION

ITOTEXT reads an IFF file and outputs a text file. Each IFF entry is output with the appropriate two letter code followed by the accompanying entry information. ITOTEXT is designed to be used for the conversion of data into text form for transfer to external customers.

FORMAT

\$ ITOTEXT IFF-file-spec text-file-spec

Command qualifiers**Defaults**

/[NO]ABSOLUTE

/NOABSOLUTE

/[NO]ADDRESS

/NOADDRESS

/[NO]BRIEF

/NOBRIEF

/[NO]CUSTOMER

/NOCUSTOMER

/[NO]DECIMAL_PLACES = integer

/NODECIMAL_PLACES

/FRT = file-spec

No FRT file is specified

/[NO]MH

/MH

/[NO]KEEP

/NOKEEP

/[NO]LOG

/NOLOG

/[NO]PRINTER

/NOPRINTER

/[NO]QUOTED_TEXT

/QUOTED_TEXT

/REVISION_LEVEL = integer

/REVISION_LEVEL = -1

/[NO]WIDTH = integer

/NOWIDTH

Output may be directed to a terminal by specifying SYS\$OUTPUT: as the output file-spec.

PROMPTS

IFF-file: IFF-file-spec
Text-file: text-file-spec

PARAMETERS**input-file-spec**

- specifies the IFF file from which a text file is to be made. Any part of the input-file-spec which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF;0'.

text-file-spec

- specifies the text file which is to be created. Any part of the output-file-spec which is not explicitly supplied will be taken from:

"SYS\$DISK:" + 'input-file-name' + ".TXT".

If a file with the specified name already exists a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS

/ABSOLUTE

/NOABSOLUTE (default)

- if /ABSOLUTE is specified all coordinate values output are converted to absolute values using the origin offset in the type 2 Map Descriptor ('MD') entry. The numbers output may be produced with a user specified format using the qualifiers /WIDTH and/or /DECIMAL_PLACES. Without these settings, the numbers are output at maximum precision with minimum width.

/ADDRESS

/NOADDRESS (default)

- causes the address (hexadecimal) of the IFF entry to be output at the end of the line of text. The address is prefixed by an exclamation mark which will be treated as a comment and ignored by IFFTEXT.

/BRIEF
/NOBRIEF (default)

- a shortened output file is produced. No coordinates are output. (See the Description Section for further details).

```
+-----+
|                                     |
|               WARNING               |
|                                     |
| The output file produced by the /BRIEF qualifier |
| is not valid for use with IFFFROMTEXT.           |
|                                     |
+-----+
```

/CUSTOMER
/NOCUSTOMER (default)

- if /CUSTOMER is specified no MH (Map Header), CC (Cubic Coefficients) or HI (HISTORY) entries are written to the text output file. /CUSTOMER is not available for use with the /MH qualifier.

/DECIMAL_PLACES = integer
/NODECIMAL_PLACES (default)

- if /DECIMAL_PLACES is specified all coordinate values output are converted to absolute values using this number of decimal places. The allowed value lies between 0 and 16 to produce sensible double precision values, if required, on output and must be less than the value specified with the /WIDTH qualifier if given.

/FRT = file-spec

- specifies an FRT file which contains an ACD definitions table. Any part of the file specification not supplied is taken from the default LSL\$FRT:FRT.FRT.

This qualifier should only be specified if non-standard ACD definitions are required to define attributes in CB entries. (See IFF user guide for a description of standard ACD definitions)

/KEEP (default)
/NOKEEP

- available only with the /PRINT qualifier. /NOKEEP causes deletion of the output file after printing.

/LOG
/NOLOG (default)

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are output when files are successfully opened, and a reassuring message output each time a new IFF layer is encountered in the IFF file.

/MH
/NOMH (default)

- if /MH is specified the map header is interpreted as an MCE map header and is output in readable format. /MH is not available for use with the /CUSTOMER qualifier.

/PRINTER
/NOPRINTER (default)

- queues the ITOTEXT output for printing under the name given as the output file-spec.

/QUOTED_TEXT (default)
/NOQUOTED_TEXT

- if /QUOTED_TEXT is specified all text values output are surrounded by quotes. This qualifier is set on by default, and the /NOQUOTED_TEXT qualifier must be used to output the text without them.

/REVISION_LEVEL = integer

- defines the representation of CB, ST and ZS entries in the IFF file. The revision level takes one of three values -1, 0 or 1. The default is revision level -1. The various values operate as follows: -
 - o /REVISION_LEVEL = -1 - all CBs, ZSs and STs are output unchanged to the text file.
 - o /REVISION_LEVEL = 0 - all CBs are output as ZSs or STs in the text file.
 - o /REVISION_LEVEL = 1 - all STs or ZSs are output as CBs in the text file.

/WIDTH = integer
/NOWIDTH (default)

- if /WIDTH is specified all coordinate values output are converted to absolute values using this field width. The allowed value lies between 0 and 20, (and must be greater than the number given with the /DECIMAL_PLACES qualifier, if supplied) to produce sensible (double precision) values on output, if required.

RESTRICTIONS

- o /CUSTOMER and /MH are mutually exclusive
- o IFROMTEXT cannot be used on a text file produced using the ITOTEXT /BRIEF qualifier

DESCRIPTION

ITOTEXT reads an IFF file and outputs a readable text file. Each IFF entry is output with the appropriate two letter entry code followed by the accompanying entry information.

It is envisaged that ITOTEXT will be used for the conversion of data into text form suitable for transfer to external customers. Alternatively ITOTEXT can be used in conjunction with IFROMTEXT to enable non-graphical editing of IFF files. This is done by converting an IFF file into text form using ITOTEXT, editing the text file, and creating a new IFF file from the edited text file using IFROMTEXT.

It is recommended that the user reads the "IFF User Guide" before attempting to edit the text file produced by ITOTEXT. In particular the user should be familiar with the section describing the structure of an IFF file.

ITOTEXT and the HI entry.

If a HI entry is present in the IFF file a full text expansion is made of every record used in the HI entry. (For a description of the effect of the /CUSTOMER qualifier see below). The following is an example of a HI entry text expansion:

```
HI
27-MAR-1987 14:13 TIM      ISTART   Create      00:00:10 00:00:00 00000001
27-MAR-1987 14:23 TIM      IPATCH    Modify      00:00:10 00:00:00 00000001
```

It is not possible to reformat an HI entry text expansion back into an IFF file using IFROMTEXT. It is considered that the HI entry should only be updated by IFF utility programs, thus preventing the falsification of HI entry records by the user.

ITOTEXT and the MH entry.

ITOTEXT will display the contents of a type 2 MH entry as longword integers. Those fields which contain real numbers are not interpreted as floating point values.

An example of a type 2 MH entry, when converted to text is given below:

```
MH 174 2
      0      436500      122000      1250      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      436500      122000      0      0
      50      1250      6500      0      0      0
      0      0      0      0      0      0
```



```

      0      0      0      0      0      0
      0      0      0      0      4260880      4260880
4260880      4260880      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
4260880      4260880      4260880      4260880      4260880      4260880
      0      0      0      9439170      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0
      0      0      0      0      0      0

```

If the IFF MH entry is a type 3 or 4 OS Map Header, then it is displayed as an ASCII block in lines of 80 characters. Type 3 and 4 OS Map Headers are only intended for use by Ordnance Survey (Great Britain).

Here is an example of a type 3 OS Map Header when converted to text as an ASCII block.

```

MH 0 3
-1 0 436500 122000 50 1250 0 6500 -20 1
    9 86 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0

```

0 0

----- ITOTEXT and the MD entry.

Two text representations of a version 2 MD (Map Descriptor) entry are produced by the ITOTEXT program. The first is unformatted and is included if ITOTEXT is used for outputting the IFF file for subsequent retranslation with IFROMTEXT, eg.:

```

MD      2      17352      0      17224
      0      17786      0      0      0
0
      0      0      0      0      0
0
      etc.

```

The second formatted version 2 MD is preceded by a '!' and is thus ignored by IFROMTEXT, eg.:

```
!MD LO 0.0000000000000000 0.0000000000000000
! SC 0.0000000000000000
! PR 0
! UN 0
! SP 0
```

A version 1 map descriptor is written only once, and will be automatically recognised by IFROMTEXT. IFROMTEXT considers a version 1 map descriptor to be one where an alphabetic character follows the MD rather than the numeric values of the unformatted version 2, eg.:

```
MD MA -1      0.0      0.0      0.0      0.0
GR 0         0.0      0.0      0.0      0.0
SC          0.0
PS 0 0
AG 0         0.0      0.0      0.0      0.0
```

Modification of ITOTEXT output by the /CUSTOMER and /NOMH qualifiers.

Although by default ITOTEXT will translate the entire contents of the IFF file into a textual representation, the treatment of specific entries will vary depending upon the selection of the /CUSTOMER and /MH qualifiers.

Display of the following entries will be suppressed if the /CUSTOMER qualifier is specified:

- o **MH** - Map Header
- o **CC** - Cubic Coefficients
- o **HI** - HHistory entry

Alternatively, display of the MH entry only will be suppressed if the /NOMH qualifier is specified.

ITOTEXT and the /BRIEF qualifier

Some user applications require only a list of the entries present within an IFF file without a need for detailed listings of the contents of each entry.

The /BRIEF qualifier enables ITOTEXT to produce such summary output. Such output will not of course be suitable for reconversion into an IFF file using IFROMTEXT.

The following example file shows the result of specifying the /BRIEF qualifier:

```
! IFF file in brief - not valid for use with IFROMTEXT
RA -0.729576 400.1828 -0.156608 401.0827
MH
MD
NS
CC
CP
NO 1 0
NF 1 1
TH 30
FS 5 0 0 0
AC 3 170.0
ST 124 0
ST 116 1
EF
EO
EM
EJ
```

ITOTEXT and the /ABSOLUTE qualifier

Some applications, in particular the Laser-Scan Gothic platform, require all coordinates to be full or absolute values, created using the origin offset stored in the MD Map Descriptor IFF entry.

The /ABSOLUTE qualifier enables ITOTEXT to produce these absolute coordinate values for all entries, in particular the RA Range, the CP Corner Point and all coordinate entries (ST, ZS and CB). In addition, the use of /ABSOLUTE allows the user to specify the output format for these double precision numbers, with the /WIDTH qualifier (with values between 0 and 20), specifying the overall size of the numbers, and the /DECIMAL_PLACES qualifier (with values between 0 and 16, and less than the /WIDTH size), specifying the number of decimal places to use for the numbers.

In the absence of these two format qualifier extensions for the /ABSOLUTE option, all double precision numbers are output with maximum precision in minimum width.

The qualifiers /WIDTH and /DECIMAL_PLACES are allowed with or without the /ABSOLUTE qualifier.

ITOTEXT and IFF junction structure

ITOTEXT produces text output for SH, JP and JB entries. These are not recoverable with IFROMTEXT. If junction structure is required in the IFF file produced by IFROMTEXT, the file should be processed using the ILINK utility (available as part of the STRUCTURE package).

ITOTEXT and text file editing

ITOTEXT example output

The following example illustrates ITOTEXT output. The /ADDRESS qualifier is used to produce the hexadecimal addresses (in words from the beginning of the IFF file) separated from the text by "!" comment delimiters.

[illegible]

The same file output with `/REVISION_LEVEL = 1` would result in the ST being converted to a CB and output as follows.

```
CB FL 0
  GT 1
  NR 4
  NC 2
  NA 0
  HC 91 92
  DA -0.534355    14.83076
      0.850608    16.83937
      2.229042    18.84541
      2.87809     20.61997
```

EXAMPLES

```
$ ITOTEXT<CR>
IFF-file: TST.IFF/LOG<CR>
Text-file: TST<CR>
%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.TRIALS]TST.IFF;8 opened for read
%ITOTEXT-I-OPTXT, text file DUA3:[BUREAU.TRIALS]TST.TXT opened for output
%ITOTEXT-I-LAYER, layer 1 found
%ITOTEXT-I-LAYER, layer 88 found
%ITOTEXT-I-LAYER, layer 66 found
%ITOTEXT-I-LAYER, layer 55 found
%ITOTEXT-I-LAYER, layer 44 found
%ITOTEXT-I-LAYER, layer 11 found
  ELAPSED: 00:00:09.83  CPU: 0:00:03.38  BUFIO: 6  DIRIO: 21  FAULTS: 118
$
```

This example shows the default mode of operation. No parameters are specified and ITOTEXT prompts for the input IFF file and output text file. The various messages are informational and are displayed because /LOG has been specified.

```
$ ITOTEXT LOTTEST/LOG/BRIEF/ADDRESS OUTFIL<CR>
%LSLLIB-I-IFFOPENED, DUA0:[LSL.IFF]LOTTEST.IFF;3 opened for read
%ITOTEXT-I-OPTXT, text file DUA0:[LSL.IFF]OUTFIL.TXT opened for output
%ITOTEXT-I-LAYER, layer 1 found
%ITOTEXT-I-LAYER, layer 2 found
%ITOTEXT-I-LAYER, layer 3 found
  ELAPSED: 00:00:01.95  CPU: 0:00:00.83  BUFIO: 6  DIRIO: 18  FAULTS: 115
$
```

Because /ADDRESS has been specified the output file will include the address of each entry at the end of the appropriate line. The /BRIEF qualifier is used to allow the user to see the contents of the file quickly without having to read through all the coordinates of each string entry.

MESSAGES (SUCCESS)

These messages are used to indicate that the program has succeeded in performing some action, and do not require any user action.

NORMAL, normal successful completion

Explanation: This message informs the user that the program run was successful

User action: None.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user. They are used to provide information on the current state of the program, or to supply explanatory information in support of a warning or error message.

LAYER, layer 'integer' found

Explanation: This message informs the user that the specified layer was found.

User action: None.

OPTXT, text file 'file-spec' opened for output

Explanation: This message informs the user that the specified file was successfully opened.

User action: None.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user or that the program will attempt to overcome.

ERRMD2, version 2 map descriptor error

Explanation: This message warns the user that the Map Descriptor was not of type 2.

User action: Check the IFF file.

MDDEFAULT, Default values used for the origin offset

Explanation: This message informs the user that the program was only able to find a 'MD' type 1 map descriptor entry in the input IFF file when the /ABSOLUTE qualifier was given, rather than a type 2 containing an origin offset. A default origin offset of (0,0) will be used.

User action: This may be due to the corruption of data in the IFF file, so check the IFF file.

UNKNIFFENT, unknown IFF code 'entry' at address 'hex' - ignored

Explanation: This message informs the user that an unknown IFF entry was encountered and ignored.

User action: Check the IFF file.

MESSAGES (ERROR)

These messages indicate an error in processing which will cause the program to terminate. The most likely causes are a corrupt or otherwise invalid input file, or an error related to command line processing and file manipulation.

BADNUMDP, Unacceptable number of decimal places: 'integer'

Explanation: This message informs the user that there was an unacceptable number of decimal places for real coordinate output values given with the /DECIMALPLACES qualifier. The supplied number should be between 0 and 16.

User action: Re-enter the command line.

BADWIDTH, Unacceptable number for the format width: 'integer'

Explanation: This message informs the user that there was an unacceptable number given for the width field for real absolute coordinate output given with the /WIDTH qualifier. The supplied number should be between 0 and 20, and greater than the supplied number of decimal places, if any was given.

User action: Re-enter the command line.

ERRCLO, error closing output file 'file-spec'

Explanation: This message informs the user that the specified file was not closed successfully.

User action: Check for any further messages.

ERROPOUT, error opening output file 'file-spec'

Explanation: This message informs the user that the specified file was not opened successfully.

User action: Check the file specification given, or for any further messages.

ERRREADFRT, error reading from FRT file 'file-spec'

Explanation: This message informs the user that there was an error reading the FRT file.

User action: Check the file specification given, or any further messages.

ERRWROUT, error writing to output file 'file-spec'

Explanation: This message informs the user that the specified file was not opened.

User action: Check the file specification given.

INVALREV, invalid revision level 'integer'

Explanation: This message informs the user that there was an unacceptable number given for the revision level qualifier. The supplied number should be 0, 1 or -1

User action: Re-enter the command line.

MDABSENT, Unable to find 'MD' entry in IFF file

Explanation: This message informs the user that the program was unable to find the 'MD' map descriptor entry from the input IFF file when the /ABSOLUTE qualifier was given. This may be due to the corruption of data in the IFF file.

User action: Check the input IFF file against data corruption.

UNEXPEOF, unexpected end of IFF file

Explanation: This message informs the user that the end of file was reached prematurely.

User action: Check the IFF file.

MESSAGES (OTHER)

In addition to the above messages which are generated by the program itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library and by the Laser-Scan I/O library, LSLLIB. IFF library messages are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user, and Laser-Scan are consulted, then the output file should be preserved to facilitate diagnosis. LSLLIB messages are introduced by '%LSLLIB' and are generally self-explanatory. They are used to explain the details of program generated errors.

CHAPTER 20

MODULE ITRANS

MODULE ITRANS

REPLACES Projection computation and transformation functions of IPR

FUNCTION

ITRANS is a general map projection program. It allows changes in geodetic datum, transformation between spheroidal data (latitude and longitude) and specified map projections (and will also transform data from one map projection directly into another).

It can also be used to transform data from one coordinate space to another, where the relationship between the coordinate systems are not known, but the values of several common points in each system are. Several different formulae are available for this approach, the required parameters being empirically determined from the common points.

ITRANS is primarily intended for operation on data held in IFF files, but it can also be used to transform individual points.

FORMAT

\$ ITRANS input-file-spec [output-file-spec]

Command Qualifiers**Defaults**

/[NO]ABSOLUTE	/NOABSOLUTE
/[NO]CONTROL	/NOCONTROL
/[NO]DATUM_SHIFT	/NODATUM_SHIFT
/DESCRIPTOR	None.
/[NO]HEIGHT_ADJUST	/NOHEIGHT_ADJUST
/[NO]FOUR	/NOFOUR
/[NO]LEARNER	/NOLEARNER
/[NO]LOG	/NOLOG
/[NO]OUTPUT[=file-spec]	/NOOUTPUT (see text)
/POINT	None.
/[NO]PROJECTIVE	/NOPROJECTIVE
/[NO]RETAIN_CP	/NORETAIN_CP
/[NO]SIX	/NOSIX
/[NO]THREE	/NOTHREE
/[NO]TWO	/NOTWO

PROMPTS

_Input File: input-file-spec
_Output File: output-file-spec

COMMAND PARAMETERS**input-file-spec**

Specifies the input IFF file to be operated on.

Any part of the file specification that is not supplied will be completed from the default of "LSL\$IF:IFF.IFF;0"

output-file-spec

Specifies the name of the IFF file to be created.

Any part of the file-spec that is not specified will be completed from the parsed input file-spec (without a version number).

A version number must not be included as the user has no control of the version number of the file to be created. A new file will be created with a version number one higher than the current highest existing in the directory.

RESTRICTIONS

- o ITRANS can only be used to convert data from one geodetic datum onto another, when the data is held as geographical coordinates (i.e. latitude and longitude)
- o /POINT and command parameters are mutually exclusive
- o /DESCRIPTOR and two command parameters are mutually exclusive
- o /POINT and /DESCRIPTOR qualifiers are mutually exclusive
- o /POINT and any of /FOUR, /PROJECTIVE, /SIX, /THREE and /TWO are mutually exclusive
- o /DESCRIPTOR and any of /FOUR, /PROJECTIVE, /SIX, /THREE and /TWO are mutually exclusive
- o Only one of /FOUR, /PROJECTIVE, /SIX, /THREE and /TWO are valid
- o /CONTROL can only be used in conjunction with one of /FOUR, /PROJECTIVE, /SIX, /THREE or /TWO
- o /SIX can only be used in conjunction with /CONTROL
- o /DATUM_SHIFT and any of /FOUR, /PROJECTIVE, /SIX, /THREE and /TWO are mutually exclusive
- o /HEIGHT_ADJUST can only be used in conjunction with /DATUM_SHIFT

- o /RETAIN_CP can only be used in conjunction with /DATUM_SHIFT
- o Empirical transformations cannot be carried out on geographical data (ie data held as latitude and longitude)
- o ITRANS does not usually perform transformations on the z value in ZS (3d string) or CB (coordinate block) entries but copies them, unaltered, to the output file. The x and y values are transformed. When doing the most accurate geodetic datum shift calculations, the z value is, however, altered.
- o ITRANS only transforms coordinates; it has no knowledge of any implied connections between these coordinates. It is the user's responsibility to ensure that data is suitable for the target projection. In particular, users should be aware that points that lie close together on one coordinate system may lie on different sides of a discontinuity on another. For example, two points with longitude 179.9 W and 179.9 E are close together on the surface of the earth but project to different extremities of a World projection with a central meridian through Greenwich. If, to avoid this problem, all data is split along 180.0 E, then this data will not be suitable to project onto a world projection with a different central meridian.

COMMAND QUALIFIERS

/ABSOLUTE
/NOABSOLUTE

- controls whether the numbers used as the right hand side of the control points when doing empirical transformations are to be retained, or whether they are all to be reduced by their minimum x and minimum y values and these values used as the local origin for the input data.

By default the values are reduced to give all the final control points positive values and also, usually, to reduce their absolute size.

/CONTROL[=file-spec]
/NOCONTROL

- specifies that a file containing the control points for an empirical transformation is to be used as opposed to those in the CP entry in the IFF file.

The file can contain up to twenty lines of control points in the order

ORIGINAL X ORIGINAL Y TARGET X TARGET Y

and comments may be included either, on separate lines anywhere in the file, or at the end of a line of control points. ITRANS will read ! and ; as comment characters and anything occurring before either of

these, other than the control points, will cause an error.

NOTE

The coordinates in this file should be entered as absolute values. Any origin offsets are handled automatically, so the user need not know about the relative coordinates that are actually stored in the file.

Each of the formulae available for empirical transformations will take up to the maximum number of lines of control points and, using the least squares method, return the best fit. Each of them also requires a minimum number of control points, as follows,

/TWO	minimum 2	maximum 800
/THREE	minimum 3	maximum 800
/FOUR	minimum 4	maximum 800
/PROJECTIVE	minimum 4	maximum 800
/SIX	minimum 6	maximum 800

If /CONTROL is entered with no file-spec, or a partial one, then the missing parts of the file-spec are taken from the default specification CONTROL.DAT.

The default is /NOCONTROL.

/DATUM_SHIFT
/NODATUM_SHIFT

- specifies that ITRANS is to carry out a conversion of geodetic datum.

The default is /NODATUM_SHIFT and ITRANS carries out projection transformations, rather than geodetic datum shifts.

/DESCRIPTOR

- specifies that ITRANS is only to update the map descriptor entry in the IFF file without doing any transformations on the data.

This qualifier should precede all other qualifiers on the command line.

By default, ITRANS does not allow the existing map descriptor entry to be altered, and transforms all the coordinate values in the file.

WARNING

If there is an existing version 2 map descriptor in the IFF file, then the MD entry will be overwritten in the existing file. If there is not a version 2 MD

entry then a new version of the file will be created.

/FOUR
/NOFOUR

- specifies that ITRANS is to carry out an empirical transformation on the data based on the the "extended four point" formulae.

See below for details of the transformation used.

The default is /NOFOUR, and ITRANS carries out a projection, rather than an empirical, transformation.

/HEIGHT_ADJUST
/NOHEIGHT_ADJUST

- specifies that ITRANS is to use the heights of points when doing geodetic datum conversion.

The default is /NOHEIGHT_ADJUST and ITRANS ignores the height of points when doing the conversion.

This command is only valid in conjunction with the /DATUM_SHIFT qualifier.

/LEARNER
/NOLEARNER

- specifies that ITRANS is to give help, before prompting for each bit of information.

The default is /NOLEARNER

NOTE

Help on any particular prompt can be obtained by responding with "?"

/LOG
/NOLOG

- controls the number of messages that ITRANS sends to SYS\$OUTPUT. E.g. when /LOG is specified messages are received when IFF files are successfully opened.

The default is /NOLOG

/OUTPUT[=file-spec]
/NOOUTPUT

- controls where the record of the session is to be sent.

The record of the session consists of the input and output map descriptors, any control points used for empirical transformations and either the name of the input and output IFF files or a list of the individual points that have been transformed (when the /POINT has been used).

If /OUTPUT is entered with no file-spec, or a partial one, then the missing parts of the file-spec are taken from the default specification SYS\$DISK:ITRANS.LIS.

By default this record is sent to SYS\$OUTPUT.

/POINT

- allows ITRANS to be used to transform individual points.

This qualifier should precede all other qualifiers on the command line.

An ASCII file of X and Y coordinate pairs can be read by giving the command "@filename" to the prompt asking for the coordinates of a point to convert.

By default ITRANS operates on IFF files.

/PROJECTIVE
/NOPROJECTIVE

- specifies that ITRANS is to carry out an empirical transformation on the data based on the "projective transformation" formulae.

See below for details of the transformation used.

The default is /NOPROJECTIVE, and ITRANS carries out a projection, rather than an empirical, transformation.

/RETAIN_CP
/NORETAIN_CP

- specifies that ITRANS is not to transform the CP (control point) entry while doing a geodetic datum conversion.

This qualifier is for use when the control points are graticule based, and the user wishes to move the data relative to them.

The default is /NORETAIN_CP and ITRANS transforms the control points as normal.

This command is only valid in conjunction with the /DATUM_SHIFT qualifier.

/SIX
/NOSIX

- specifies that ITRANS is to carry out an empirical transformation on the data based on the general second order transformation formulae.

See below for details of the transformation used.

The default is /NOSIX, and ITRANS carries out a projection, rather than an empirical, transformation.

/THREE
/NOTHREE

- specifies that ITRANS is to carry out an empirical transformation on the data based on the the "affine" formulae.

See below for details of the transformation used.

The default is /NOTHREE, and ITRANS carries out a projection, rather than an empirical, transformation.

/TWO
/NOTWO

- specifies that ITRANS is to carry out an empirical transformation on the data based on the the "orthogonal transformation" formulae.

See below for details of the transformation used.

The default is /NOTWO, and ITRANS carries out a projection, rather than an empirical, transformation.

DESCRIPTION**o The MD (Map Descriptor) entry.**

The MD (Map Descriptor) entry in the IFF file carries information about the projection that the coordinate data is currently held in. Each run of ITRANS will ask the operator for details of the required projection and the data in the file will be transformed into this projection in a new IFF file with a corresponding MD entry.

NOTE

The MD entry produced by ITRANS is of a different structure and length from that used by IPR, and thus the use of IPR and ITRANS are mutually exclusive. The new MD is called a "version 2 Map Descriptor"

If ITRANS is run in its projection or datum_shift mode on an IFF file without a valid, or complete, version 2 MD entry, the operator will be asked to enter details of the current projection before being asked for the required projection. The IFF file resulting from the transformation will contain a valid version 2 MD entry.

If however ITRANS is run in its empirical transformation mode on an IFF file without a valid or complete version 2 MD entry, then the operator will only be asked to supply the origin offset for the output IFF file. A version 2 MD entry will be created in the output file, but the other entries in it will be set to 0.0, or will be unset.

ITRANS can be used to edit an existing version 2 MD entry (or insert one, if none exists) without transforming any of the coordinate data.

The version 2 map descriptor contains the following information:-

i) The local origin for the IFF file.

This is an XY coordinate pair, each held as a REAL*8 (double precision) number, that is added to any pair of coordinates in the IFF file to give the true projection coordinates of the points.

This allows the absolute size of the coordinates in the IFF file, which are only held as REAL*4 (single precision) numbers to be reduced, thus avoiding problems of rounding and reduced accuracy.

ii) The scale of the map that the IFF file represents.

This is a REAL*8 (double precision) number which when multiplied by map (or sheet) units gives IFF units - e.g. 50000.0 for a 1:50000 map.

It is used to relate IFF units to e.g. symbol and text sizes on the map sheet and also as a general indication of the accuracy of the digital data.

iii) The spheroid that the data relates to.

All geographical data is ultimately related to a particular geodetic datum. A geodetic datum is defined by the shape and size of a reference spheroid and by its orientation with respect to the earth. The data about the spheroid is held in the map descriptor; the other elements of the geodetic datum are discussed below under "Carrying out datum shifts".

The spheroid is the geometrical model of the earth that was used when the original survey was carried out.

The spheroids that ITRANS knows about, and their code numbers are as follows:-

- 0 Clarke 1866
- 1 Clarke 1880
- 2 Bessel
- 3 New International 1967
- 4 International 1924 (Hayford 1909)
- 5 World Geodetic System 72 (WGS 72)
- 6 Everest
- 7 World Geodetic System 66 (WGS 66)
- 8 Geodetic Reference System 1980 (GRS 1980)
- 9 Airy
- 10 Modified Everest
- 11 Modified Airy
- 12 Walbeck
- 13 Southeast Asia
- 14 Australian National
- 15 Krassovsky
- 16 Hough
- 17 Mercury 1960
- 18 Modified Mercury 1968
- 19 Sphere of radius 6370997 M
- 20 Sphere of radius 6371229.3M
- 21 Clarke 1880 (IGN)
- 22 World Geodetic System 84 (WGS 84)
- 101 User specified spheroid

The spheroid code is held in the map descriptor as an integer; when code 101 is specified two additional REAL*8 (double precision) values are also stored (see below).

The shape of a spheroid is determined by the relationship between the semi-major (equatorial) and semi-minor (polar) axes of the ellipsoid of revolution. This is all that is required if geographical coordinates are required as they are expressed in angular units, and are thus scaleless.

The size of the spheroid is also important, and this is determined by defining the length of one of the axes of the ellipsoid. The shape and size of the spheroid can thus be uniquely defined by specifying the length of the semi-major and semi-minor axes.

When points on the spheroid are related to a plane, i.e. a projection, the units of the projection are the units that the lengths of axes of the spheroid were expressed in. All the spheroids specified above are defined within ITRANS in international metres.

See Appendix A for the details of these spheroids.

When a user specified spheroid is used (code 101) then information about is stored in two REAL*8 (double precision) numbers. The first of these is the length of the semi-major axis; the second contains one of:-

- 1) the length of the semi-minor axis (in the same units), or
- 2) the eccentricity squared

If the second number is 0.0 then a sphere is defined.

NOTES

- 1) The eccentricity squared is defined as:-

$$e^{**2} = (a^{**2} - b^{**2}) / a^{**2}$$

where a is the semi-major axis and b is the semi-minor axis.

- 2) This system works because e^{**2} is less than 1.0, while the semi-minor axis is usually greater than 1.0.

If it is required to define a spheroid where the semi-minor axis is less than 1.0 (e.g. a spheroid with a semi-major axis of 1.0) then the second number must contain the eccentricity squared.

- iv) The projection that the data are on.

The projections that ITRANS can work with (and their codes) are:

- 0 Unset
- 1 Universal Transverse Mercator
- 2 State Plane Coordinates
- 3 Albers Conical Equal Area
- 4 Lambert Conformal Conic
- 5 Mercator
- 6 Polar Stereographic
- 7 Polyconic
- 8 Equidistant Conic
- 9 Transverse Mercator
- 10* Stereographic
- 11* Lambert Azimuthal Equal Area
- 12* Azimuthal Equidistant
- 13* Gnomonic
- 14* Orthographic
- 15* General Vertical Near-Side Perspective
- 16* Sinusoidal
- 17* Equirectangular
- 18* Miller Cylindrical

19* Van der Grinten
20 Oblique Mercator (Hotine)
21* Oblique Mercator (Spherical)
22 Stereographic (UKSL formulae)
23 Cassini
24 Krovak
25 Bonne
26* Mollweide
27* Hammer-Aitoff
28* Winkel III
29* Modified-Stereographic
100 Geographic (ie. Latitude and Longitude)
101 UK national grid (a special case of 9)

NOTES

1) Those projections marked with a '*' are only defined for a sphere, not a spheroid

2) Data can be held in spheroidal, or spherical, coordinates (i.e. Latitude and Longitude) by specifying projection code 100

The projections listed above specify the formulae to be used in the projection, but in any particular case more data will be required, so there is provision in the map descriptor to store up to 15 additional pieces of data as REAL*8 (double precision) numbers.

See Appendix B for data about the individual projections.

v) The units that the data represent.

When the data in the IFF file represents geographical coordinates (projection code 100) then the numbers in the IFF file can represent radians, seconds of arc, degrees of arc or 1/10's of seconds of arc.

As explained in (iii) above, the units of a projection are the units used to define the spheroid, thus for all the spheroids that are predefined in ITRANS the projection units are international metres, and these are the units that are used for the projection calculations. For these spheroids ITRANS will allow data on the projections to be stored and manipulated in feet (see below), but for user defined spheroids then the only units that can be used on a projection are the original projection units.

It is possible that the coordinates in an IFF file are in some other units (e.g. mm on the table). As explained, before doing any **projection** calculations these units must be converted to projection units and this is achieved by doing an empirical transformation.

In this case ITRANS will ask the operator for values of the

control points in projection units before doing the projection calculations.

In summary, the possible units in IFF files, and their codes which are stored as an integer in the map descriptor, are:-

- 0 Unset
- 1 Feet (assuming projection units are metres - see note below)
- 2 Projection units (i.e. units used to specify spheroid). These will be International metres, except for user-specified spheroids.
- 3 Seconds of arc (only valid if projection is 100)
- 4 Degrees of arc (only valid if projection is 100)
- 5 Radians (only valid if projection is 100)
- 101 mm on table (digitising scale)
- 102 mm on map sheet (source scale)
- 103 thousands of an inch on table (digitising scale)
- 104 thousands of an inch on map sheet (source scale)
- 105 user specified units
- 110 1/10 seconds of arc (only valid if projection is 100)

NOTES

1) For projection 2, 1 foot = 0.304800610m (1m = 39.37" exactly) for all other projections, 1 foot = 0.3048m exactly

2) Units 102 can be converted to projection units by dividing by 1000 and multiplying by the scale. On output, projection units may be converted to units 101 (table mm) in the same way.

3) Units 104 can be converted to projection units by dividing by 39370.08 and multiplying by the scale. On output, projection units may be converted to units 103 (table thou) in the same way.

4) User units (code 105) can only be output as a result of an empirical transformation. As they are linear (not angular) units, they cannot be used in conjunction with geographicals (projection code 100).

o Carrying out projection transformations.

To transform data from one projection system into another, ITRANS uses data from the map descriptor for the current IFF file and a map descriptor for the target projection. If the current IFF file has a complete version 2 map descriptor then this will be used, but if it is either not a version 2 type or it is not complete (e.g. the projection code or unit code are set to 0) then the user will be asked to enter the required data by a series of prompts.

NOTE

Within IFF files, coordinate data is stored as X Y coordinate pairs. This is the case even when geographical coordinates are held; ie geographical coordinates are held as Longitude and Latitude.

If data has come from some non Laser-scan source, it is possible that geographical coordinates are in the order Latitude Longitude, and to allow for this possibility there is a flag in the map descriptor called MD2FLP, which can be set by the conversion program. If this flag is set TRUE, and the data is in geographical coordinates, ITRANS asks if the coordinates are in the order Latitude Longitude. If they are, ITRANS converts them to Longitude Latitude.

Note that if ITRANS is being used to patch the map descriptor (/DESCRIPTOR), this procedure will produce a new copy of the file with all the coordinates in the order Longitude Latitude, even in cases when normally the changes take place in situ.

ITRANS always clears MD2FLP.

If the unit code in the current map descriptor represents some non-angular, non-projection unit (i.e. code 101-105) then ITRANS will do an empirical transformation on the data before doing the projection calculation. This will be a "2 point" transformation. If another empirical transformation is required, then this must be done in a preprocessing run of ITRANS. See below for details of other empirical transformations.

The target map descriptor is set up by the operator responding to the relevant prompts.

When prompting for information, ITRANS tries to provide sensible default values either from the existing map descriptor or from the context of the prompt and these can be accepted by simply pressing <CR>.

Help on all but the most trivial of prompts can be obtained by replying with a "?". When run with the /LEARNER qualifier, ITRANS will output the help information automatically before asking the question.

Replying with <CTRL/Z> to the first prompt for a map descriptor will terminate the program; using it in reply to other prompts allows the user to re-enter the last few values.

See Appendix C for allowable number formats to be used when replying to prompts.

o **Carrying out datum shifts.**

When data is surveyed it is captured relative to some geodetic datum. One part of this datum is the spheroid used, which has been discussed above. The other part is the orientation of the spheroid relative to the earth. This is called the origin of the geodetic datum.

The spheroid has 7 degrees of freedom, three defining the position of its centre in relation (say) to the earth's centre of gravity (or any other earth fixed point), three defining rotations about three mutually perpendicular axes to orient it in any desired way and scale to define its size.

In practice, all geodetic datums are defined such that the minor axes are parallel, so there are only 5 freedoms used to define the location of the spheroid in space. These are often called the origin of the geodetic datum and can be expressed as the cartesian coordinates X, Y and Z of the centre of the spheroid, the rotation about the minor (or Z) axis and the size (or scale) of the spheroid. This rotation is often 0.0, and when present is always very small (much less than 1 degree).

ITRANS is only concerned with changes of datum, so only the differences in these values are required i.e. DX, DY, DZ, RZ and SF for the differences in X, Y and Z, the rotation about the Z axis and the difference in scale of the second spheroid with respect to the first. These values are not stored in the map descriptor, but are prompted for when required.

ITRANS can be used to convert data from one geodetic datum to another by using the /DATUM_SHIFT qualifier. It is only possible to carry out this conversion if the input (and output) data are in geographical coordinates (i.e. projection type 100 - Latitude and Longitude).

ITRANS does this conversion by:

1. converting the Latitude, Longitude and Height of the point on the spheroid of the first geodetic datum to cartesian coordinates (X, Y and Z) relative to the spheroid's centre.
2. applying the differences between the datums (DX, DY, DZ, RZ and SF) to these coordinates to convert them to the new datum. The user is prompted for these values.

NOTE

The values of these parameters will generally be obtained from published tables or directly from Geodetic Departments, and care must be taken to ensure that the values so obtained are in the same sense as ITRANS expects them.

1. The scale factor is generally expressed in "parts per million" and this is the form in which ITRANS expects the value to be entered. For example the scale factor between WGS72 and WGS84 is approximately +0.23 ppm. (This implies a scale factor of 1.00000023 in the formulae below.)
2. The rotation about the Z axis assumes a right angled coordinate system, where anticlockwise angles are considered positive, and the solid is considered to rotate relative to the axes. For example, again taking the datum shift from WGS72 to WGS84, the required angle is +0.554".

PLEASE NOTE THAT THERE HAS BEEN CONSIDERABLE CONFUSION ABOUT THE SENSE OF THIS ANGLE IN PUBLISHED ARTICLES. If there is any doubt about it apply Jensen's law - "If the correction is small, don't apply it as you will surely apply it the wrong way".

A standard 3 dimensional conformal transformation (assuming small rotations) is used. Rotations about the X and Y axes are 0.0. This gives:

$$\begin{vmatrix} X \\ Y \\ Z \end{vmatrix}_{\text{new}} = \begin{vmatrix} DX \\ DY \\ DZ \end{vmatrix} + SF \begin{vmatrix} 1 & -RZ & 0 \\ RZ & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} X \\ Y \\ Z \end{vmatrix}_{\text{old}}$$

3. converting the new cartesian X, Y and Z to Latitude, Longitude and Height on the spheroid for the second datum. This will usually be a different spheroid from the original datum, so this is the only mode in which ITRANS allows the user to input the output spheroid.

When converting most map data, it is adequate to assume that the points lie on the spheroid (and thus have a height of 0.0). This is the default mode of operation and as with projection calculations the Z values in ZS or CB entries are just carried through to the output (i.e. only their Latitude and Longitude are altered).

In the general case, however, the height of a point can be significant in the above calculations. This situation can be accommodated in ITRANS by the use of the /HEIGHT_ADJUST qualifier in conjunction with /DATUM_SHIFT. However, this introduces another complication in that heights are measured above the geoid (or mean sea level), while the above calculation uses the height above the spheroid. This geoid-spheroid separation is required for both datums and must be entered by the operator. The entered value for the input datum is added to the height before converting to cartesian coordinates in 1) above, and the value for the target datum is

subtracted from the height after converting to geographicals in 3). It is assumed that the geoid-spheroid separation is constant over the area of the map.

When using the /HEIGHT_ADJUST qualifier, as well as the height of the points affecting the position (latitude and longitude) of the points, the height is also output relative to the new datum.

NOTE

In IFF files this is only the case if the height is held as an attribute of a **point** (i.e. in a ZS or CB entry). When the height is held as the attribute of a **feature** (i.e. as an AC entry) the feature is converted as though it lay on the spheroid, and the value of the AC is unaffected.

WARNING

While the above gives a general mechanism for conversion of spheroid/geodetic datum, it should be noted that "real" survey data often contains errors that render it unsuitable to the above approach. For example, in the United Kingdom the National Grid is based on the OSGB36 datum, which is an inhomogeneous system unsuitable for the above conformal approach.

This mode of ITRANS should be used with care and where possible it is recommended that the results that it produces are verified on known points, before mass data conversions are carried out. This is particularly true when using the /HEIGHT_ADJUST qualifier.

- o **Carrying out empirical transformations.** The situation often arises that data must be transformed from one coordinate space to another, when the mathematical relationships between these spaces are not known, but the coordinates of common points in both systems are available.

ITRANS provides 5 sets of transformation formulae that may be used for these transformations, and the ability to use the coordinates of the common points to determine the values of the coefficients to be used in a particular case. If there are more than the minimum required number of common points, then the best solution is found using the "principle of least squares".

Important Notes.

1. A least squares solution is only valid if the errors that it is required to distribute are *random*. If there are systematic errors or the input data is wrong, then the solution obtained

is of no value.

2. Systematic errors may be allowed for by choosing a suitable set of transformation formulae. It is preferable if this choice can be made based on a knowledge of the distortions, rather than empirically.
3. The control points must be distributed so that they adequately define the relationship between the two coordinate spaces as specified by the formulae chosen. For example, points in a straight line are not suitable for most formulae, except perhaps the orthogonal transformation.
4. These empirical transformations are designed for "local" fitting. They operate best if the two coordinate spaces are of a similar magnitude and any large constant values in the coordinates are first removed (either by a run of ITRANS to move them to the origin offset if they exist in the input space, and by avoiding the use of the /ABS qualifier, to ensure that any constant is put in the origin offset on output).

Ideally the range of the control points should be greater than, say, 10% of the average coordinate values.

When carrying out these empirical transformations on data in an IFF file ITRANS uses one of:

1. the data in the left hand (or "original") side of the CP entry along with one of:
 - o the data in the right hand (or "target") side of the CP entry, or
 - o four pairs of coordinates entered by the user, or
 - o the data in the right hand (or "target") side of the CP entry added to the origin offset and scaled up to projection units (by multiplying by the scale factor and dividing by 1000.0 for mm or 39370.08 for thousands of an inch).

This is only available if the input data is in mm on map sheet (unit code 102) or thousands of an inch on map sheet (unit code 104) and a projection calculation is being carried out. The data is displayed and the user is given the option of accepting these scaled up values or using one of the other options.

2. a file containing the original and target control points, this is defined by the /CONTROL qualifier. A description of the file appears under /CONTROL in the command qualifiers section.

There are five transformations that can be carried out by ITRANS. These are:

1. If the /FOUR qualifier is given then ITRANS uses a least squares fit of the sets of coordinates to compute the transformation

$$\begin{aligned}X &= a1 + a2*x + a3*y + a4*x*y \\Y &= b1 + b2*x + b3*y + b4*x*y\end{aligned}$$

This transformation requires a minimum of 4 control points.

This transformation will force the minimum 4 input points to fit the output points exactly; other points in the file suffer a "saddle" distortion. It is thought to be useful for correcting paper distortion, as when data is digitised from a paper map based on known exact control points such as graticule intersections.

This transformation is called the "Extended four point" transformation in other Laser-Scan programs.

NOTE

This transformation breaks down if a rotation of exactly 45 degrees, a scale change and a shift are the only differences between the input points and the output points. In this case /TWO or /THREE will give the required results.

This is a case of badly distributed control points.

2. The /PROJECTIVE qualifier uses a least squares fit of the sets of coordinates to compute the transformation

$$\begin{aligned}X &= (a1*x + a2*y + a3) / (a7*x + a8*y + 1) \\Y &= (a4*x + a5*y + a6) / (a7*x + a8*y + 1)\end{aligned}$$

This transformation requires a minimum of 4 control points.

This transformation uses a projective algorithm and will force 4 input points to fit the output points exactly. It tends to fit the outside edges of the area, while points in the middle are moved more than with the other formulae.

3. If the /SIX qualifier is given then ITRANS uses a least squares fit of the sets of coordinates to compute the transformation

$$\begin{aligned} X &= a1 + a2*x + a3*y + a4*x*y + a5*x*x + a6*y*y \\ Y &= b1 + b2*x + b3*y + b4*x*y + b5*x*x + b6*y*y \end{aligned}$$

This transformation requires a minimum of 6 control points, and can thus only be used in conjunction with the /CONTROL qualifier.

This transformation is known as the "general second order" or "quadratic" transformation.

4. The /THREE qualifier uses a least squares fit of the sets of coordinates to compute the transformation

$$\begin{aligned} X &= a1 + a2*x + a3*y \\ Y &= b1 + b2*x + b3*y \end{aligned}$$

This transformation requires a minimum of 3 control points.

This transformation is sufficient to correct for rotation, scaling, translation and shearing. This means that shapes may be altered, but in a predictable way (e.g. a square may be transformed into a parallelogram).

This transformation is also known as the "Affine" transformation.

5. The /TWO qualifier uses a least squares fit of the sets of coordinates to compute the transformation

$$\begin{aligned} X &= a1 + a2*x - a3*y \\ Y &= a4 + a3*x + a2*y \end{aligned}$$

This transformation requires a minimum of 2 control points.

This transformation is sufficient to correct for rotation, scaling and translation only. This means that the shapes of features will not be altered and is the transformation that is used by default.

This transformation is called the "Orthogonal" transformation in other Laser-Scan programs. It is also known as the "Helmert" transformation.

When more than the minimum number of control points are supplied to the above formulae, a least squares solution is applied to resolve the inevitable ambiguities. This means that the final coordinates of the control points may be slightly different from the target values. Where the maximum of these discrepancies is greater than 1/1000th of the range of the target control point values then the operator is asked if the transformation is to be accepted.

By default, when doing an empirical transformation, the target coordinates are all reduced by the minimum of their values in x and y, and this value is used as the local origin of the current data. This ensures that all the points will be positive, and it will also,

usually, reduce their absolute size. If it is required to transform a file onto particular values (with no offset) then the /ABSOLUTE qualifier should be used. The use of this qualifier is not recommended, except in exceptional circumstances (e.g. when it is desired to further reduce the size of the coordinates in the output space by choosing the "centre of gravity" of the output coordinates as the origin). It is preferable to use a subsequent run of ITRANS to apply the origin offset to the coordinates in the IFF file.

ITRANS carries out empirical transformations in two circumstances:

1. When a projection calculation is being carried out and the input data is in table, sheet or user units (units code 101-105) then the user is forced to do an empirical transformation before the projection calculation. This will be a two point transformation and in this case the output units cannot be user units.
2. When ITRANS is run with the /TWO, /THREE, /SIX, /FOUR or /PROJECTIVE qualifiers then the corresponding transformation is carried out on the data.

Any scalar output units can be specified; their dimensions are taken from the target control points.

In this case it is not possible to combine the empirical transformation with projection calculations.

NOTE

If a projection calculation is required, and the input data is in table, sheet or user units anything but an orthogonal transformation is required on the data, then two runs of ITRANS will be required. The first to transform the data into projection units (using the appropriate formulae) and the second to do the projection calculation.

See Appendix C for allowable number formats to be used when replying to prompts.

o **Carrying out projection and datum shift calculations on individual points.**

To transform individual points from one projection to another use ITRANS with the /POINT qualifier.

The operator will be asked to enter data about the input projection system and the target projection system.

When this has been achieved, the user is asked to input points in the first system, in the appropriate units and these will be converted to the target system and displayed.

When using the /POINT descriptor on geographical coordinates (either

as input or output) it is possible to specify unit code 200. This allows angular values to be read/written in the format DDD MM SS.SSS.

An ASCII file of X and Y coordinate pairs can be read by giving the command "@filename" to the prompt asking for the coordinates of a point to convert. Missing parts of the file-spec are taken from the default specification SYS\$DISK:ITRANS.DAT.

See Appendix C for allowable number formats to be used when replying to prompts.

Use <CTRL/Z> to exit from the program.

o **Editing the map descriptor.**

Use ITRANS with the /DESCRIPTOR qualifier to edit an existing version 2 map descriptor or to insert one in a file that has an old type.

WARNING

In the first case, the entry is patched into the existing IFF file, so a copy of the file should be taken before use.

In the second case a new version of the file is created.

The operator is prompted to enter the various fields of the map descriptor.

EXAMPLES

\$ ITRANS EXAMPLE EXGEOG<CR>

%ITRANS-W-NOV2MD, map descriptor in 'DUA0:[IFF.IMP.ITRANS]EXAMPLE.IFF;1' is not version 2

Enter projection data for current IFF file

Enter spheroid code (0): 9<CR>
(Airy)

Enter projection code (100): 101<CR>
(UK national grid)

Enter scale of map (0.0): 2500<CR>

Enter unit code (2): <CR>
(metres)

Define local origin as Eastings and Northings (0.0 0.0): 599000 499000<CR>

Enter projection data for target IFF file

Using spheroid : 9 (Airy)

Enter projection code (101): 100<CR>
(Geographic (ie Lat and Long))

Enter scale of map (2500.0): <CR>

Enter unit code (5): ?<CR>

Allowable Units and their codes are:-

0 Unset	1 feet
2 metres	3 seconds of arc
4 degrees of arc	5 radians
101 mm on table	102 mm on map sheet
103 thous (mils) on table	104 thous (mils) on map sheet
105 user units	110 1/10 seconds of arc

Enter unit code (5): 4<CR>
(degrees of arc)

Define local origin as Latitude and Longitude (54.347 1.062): <CR>
Longitude (1.062): <CR>

Input projection system

```
Spheroid      :    9 (Airy)
Projection    : 101 (UK national grid)

Scale        :      2500.0
Units        :    2 (metres)
Local origin :    599000.000 (Eastings)    499000.000 (Northings)
```

```
Output projection system
*****
```

```
Spheroid      :    9 (Airy)
Projection    : 100 (Geographic (ie Lat and Long))

Scale        :      2500.0
Units        :    4 (degrees of arc)
Local origin :    54.347 (Latitude)        1.062 (Longitude)
ELAPSED: 00:00:47.89 CPU: 0:00:05.67 BUFIO: 68 DIRIO: 55 FAULTS: 108
$
```

In this example ITRANS is used to transform the IFF file EXAMPLE (which is held in metres on the UK national grid) into geographicals in a file called EXGEOG.

As EXAMPLE does not have a valid version 2 map descriptor, the user is first asked to enter its projection data. Projection 101 (UK national grid) does not require any additional information.

The user is next asked to enter data for the target projection. Projection 100 (geographicals) also does not require any additional information.

In reply to the prompt "Enter unit code", the user has entered "?", so a list of possible units codes has been output.

As the /OUTPUT qualifier has not been given, the record of the transformation is output to SYS\$OUTPUT

The program finishes by outputting timing statistics. \$STATUS is set to SS\$_NORMAL.

```
$ ITRANS/LOG/OUTPUT EXGEOG EXMERC<CR>
```

```
%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.ITRANS]EXGEOG.IFF;1 opened for read
```

Existing map descriptor:

```
Spheroid      :    9 (Airy)
Projection    : 100 (Geographic  (ie Lat and Long))

Scale        :      2500.0
Units        :    4 (degrees of arc)
Local origin :      54.347 (Latitude)      1.062 (Longitude)
```

```
Enter projection data for target IFF file
*****
```

```
Using spheroid :    9 (Airy)
```

```
Enter projection code (100): 5<CR>
                           (Mercator)
```

The Mercator projection requires the following additional information:-

```
Latitude that has true scale
in DDD MM SS.SS format ( 0 00 00.000N): 49N<CR>
Longitude of the central meridian
in DDD MM SS.SS format ( 0 00 00.000E): 2W<CR>
Eastings of the origin of the projection (0.0): <CR>
Northings of the origin of the projection (0.0): <CR>
```

```
Enter scale of map (2500.0): <CR>
```

```
Enter unit code (2): <CR>
                   (metres)
```

```
Define local origin as Eastings and Northings (224044.019 4733264.099): <CR>
Northings (4733264.099): <CR>
```

```
%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.ITRANS]EXMERC.IFF;1 opened for write
ELAPSED: 00:00:40.70 CPU: 0:00:04.43 BUFIO: 45 DIRIO: 69 FAULTS: 280
$
```

In this example ITRANS is used to transform the IFF file EXGEOG (created in the last example) into a mercator projection in a file called EXMERC.IFF.

As the /LOG qualifier has been given, a message is given whenever files are opened.

As the input file has a valid map descriptor the user is only asked to enter data for the target projection. Sensible values are offered as defaults.

The Mercator projection requires four additional pieces of information. As the input projection is different from the target system, defaults of 0.0 are offered.

As the target system is a projection, the default units are projection units.

The local origin of the input system is transformed into the target system and is offered as a default.

\$ ITRANS/POINT<CR>

Enter projection data for source system

Enter spheroid code (0): **<CTRL/Z>**

%ITRANS-W-EOF, program terminated prematurely by '^Z'

ELAPSED: 00:00:03.19 CPU: 0:00:00.34 BUFIO: 6 DIRIO: 0 FAULTS: 135

In this example ITRANS is to be used to transform individual points.

The run is terminated prematurely by entering <CTRL/Z> to the first prompt.

\$ ITRANS/POINT/OUTPUT=EXPOINT<CR>

Enter projection data for source system

Enter spheroid code (0): **9<CR>**
(Airy)

Enter projection code (100): **101<CR>**
(UK national grid)

Enter unit code (2): **<CR>**
(metres)

Enter projection data for target system

Using spheroid : 9 (Airy)

Enter projection code (101): **5<CR>**
(Mercator)

The Mercator projection requires the following additional information:-

Latitude that has true scale
in DDD MM SS.SS format (0 00 00.000N): **49N<CR>**
Longitude of the central meridian
in DDD MM SS.SS format (0 00 00.000E): **2W<CR>**

Eastings of the origin of the projection (0.0): <CR>
Northings of the origin of the projection (0.0): <CR>

Enter unit code (2): <CR>
(metres)

Eastings and Northings: 599000 500000<CR>
Transformed point 224092.895 4734388.290

Eastings and Northings: 599000 499000<CR>
Transformed point 224044.019 4733264.099

Eastings and Northings: 600000 499000<CR>
Transformed point 225168.075 4733215.114

Eastings and Northings: 600000 500000<CR>
Transformed point 225217.195 4734339.279

Eastings and Northings: <CTRL/Z>
ELAPSED: 00:00:45.68 CPU: 0:00:00.84 BUFIO: 50 DIRIO: 15 FAULTS: 145
\$

In this example ITRANS is used to convert individual points from UK National Grid to a Mercator projection.

As the /OUTPUT qualifier has been given, the file EXPOINT.LIS has been created. This contains the following:-

===== ITRANS =====

ITRANS invoked by TIM using terminal LTA75: at 9-NOV-1986 08:52:29.05

Command line:
ITRANS/POINT/OUTPUT=EXPOINT

Input projection system

Spheroid : 9 (Airy)
Projection : 101 (UK national grid)

Units : 2 (metres)

Output projection system

Spheroid : 9 (Airy)
Projection : 5 (Mercator)


```
Latitude of true scale      : 49 00 00.000N
Longitude of the central meridian : 2 00 00.000W
Eastings of origin         : 0.000
Northings of origin        : 0.000
```

Units : 2 (metres)

Transformed points

Input Point		Output Point	
Eastings	Northings	Eastings	Northings
599000.000	500000.000	224092.895	4734388.290
599000.000	499000.000	224044.019	4733264.099
600000.000	499000.000	225168.075	4733215.114
600000.000	500000.000	225217.195	4734339.279

\$ ITRANS/DESCR/OUTPUT=EXDESC/LOG EXMERC<CR>

%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.ITRANS]EXMERC.IFF;1 opened for read

Existing map descriptor:

Spheroid : 9 (Airy)
Projection : 5 (Mercator)

```
Latitude of true scale      : 49 00 00.000N
Longitude of the central meridian : 2 00 00.000W
Eastings of origin         : 0.000
Northings of origin        : 0.000
```

Scale : 2500.0
Units : 2 (metres)
Local origin : 224044.019 (Eastings) 4733264.099 (Northings)

Enter projection data for current IFF file

Enter spheroid code (9): <CR>
(Airy)

Enter projection code (5): <CR>
(Mercator)

The Mercator projection requires the following
additional information:-

```
Latitude that has true scale
in DDD MM SS.SS format ( 49 00 00.000N): <CR>
Longitude of the central meridian
in DDD MM SS.SS format ( 2 00 00.000W): <CR>
```

Eastings of the origin of the projection (0.0): <CR>
Northings of the origin of the projection (0.0): <CR>

Enter scale of map (2500.0): <CR>

Enter unit code (2): **105**<CR>
(user units)

Define local origin as Eastings and Northings (224044.019 4733264.099): **0**<CR>
Northings (4733264.099): **0**<CR>

%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.ITRANS]EXMERC.IFF;1 opened for update
%ITRANS-I-MDPTCHED, MD entry patched into 'DUA0:[IFF.IMP.ITRANS]EXMERC.IFF;1'
ELAPSED: 00:00:26.64 CPU: 0:00:01.15 BUFIO: 52 DIRIO: 25 FAULTS: 165

In this example ITRANS is used to alter the unit code in the map descriptor of the file EXMERC.IFF without doing any transformation.

As the file has a valid version 2 map descriptor, the MD entry is patched in situ.

\$ ITRANS/OUTPUT/LOG EXMERC EXAMPLE1<CR>

%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.ITRANS]EXMERC.IFF;1 opened for read

Existing map descriptor:

Spheroid : 9 (Airy)
Projection : 5 (Mercator)

Latitude of true scale	:	49 00 00.000N
Longitude of the central meridian	:	2 00 00.000W
Eastings of origin	:	0.000
Northings of origin	:	0.000

Scale : 2500.0
Units : 105 (user units)
Local origin : 0.000 (Eastings) 0.000 (Northings)

Data are not in projection units

Existing control points are:

48.876	1124.191
0.000	0.000
1124.056	-48.985
1173.176	1075.180

Do you wish to use these control points [N]? **N**<CR>

Enter control point values in metres on Mercator projection

Enter coordinates of NW corner: **224092.90 4734388.29**

Enter coordinates of SW corner: **224044.02 4733264.10**

Enter coordinates of SE corner: **225168.08 4733215.11**

Enter coordinates of NE corner: **225217.20 4734339.28**

Enter projection data for target IFF file

Using spheroid : 9 (Airy)

Enter projection code (5): **101<CR>**
(UK national grid)

Enter scale of map (2500.0): **<CR>**

Enter unit code (2): **<CR>**
(metres)

Define local origin as Eastings and Northings (599001.892 498956.5): **599000
499000<CR>**

%LSLLIB-I-IFFOPENED, DUA0:[IFF.IMP.ITRANS]EXAMPLE1.IFF;1 opened for write
ELAPSED: 00:02:07.74 CPU: 0:00:08.42 BUFIO: 52 DIRIO: 72 FAULTS: 272

In this example ITRANS is used to convert EXMERC back into UK national grid.

As the units in the input file are not in projection units, the user is required to set up data for a two point transformation, before the projection calculation can be carried out.

The user chooses not to use the values in the right hand side of the existing CP entry, but enters full Mercator coordinates for each control point. As the /ABS qualifier has not been given, these values are reduced by their minimum x and y values to produce smaller absolute values, and this minimum x and y is used as the local origin.

This point is then transformed and offered as the default local origin for the target system, but is not accepted.

As the /OUTPUT qualifier has been given, the file ITRANS.LIS has been created. Note that as a 2 point empirical transformation has been used, there are small discrepancies between the target values of the control points and their final values.

===== ITRANS =====

ITRANS invoked by TIM using terminal LTA75: at 9-DEC-1986 09:12:29.05

Command line:
ITRANS/OUTPUT/LOG EXMERC EXAMPLE1

Input projection system

Spheroid : 9 (Airy)
 Projection : 5 (Mercator)

Latitude of true scale : 49 00 00.000N
 Longitude of the central meridian : 2 00 00.000W
 Eastings of origin : 0.000
 Northings of origin : 0.000

Scale : 2500.0
 Units : 105 (user units)
 Local origin : 0.000 (Eastings) 0.000 (Northings)

Control points (from CP entry) used for 2 point transformation

Input		Target		Final	
48.876	1124.191	48.880	1173.180	48.879	1173.182
0.000	0.000	0.000	48.990	0.001	48.988
1124.056	-48.985	1124.060	0.000	1124.059	0.002
1173.176	1075.180	1173.180	1124.170	1173.181	1124.168

Default origin calculated as: 224044.020 4733215.110

Output projection system

Spheroid : 9 (Airy)
 Projection : 101 (UK national grid)

Scale : 2500.0
 Units : 2 (metres)
 Local origin : 599000.000 (Eastings) 499000.000 (Northings)

Input file : DUA0:[IFF.IMP.ITRANS]EXMERC.IFF;1

Output file: DUA0:[IFF.IMP.ITRANS]EXAMPLE1.IFF;1

\$ **itrans/point/datum/height<CR>**

Enter projection data for source system

Enter spheroid code (0): **5<CR>**
 (World Geodetic System 72 (WGS 72))
 Define Geoidal - spheroidal separation (0.0): **<CR>**

Enter projection code (100): **<CR>**
 (Geographic (ie Lat and Long))

Enter unit code (200): **<CR>**
(dd mm ss.ss format)

Define shift in X from original datum to target datum (0.0): **0.0<CR>**
Define shift in Y from original datum to target datum (0.0): **0.0<CR>**
Define shift in Z from original datum to target datum (0.0): **4.5<CR>**
Define rotation about Z axis of the target datum from the original
datum (in DDD MM SS.SS format) (+ 0 00 00.000) : **00 00 00.554<CR>**
Define scale difference between datums (in ppm) (0.0) : **0.2263<CR>**

Enter projection data for target system

Enter spheroid code (5): **22<CR>**
(World Geodetic System 84 (WGS 84))
Define Geoidal - spheroidal separation (0.0): **<CR>**

Enter projection code (100): **<CR>**
(Geographic (ie Lat and Long))

Enter unit code (200): **<CR>**
(dd mm ss.ss format)

Input projection system

Spheroid : 5 (World Geodetic System 72 (WGS 72))
(with Geoid - Spheroid separation of 0.0)
Projection : 100 (Geographic (ie Lat and Long))
Units : 200 (dd mm ss.ss format)

Parameters used for datum conversion

DX = 0.00
DY = 0.00
DZ = 4.50
RZ = + 0 00 00.554
SF = 0.23 (parts per million)

Output projection system

Spheroid : 22 (World Geodetic System 84 (WGS 84))
(with Geoid - Spheroid separation of 0.0)
Projection : 100 (Geographic (ie Lat and Long))
Units : 200 (dd mm ss.ss format)

Latitude, Longitude and Height: **39 13 26.5782n 98 32 32.2870w 570.88<CR>**
Transformed point 39 13 26.698N 98 32 31.733W 573.25

Latitude, Longitude and Height: **Z**
ELAPSED: 0 00:02:07.87 CPU: 0:00:00.18 BUFIO: 66 DIRIO: 0 FAULTS: 147

\$

In this example ITRANS is used to do an accurate geodetic datum shift between WGS72 and WGS84.

As the /HEIGHT_ADJUST qualifier has been given, ITRANS prompts for the spheroid geoid separation. In this case the heights of the points are spheroidal heights, so the default value of 0.0 is used.

Between defining the input and output systems, ITRANS prompts for the shifts to be applied. Note that the rotation is given in degrees minutes and seconds format. The scale change is given as parts per million.

As the /HEIGHT_ADJUST qualifier has been given, ITRANS requires latitude, longitude and height for each point to convert.

\$ ITRANS/FOUR/CONTROL=EXAMPLE RWR_GRID RWR_OUT<CR>

Existing map descriptor:

Spheroid : 9 (Airy)
Projection : 101 (UK national grid)

Scale : 2500.0
Units : 2 (metres)
Local origin : 499000.000 (Eastings) 599000.000 (Northings)
%ITRANS_W_ERRCPTS, residuals in X and Y of 0.59 1.81 metres
Do you wish to continue with these residuals [N]? Y

Default origin of current IFF file has been computed
as: Easting 499000.000 Northing 599000.0

Enter projection data for target IFF file

Using spheroid : 9 (Airy)

Using projection: 101 (UK national grid)

Enter unit code (105): <CR>
(user units)

Define local origin as Eastings Northings (499000.0 599000.0): <CR>
(599000.0) <CR>

Input projection system

Spheroid : 9 (Airy)
 Projection : 101 (UK national grid)

Scale : 2500.0
 Units : 2 (metres)
 Local origin : 499000.000 (Eastings) 599000.000 (Northings)

Control points (from command file) used for 4 point transformation

Input		Target		Final	
499000.000	600000.000	499000.000	600000.000	498999.688	599999.375
499000.000	599000.000	499000.000	599000.000	498999.826	598999.653
500000.000	599000.000	500000.000	599000.000	499999.688	598999.375
500000.000	600000.000	500002.500	600005.000	500002.049	600004.097
499250.000	599500.000	499250.000	599500.000	499250.035	599500.069
499750.000	599500.000	499750.000	599500.000	499750.590	599501.181
499500.000	599250.000	499500.000	599250.000	499500.035	599250.069
499500.000	599750.000	499500.000	599750.000	499500.590	599751.181

Default origin calculated as: 499000.000 599000.000

Output projection system

Spheroid : 9 (Airy)
 Projection : 101 (UK national grid)

Scale : 2500.0
 Units : 105 (user units)
 Local origin : 499000.000 (Eastings) 599000.000 (Northings)
 ELAPSED: 0 00:00:39.71 CPU: 0:00:01.48 BUFIO: 67 DIRIO: 104 FAULTS: 218

In this example the /CONTROL qualifier has been given in conjunction with the /FOUR qualifier. This has the effect of reading the control points from a file, in this case called EXAMPLE.DAT (shown below), which is held in the current directory and performing a four point empirical transformation using these control points.

```
!
! This is an example control point file
!
! The coordinates are absolute; ie they include the local origin
!
499000.000 600000.000 499000.000 600000.000
499000.000 599000.000 499000.000 599000.000
500000.000 599000.000 500000.000 599000.000
500000.000 600000.000 500002.500 600005.000 ! this is only point in error
499250.000 599500.000 499250.000 599500.000
499750.000 599500.000 499750.000 599500.000
499500.000 599250.000 499500.000 599250.000
```

499500.000 599750.000 499500.000 599750.000
;
; and this is the end of it

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action from the user.

MDCOPIED, new version of IFF file created

Explanation: When using ITRANS with the /DESCRIPTOR qualifier, the specified IFF file did not have a version 2 map descriptor entry, so a new IFF file has been created which has a MD entry of the correct length.

User action: Delete the original IFF file on completion of this run of ITRANS.

MDPTCHED, MD entry patched into IFF file

Explanation: When using ITRANS with the /DESCRIPTOR qualifier, the specified IFF file had a version 2 map descriptor. The new values have been patched into this, and the original values are lost.

User action: None.

NEWRAD, using sphere of equal area to spheroid

Explanation: A spheroid has been specified, while working on a projection that is only defined for a sphere. Under these circumstances, ITRANS uses a sphere which has the same area as the specified spheroid.

User action: If this is not the desired action, on completion of the run of ITRANS, delete the transformed IFF file and re-run ITRANS, specifying a sphere rather than a spheroid.

OUTFILEDEF, using default output file <'file-spec'>

Explanation: When the /LOG qualifier has been given, this message is output if no output IFF file has been specified.

User action: None.

REENTER, reenter input values

Explanation: <CTRL/Z> has been entered in response to a prompt or an internal arithmetic error has occurred while initialising a projection or empirical transformation.

User action: Reenter the last few values.

MESSAGES (WARNING)

These messages are output when an error has occurred that can be corrected immediately by the user.

BADCPTS, badly conditioned control points

Explanation: ITRANS has not been able to solve the empirical transformation using the supplied control points, perhaps because they have been given in the wrong order, or they do not adequately control the relationship between the input and output coordinate spaces.

User action: Reenter the target values for the control points.

BADHEMI, two angles are both latitude or both longitude

Explanation: User has input two longitude or two latitude values, rather than one of each, when transforming individual geographical points.

User action: Enter two angles with correct hemisphere suffix.

BADPRJCDW, invalid projection code

Explanation: User has entered a number that does not correspond to a valid projection code.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

BADMODSTOPT, 'integer' is invalid modified-stereographic option code

Explanation: User has entered a number that does not correspond to a valid projection code.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

BADPRJSPH, inconsistent projection and spheroid

Explanation: Certain projections are only defined for specific spheroids. The projection code that has been specified is not valid with the specified spheroid as it is one of these.

User action: Select another projection code or, if the spheroid is wrong, exit from ITRANS and convert the data onto the correct spheroid and use ITRANS with the /DESCRIPTOR qualifier to alter the spheroid code in the map descriptor.

BADPRJTYP, cannot mix spherical and spheroidal projections

Explanation: Some of the projections that ITRANS operates on are only defined for a sphere, not a spheroid. It is not sensible to convert from one type to another.
If a sphere has been specified, (either spheroid code 19 or 20 or a user

specified spheroid (code 101) defined as a sphere), this error will not occur.

User action: Select another (spheroidal) projection, or exit from ITRANS, convert the data from the spheroid onto a sphere and then use ITRANS with the /DESCRIPTOR qualifier to update the map descriptor.

BADSPHCDW, invalid spheroid code

Explanation: User has entered a number that does not correspond to a valid spheroid code.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

BADSPSZON, invalid SPS zone

Explanation: User has entered a number that does not correspond to a valid state plane projection system zone.

User action: Enter '?' to obtain list of valid zones, and enter one of them.

BADUNTCDW, invalid unit code

Explanation: User has entered a number that does not correspond to a valid unit code.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

BADUTMZON, invalid UTM zone

Explanation: User has entered a number that does not correspond to a valid UTM zone.

User action: Enter '?' to obtain data about UTM zones and enter another value, or if the UTM zone is unknown enter zone 0. This will make ITRANS prompt further for the latitude and longitude of a point in the area of interest.

DSERR, arithmetic error while shifting datum

Explanation: An arithmetic error has occurred while converting from one geodetic datum to another. This is probably caused by the operator entering the datum shift parameters incorrectly.

User action: Check the values of the datum shift parameters, and rerun the program

EOF, program terminated prematurely by '^Z'

Explanation: The user has entered <CTRL/Z> to the first prompt when defining a projection.

User action: None.

ERRCPTS, residuals in X and Y of 'X real' and 'Y real'

Explanation: When there are more than the minimum number of control points required for a particular transformation, the ambiguity is resolved by the use of a least squares solution. This leaves residuals at the control points, and when the maximum residual in X or the maximum residual in Y is greater than 1/1000th of the range in X and Y then this message is output.

User action: The user can either accept these residuals and continue, or choose to reenter the target values of the control points.

ILLEGANGVAL, impossible value for component of angle

Explanation: An angle has been input with a value of one of its components that is impossible (e.g. minutes greater than 60)

User action: Reenter the angle with correct value.

ILLEGDECPT, only last component in angle may have decimal point

Explanation: An angle has been input where more than one of its components has a decimal point.

User action: Reenter the angle with correct value.

ILLEGINPUT, illegal input units for datum shift

Explanation: When using the /DATUM_SHIFT qualifier, the input units must be angular, not linear. This message should not occur with files that have been correctly processed.

User action: Use ITRANS/DESCRIPTOR to correct the unit code

ILLEGLAT, illegal latitude value

Explanation: A latitude value has been entered with an absolute value greater than 90 degrees.

User action: Reenter the angle with correct value.

ILLEGLONG, illegal longitude value

Explanation: A longitude value has been entered with an absolute value greater than 180 degrees.

User action: Reenter the angle with correct value.

ILLEGOUTPUT, illegal output units for datum shift

Explanation: When using the /DATUM_SHIFT qualifier, the output units must be angular, not linear. This message should not occur.

User action: Report problem to Laser-Scan

IMPOSUNT, no FEET with user-specified spheroid

Explanation: Projection units for all the spheroids defined within ITRANS are metres, and there is a known relationship between them and feet. When using a user specified spheroid, the projection units are unknown, and it is not possible to use FEET

User action: Use an internally defined spheroid, or specify a user defined spheroid in feet, and specify unit code 2 (projection units).

INVALANGUNT, no 'ddd mm ss.ss' format when transforming files

Explanation: Unit code 200 is not possible when dealing with files. This message should not occur.

User action: Exit from ITRANS and patch the map descriptor entry, using ITRANS with the /DESCRIPTOR qualifier, to contain a valid unit code.

INVALSCL, scale of 1:'scale' is invalid when dealing with 'units'

Explanation: Using table or sheet units requires a sensible scale.

User action: Enter valid value for scale factor (a value greater than 0)

INVALTABUNT, no table units when transforming points

Explanation: It is not possible to specify unit codes 101 - 105 when dealing with points.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

INVALUNT, inconsistent unit on projection

Explanation: Angular units can only be used when dealing with geographical coordinates, and scalar units can only be used when dealing with a projection.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

INVALUSEUNT, no user units without an empirical transformation

Explanation: The size of user units (unit code 105) are defined by the values of the control points that are entered when doing an empirical transformation. This is the only way that data can be converted into user units.

User action: Complete any required projection calculations by selecting an acceptable target unit code and then re-run ITRANS with the /TWO, /THREE, /FOUR or /PROJECTIVE qualifier.

NOTGEOGS, need geographical coordinates for datum shift

Explanation: The user has entered a projection code other than 100 while using the /DATUM_SHIFT qualifier. Geodetic datum shifts can only be carried out on Geographical coordinates (Latitude and Longitude)

User action: Enter 100 to the "Projection :" prompt

NOHEMI, no hemisphere allowed with angle in this context

Explanation: The angle required at this point must be in the form sDDD MM SS.SS, without a hemisphere suffix

User action: Reenter the angle in the correct format

NOTUNSET, cannot specify UNSET code

Explanation: It is not possible to do projection calculations if the projection code or unit code is set to 0.

User action: Enter '?' to obtain list of valid codes, and enter one of them.

NOV2MD, map descriptor is not version 2

Explanation: The input IFF file does not have a version 2 map descriptor.

User action: The user will be asked to specify a map descriptor for the current IFF file.

SGNANDHEMI, sign and hemisphere present in angle

Explanation: An angle has been input that is both preceded by a sign and succeeded by a hemisphere code.

User action: Reenter a valid angle.

UNSETMD, map descriptor is unset

Explanation: The map descriptor in the input IFF file is unset.

User action: The user will be asked to specify a map descriptor for the current IFF file.

UNEXPEOL, unexpected end of line

Explanation: An incomplete line has been input.

This error should not occur.

User action: Reenter complete line

UNEXPCH, unexpected character 'character' found

Explanation: An invalid character 'character' has occurred in an input line.

User action: Reenter a valid value.

UNSETPRJ, projection code unset

Explanation: The projection code field in the map descriptor in the input

IFF file is unset.

User action: The user will be asked to specify a map descriptor for the current IFF file (using the fields that are set as defaults).

UNSETUNT, unit code unset

Explanation: The unit code field in the map descriptor in the input IFF file is unset.

User action: The user will be asked to specify a map descriptor for the current IFF file (using the fields that are set as defaults).

WGCTPERR, projection error

Explanation: While using ITRANS with the /POINT qualifier, an error has occurred while carrying out the projection transformation. It will be followed by an explanation of the error.

User action: Reenter a valid point to transform.

WRNGHEM, wrong hemisphere

Explanation: A latitude value has been input when a longitude was expected or vice versa.

User action: Reenter the angle with the correct hemisphere letter.

2MANYLINES Too many lines of points in control file, truncating.

Explanation: The number of lines of control points in the file specified with the /CONTROL qualifier exceeds the maximum allowed, (see above), all lines after the maximum allowed will be ignored and ITRANS will continue as normal

User action: The user may decided to edit the file and rerun ITRANS.

MESSAGES (FATAL)

After these messages have been output, ITRANS will be aborted. This may leave an invalid IFF file that should be deleted.

BADPRJCDS, invalid projection code

Explanation: ITRANS tries to output an invalid projection code. This should not happen.

User action: Correct the projection code field of the map descriptor with ITRANS and the /DESCRIPTOR qualifier, and try again.

BADSPHCDS, invalid spheroid code

Explanation: ITRANS tries to output an invalid spheroid code. This should not happen.

User action: Correct the spheroid code field of the map descriptor with ITRANS and the /DESCRIPTOR qualifier, and try again.

BADUNTCDS, invalid unit code

Explanation: ITRANS tries to output an invalid unit code. This should not happen.

User action: Correct the unit code field of the map descriptor with ITRANS and the /DESCRIPTOR qualifier, and try again.

CTRLFILERR, error reading the control file

Explanation: An error has been encountered trying to read from the control file.

This message will be followed by a description of the error.

User action: Check the control file, according to the given message.

ERROPNCTRL, error opening the control file 'filename'

Explanation: An error has been encountered trying to open the control file.

This message will be followed by a description of the error.

User action: Check that the control file exists, according to the given message, and rerun ITRANS with a valid control point file.

FGCTPERR, projection error

Explanation: ITRANS has tried to carry out an illegal projection transformation on a point in the IFF file. This may be a control point. This message will be followed by a description of the error.

User action: Correct the value of the point in the IFF file and re-run

ITRANS, or do not carry out this projection calculation.

INCONSPH, inconsistent input and output spheroids. See documentation

Explanation: ITRANS cannot be used to change spheroids.

User action: use ITRANS/DATUM_SHIFT to change geodetic datum

NOCP, no control point entry found in IFF file

Explanation: The input IFF file does not have a valid format and the control point entry is missing.

User action: Correct the error in the data and re-run ITRANS

NOTGEOGS, need geographical coordinates for datum shift

Explanation: The input IFF file does not have projection code 100, but the /DATUM_SHIFT qualifier has been given.

User action: Convert the IFF file to geographicals using ITRANS in its default mode, then change the datum

NOMD, no map descriptor

Explanation: The input IFF file does not have a valid format and the map descriptor entry is missing.

User action: Correct the error in the data and re-run ITRANS

NOTPLANE, no empirical transformation on geographical coordinates

Explanation: It is not possible to do a empirical transformation on geographical coordinates.

User action: Transform the geographical data onto a projection before trying to do an empirical transformation

PREMEOL, premature end of line, four control points expected

Explanation: A line in the control file appears to contain one, two or three control points, where four was expected.

User action: Amend the control file so that each line containing control points has four points - originalX originalY targetX targetY

UNEXPAT, "@" not allowed in input file

Explanation: A file being used to input coordinates to ITRANS in /POINT contains a "@" sign. It is not possible to have nested files of coordinates.

User action: Correct the input file, and rerun ITRANS

WRNGCPTS, LHS and RHS of control point entry not equal

Explanation: The control point entry in the input IFF file has different left hand side and right hand side values.

User action: Correct the CP entry in the IFF file, or run ITRANS with the /TWO, /THREE, /FOUR or /PROJECTIVE qualifier to do an empirical transformation on the data using the existing right hand side control point values.

2FEWPTS, not enough sets of control points for specified transformation

Explanation: The number of lines of control points in the control file is less than the minimum required, (see above).

User action: Add more lines of control points to the control file until the minimum required has been reached.

MESSAGES (OTHER)

In addition to the above messages which are generated by ITRANS itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 21

MODULE IWINDOW

MODULE IWINDOW

REPLACES DAMP modules ICLIP and OSCLIP

FUNCTION

IWINDOW clips IFF map data to a rectangular boundary or "window".

FORMAT

\$ IWINDOW input-file-spec output-file-spec

Command qualifiers**Defaults**

/ABSOLUTE	/NOABSOLUTE
/AREA[=keyword=integer[,...]]	/NOAREA
/CONTROL_POINTS	None.
/[NO]DEBUG	/NODEBUG
/EAST=real	None.
/[NO]FC_IGNORE=integer[,...]	/NOFC_IGNORE
/FRT=file-spec	/NOFRT
/FSN_BY_LAYER	/NOFSN_BY_LAYER
/[NO]LAYER_IGNORE=integer[,...]	/NOLAYER_IGNORE
/LITES2	/NOLITES2
/[NO]LOG	/NOLOG
/OS	See text.
/MAP_HEADER	None.
/NORTH=real	None.
/OFFSET	None.
/ORIGIN	SW control coordinates unchanged.
/OUTPUT=file-spec	Output to SYS\$OUTPUT.
/[NO]RESET_CP	/NORESET_CP
/[NO]RETAIN_FSN	/NORETAIN_FSN
/SOUTH=real	None.
/WEST=real	None.
/VOID	None.

Ordnance Survey mode

/CLIP_GRID	See text
/KEEP_TEXT	See text.

RESTRICTIONS

- o /ABSOLUTE is invalid with /CONTROL_POINTS or /MAP_HEADER
- o /AREA with no keyword requires /FRT
- o /AREA and /VOID are mutually exclusive
- o /AREA keywords LAYER and FC are mutually exclusive
- o /CLIP_GRID and /KEEP_TEXT require /OS
- o /CONTROL_POINTS and /MAP_HEADER are mutually exclusive
- o /OUTPUT requires /LOG
- o /OFFSET requires either /CONTROL_POINTS or /MAP_HEADER or /OS
- o /RETAIN_FSN and /FSN_BY_LAYER are mutually exclusive
- o /RETAIN_FSN and /FSN_BY_LAYER are invalid with /OS
- o /VOID and /ORIGIN are mutually exclusive

PROMPTS

From: input-file-spec
To: output-file-spec

PARAMETERS**input-file-spec**

- specifies the IFF file which is to be clipped. Any part of the file-spec which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

output-file-spec

- specifies the IFF file which is to be created to contain the data clipped out of the input file. Any part of the file name which is not explicitly given will be taken from the parsed input specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS**/ABSOLUTE**

- Enables the user to specify clipping bounds in absolute (whole world) coordinates. By default clipping bounds are relative to the sheet local origin defined by the origin offset in a type 2 IFF MD (Map Descriptor) entry.

/AREA**/AREA=FC=integer[,integer...]****/AREA=LAYER=integer[,integer...]**

- Area features are identified to IWINDOW either using the keyword arguments to the /AREA qualifier, or using the graphical type information from a FRT file specified using the /FRT qualifier. Area feature identifications from /AREA keyword arguments take precedence over those from a FRT file. Area features which intersect with a clipping bound are closed with a straight line segment along the clipping bound. A single keyword argument to /AREA is obligatory unless used in conjunction with the /FRT qualifier. /AREA=FC and /AREA=LAYER are mutually exclusive. Clipping suppression using /FC_IGNORE and /LAYER_IGNORE overrides /AREA=FC and /AREA=LAYER specifications.

/CONTROL_POINTS

- derive the clipping bounds from the CP (Control Point) entry.

/DEBUG**/NODEBUG**

- reserved for Laser-Scan use. /DEBUG provides voluminous output relating to IFF entries found. It is intended as a debugging aid for use by LSL personnel only.

/EAST=real

- specifies the eastern bound of the clipping window in IFF units.

/FC_IGNORE=integer[,...]

- turn off clipping for the specified feature codes. The /FC_IGNORE qualifier will accept single, multiple or ranges of feature code arguments. The maximum number of feature codes which may be specified is 1024.

/FRT=file-spec

- specifies the FRT (Feature Representation Table) file used to identify symbol strings, and area features when the /AREA qualifier is specified. The FRT file-specification is required. Any part of the file-spec not supplied will be taken from the default specification 'LSL\$FRT:.FRT'.

/FSN_BY_LAYER

- Specification of the /FSN_BY_LAYER qualifier causes IWINDOW to keep a record of the maximum FSN per layer. Individual layer maximum FSN + 1 is then used for reentrant features within that particular layer. By default IWINDOW allocates uses the whole file maximum FSN + 1 for reentrant features.

/LAYER_IGNORE=integer[,...]
/NOLAYER_IGNORE

- turn off clipping for the specified layers. /LAYER_IGNORE will accept single, multiple or ranges of layer arguments. The maximum number of layers which may be specified is 1024.

/LITES2

- If /LITES2 is specified IWINDOW creates a LITES2 command file which is used to guide the user to re-entrant features within the LITES2 editor. The LITES2 command file specification is derived from the output filename but with the substitution of the logical name LSL\$LITES2CMD: and the extension '.LCM' Thus if the /LITES2 qualifier is specified for use with an output file called LSL\$IF:TST.IFF then the LITES2 command file will be LSL\$LITES2CMD:TST.LCM

If there were no re-entrant features an empty guidance file, containing just a header and message stating there were no re-entrant features, can still be produced by setting the logical name LSL\$EMPTY_LITES2_GUIDANCE to 1. If this logical name is set to 0, no empty guidance files will be produced, (the normal behaviour). If the /LITES qualifier is given, this logical name must be set to 0 or 1.

If the map is totally within the clipping boundary, a suitable message is given and there will be no output file or guidance file produced, regardless of the logical name setting.

/LOG
/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT:. Supplementary messages are generated when an IFF file is successfully opened and a reassuring message output each time a new IFF layer is found in the input IFF file.

IWINDOW also writes out the actual bounds being clipped to and a summary of the options chosen. For each layer, a note is made of each feature changed, and a summary is given of how many features were clipped, deleted and created. A similar summary is given at the end

of clipping for the whole map. If no clipping was necessary then this is indicated in the summary information.

If /OUTPUT is specified in conjunction with /LOG then the clipping bound information and summary information is sent to the specified file and will not be reflected at the terminal.

/MAP_HEADER

- clip to the window bounds held in the MH (Map Header) entry, which must be either an OS or MCE map header depending on whether or not the program is running in OS mode.

/NORTH=real

- specifies the northern bound of the clipping window in IFF units.

/OFFSET=real

- specifies the boundary offset in IFF units. For use with /CONTROL_POINTS and /MAP_HEADER qualifiers. This is the distance outside the edge of the map to which all features will be clipped. The default is zero, clip to the true map edge.

/ORIGIN

- translates coordinates so that the SW corner of the window is at (0,0). The map origin in the OS or MCE map header and type 2 map descriptor is changed in accordance with this. If the OS map header is type 3 or 4, the origin is updated in the header using the position and size specified in the OS map header translation table, LSL\$OS_MH_TABLE. This table is described in the DATA PREPARATION section of the IFFOSTF chapter in the "Convert User Guide". Since a type 4 (CITF) Map Header contains no basic grid interval it is calculated from the scale so that maps at scales 1:1250, 1:2500 and 1:10000 have basic grid intervals of 50m, 100m, and 400m respectively. These are the only scales intended for use with CITF data.

/OS

- forces IWINDOW into Ordnance Survey (UK) mode. Command qualifiers preceding the /OS qualifier are ignored by the Command Line Interpreter.

/OUTPUT[=file-spec]

- for use with /LOG. This enables the user to send to file the clipping bound and summary information produced by the /LOG qualifier. Additional information sent to the /OUTPUT file includes details of the current process and the IFF HHistory information if this was present in the input file. The output file specification defaults to SYS\$DISK:IWINDOW.LIS.

/RESET_CP
/NORESET_CP

- if /RESET_CP is specified the control point values in the output IFF file will be reset to the clipping bound coordinates. By default the CP values in the input file are copied to the output file.

/RETAIN_FSN
/NORETAIN_FSN

- when splitting a reentrant feature, the new feature created should have the same FSN as its 'parent'.

/SOUTH=real

- specifies the southern bound of the clipping window in IFF units.

/VOID

- clip a rectangular void (or "hole") inside the map. The void boundary is defined in the same way as a clipping window.

/WEST=real

- specifies the western bound of the clipping window in IFF units.

ORDNANCE SURVEY MODE

Two additional qualifiers are available in Ordnance Survey mode:

/CLIP_GRID

- force clipping of grids.

/KEEP_TEXT

- retain text outside the window. This qualifier has no effect with /VOID ,i.e text is not retained inside the window.

DESCRIPTION

IWINDOW clips IFF map data to a rectangular boundary or window. The clipping window can be given explicitly using the /NORTH, /SOUTH, /EAST and /WEST qualifiers or it can be extracted from the MH (Map Header) or CP (Control Point) entries by use of the /MAP_HEADER or /CONTROL_POINT qualifiers. The default clipping window is defined by the values in the RA (Range) entry, except when running in Ordnance Survey mode, when /MAP_HEADER is assumed. Explicit bounds can be given in IFF units, or absolute values if /ABSOLUTE is specified.

The map header option uses either

- (i) the basic grid interval in an OS map header, so that the rectangle to which the map is clipped defaults to the standard map size, i.e. with minimum bounds of 0 (zero) and maximum bounds of 10 times the basic grid interval, or
- (ii) the map origin and map extent in a MCE map header. These values must have been correctly set up.

If the OS map header is type 3 or 4, the basic grid interval is read from the header using the position and size specified in the OS map header translation table, LSL\$OS_MH_TABLE. This table is described in the DATA PREPARATION section of the IFFOSTF chapter in the "Convert User Guide". Since a type 4 (CITF) Map Header contains no basic grid interval it is calculated from the scale so that maps at scales 1:1250, 1:2500 and 1:10000 have basic grid intervals of 50m, 100m, and 400m respectively. These are the only scales intended for use with CITF data.

For type 3 and 4 OS map headers, Laser-Scan regard the definitive values for the Eastings, Northings and scale to be those in the map descriptor (MD) entry, not those in the map header. If these values need to be changed, Laser-Scan recommend that they are updated in the MD entry using ITRANS/DESCRIPTOR.

If the control points are being used to define the window then they must be rectangular.

IWINDOW also performs checks on any additional clipping bounds supplied using the /NORTH, /SOUTH, /EAST, and /WEST qualifiers. These checks ensure that

- the user specified clipping bounds are not outside the default bounds, (i.e the window defined by the /MAP_HEADER or /CONTROL_POINT qualifiers, or the range, as appropriate)
- the map bounds chosen are the right way round (East > West, North > South).

If the window contains the whole map, the program does not normally produce an output file. However, if the /ORIGIN qualifier has been given and the window contains the whole map, then an output file is produced containing all the input data, with coordinates relative to the clipping origin. This is an inefficient but possibly useful means of performing a simple map translation.

By default the program will delete all features outside the clipping bounds, but this action can normally be modified through the use of the /FC_IGNORE, /LAYER_IGNORE and /OS qualifiers.

Ordnance Survey mode makes available the following features:

- o By default, IWINDOW will not delete any text features (FC 28) outside the clipping bounds, i.e. /KEEP_TEXT is assumed. This is so that boundary texts are retained when clipping to the map

limits. However, if any of the boundary qualifiers are given (/NORTH, /SOUTH, /EAST, /WEST), or if /VOID is specified, text will be clipped as any other feature. When text clipping is being performed on composite text features, all components that are outside the clipping area will be removed.

- o Grids (FC 398 or 998) will not normally be clipped. To force such clipping, the /CLIP_GRID qualifier must be given.

When splitting a reentrant feature, the new feature will normally be given a new, unique FSN. If it is required that it should have the same FSN as the original, then the /RETAIN_FSN qualifier should be given (except in OS mode when it is invalid).

If a feature is deleted then any TC (Transmitted Comment) or CH (CHaracter command) entries associated with that feature (ie. immediately in front of it) are also deleted. Likewise, if a new feature is generated then copies of the TC/CH entries will be made.

Empty features with FSN 0 are deleted. Any TC/CH entries preceding them are retained.

For IFF revision 0 files containing ZS entries and IFF revision 1 files containing CB entries with Z-coordinates, the Z-value of the intersection between string and clipping boundary is estimated using simple linear interpolation. This is based upon the known heights of the two points defining the segment which contains the intersection point. If the height of either of the end points of such a segment is unknown, the intersection point will also be given an unknown height.

All other attributes in CB entries are copied over for points inside the clipping area. New points created on the clipping boundary will have their attributes unset.

PROGRAM OPERATION

IWINDOW opens the given input file and reads the RA (RAnge) entry to ascertain the maximum extent of the map. It then reads the MH (Map Header) entry to check the data type, and uses this entry if /MAP_HEADER was specified, or the CP (Control Point) entry if /CONTROL_POINTS was specified, to work out the default clipping bounds. There are no default bounds if neither of the IFF entry qualifiers was given. If /OFFSET was specified, account is taken of it, and then any explicit bounds are transferred to give the actual clipping bounds.

The output file is opened and the input file is clipped into it. This involves copying entries from the input file into the output file, with the exception of features, and layer entries, which are dealt with in a more complex manner.

Internally, IWINDOW treats ST and ZS entries in IFF revision 0 input files as if they were CB entries. For each feature, the program notes where the feature starts, and finds the associated CB (which may be the first of several). The points in the CB are clipped to the

clipping bounds.

If a feature is outside the clipping bounds, it is discarded, and if it is inside it is simply copied across.

If a feature crosses the boundary, the first part inside is written out with the feature's serial number, and any subsequent (reentrant) parts are written out as features with either a temporary feature number of -1, or, if /RETAIN_FSN was specified, the same feature serial number as the original.

After the whole file has been processed, the program goes back to any features with a dummy feature serial number and assigns each a new, unique serial number (greater than all the original feature serial numbers).

Area features specified by the /AREA qualifier are dealt with using a different algorithm and are closed correctly if clipped.

EXAMPLES

```
$ IWINDOW/CONTROL_POINTS/WEST=9600/SOUTH=23100/LAYER_IGNORE=3 DFAD1
NEW_DFAD1<CR>
  ELAPSED: 00:00:49.0 CPU: 0:00:26.69  BUFIO: 4  DIRIO: 602  FAULTS: 177
$
```

This example illustrates a common use of IWINDOW. Input is from LSL\$IF:DFAD1.IFF;0 while output is to LSL\$IF:NEW_DFAD1.IFF. The control points have been used in conjunction with explicit western and southern bounds to form the window. In addition, clipping has been suppressed for layer 3, so all features in that layer will be copied, unchanged, to the output file. \$STATUS is set to SS\$_NORMAL.

```
$
$ IWINDOW/MAP_HEADER/SOUTH=14100/NORTH=18600/EAST=14100/VOID/LOG<CR>
_From: DFAD2<CR>
_To: NEW_DFAD2<CR>
%LSLLIB-I-OPENED, IFF file DUA0:[JAMIE.TEST]DFAD2.IFF;0 opened for read
Bounds N =18600.0, S =14100.0, E =14100.0, W = 600.000
Clipping not suppressed for any feature codes
Clipping not suppressed for any layers
Clipping to void

%LSLLIB-I-OPENED, IFF file DUA0:[JAMIE.TEST]NEW_DFAD2.IFF opened for write
%IWIN-I-LAYER, layer 0 found
Layer 0: 0 features deleted, 0 clipped and 0 created

%IWIN-I-LAYER, layer 1 found
Feature 1 (1) split 1 time
Feature 45 (45) deleted
Feature 46 (46) deleted
Feature 47 (47) deleted
Feature 48 (48) deleted
Feature 49 (49) split 1 time
Feature 50 (50) split 3 times
Feature 51 (51) deleted
Feature 52 (52) deleted
Feature 53 (53) deleted
Feature 54 (54) deleted
Feature 55 (55) deleted
Feature 56 (56) deleted
Feature 57 (57) deleted
Feature 58 (58) deleted
Feature 59 (59) deleted
Feature 60 (60) deleted
Feature 61 (61) deleted
Feature 62 (62) deleted
Feature 63 (63) deleted
Feature 64 (64) deleted
Feature 65 (65) deleted
Feature 66 (66) deleted
Feature 67 (67) deleted
Feature 68 (68) deleted
Feature 69 (69) deleted
```

Feature 70 (70) deleted
Feature 71 (71) deleted
Feature 72 (72) deleted
Feature 73 (73) deleted
Feature 74 (74) deleted
Feature 75 (75) was clipped
Feature 100 (100) deleted
Feature 101 (101) deleted
Feature 102 (102) deleted
Feature 103 (103) deleted
Feature 104 (104) deleted
Feature 105 (105) deleted
Feature 106 (106) deleted
Feature 108 (108) deleted
Feature 109 (109) deleted
Feature 110 (110) deleted
Feature 111 (111) deleted
Feature 112 (112) deleted
Feature 113 (113) deleted
Feature 114 (114) split 1 time
Feature 124 (124) deleted
Feature 125 (125) split 1 time
Layer 1: 42 features deleted, 6 clipped and 7 created

%IWIN-I-LAYER, layer 2 found
Feature 346 (346) deleted
Feature 347 (347) deleted
Feature 348 (348) deleted
Feature 349 (349) deleted
Feature 350 (350) deleted
Layer 2: 5 features deleted, 0 clipped and 0 created

%IWIN-I-LAYER, layer 3 found
Single point feature 2576 (2576) deleted
Single point feature 2577 (2577) deleted
Single point feature 2582 (2582) deleted
Single point feature 2858 (2858) deleted
Single point feature 2859 (2859) deleted
Single point feature 2860 (2860) deleted
Feature 2910 split 1 time
Layer 3: 6 features deleted, 1 clipped and 1 created

Summary: 53 features deleted, 7 clipped and 8 created
Features 10832 to 10839 were created
No layers suppressed from clipping

ELAPSED: 00:01:05.23 CPU: 0:00:35.55 BUFIO: 9 DIRIO: 673 FAULTS: 126
\$

In this example, the map header has been used in conjunction with explicit southern, northern and eastern bounds to form the window. Thus only one edge of the window coincides with the original map boundary. Clipping has been performed to a void, i.e. the rectangle has been 'cut out' of the map. The user has also selected the /LOG qualifier which has resulted in the output of supplementary messages and summary information. \$STATUS is set to SS\$_NORMAL.

```
$ IWINDOW/OS/NORTH=500/CLIP_GRID/LOG/OUTPUT=SP8061 SP8061 SP8061S<CR>
%LSLLIB-I-OPENED, IFF file DUA0:[LSL.IFF]SP8061.IFF;7 opened for read
%LSLLIB-I-OPENED, IFF file DUA0:[LSL.IFF]SP8061S.IFF;4 opened for write
%IWIN-I-LAYER, layer 1 found
%IWIN-I-LAYER, layer 11 found
  ELAPSED: 00:00:36.60  CPU: 0:00:18.15  BUFIO: 9  DIRIO: 341  FAULTS: 144
$
```

This example illustrates the use of IWINDOW in OS mode. The /MAP_HEADER qualifier was present by default, and the window was modified by an explicit northern bound. Grid lines were clipped, and because of the presence of /NORTH, texts have also been clipped. Use of the /LOG qualifier has resulted in the appearance of supplementary messages. Because /OUTPUT has also been specified the summary information has been sent to the file SP8061.LIS, reproduced below. The format is similar to that when output is to the terminal, but differs in the extra header information. If there had been a history entry in the IFF file, this would also have been printed out in full. \$STATUS is set to SS\$_NORMAL.

```
===== IWINDOW
=====
```

IWINDOW invoked by: JAMIE, Process: Jamie Hulme, on: 28-AUG-86 13:05:13

Input file : LSL\$IF:SP8061.IFF;0

Running in OS mode

Bounds N = 500.000, S = 0.000, E =1000.000, W = 0.000

Clipping not suppressed for any feature codes

Clipping texts

Clipping grids

Output file : LSL\$IF:SP8061S.IFF

Feature 124 (124) was clipped
Feature 190 (190) was clipped
Feature 191 (191) was clipped
Feature 214 (214) was clipped
Feature 215 (215) was clipped
Feature 217 (217) was clipped
Feature 240 (240) was clipped
Feature 247 (247) was clipped
Feature 248 (248) was clipped
Feature 251 (251) was clipped
Feature 254 (254) was clipped
Feature 260 (260) was clipped
Feature 277 (277) was clipped
Feature 300 (300) was clipped
Single point feature 303 (303) deleted
Single point feature 304 (304) deleted
Single point feature 305 (305) deleted
Single point feature 306 (306) deleted
Single point feature 307 (307) deleted
Single point feature 308 (308) deleted

Single point feature 309 (309) deleted
Single point feature 310 (310) deleted
Single point feature 311 (311) deleted
Feature 317 (317) was clipped
Feature 351 (351) was clipped
Layer 1: 9 features deleted, 16 clipped and 0 created

Feature 9974 (9974) deleted
Feature 9975 (9975) deleted
Feature 9976 (9976) deleted
Feature 9977 (9977) deleted
Feature 9978 (9978) deleted
Feature 9980 (9980) was clipped
Feature 9981 (9981) was clipped
Feature 9982 (9982) was clipped
Feature 9983 (9983) was clipped
Feature 9984 (9984) was clipped
Feature 9985 (9985) deleted
Feature 9986 (9986) was clipped
Feature 9987 (9987) was clipped
Feature 9988 (9988) was clipped
Feature 9989 (9989) was clipped
Feature 9990 (9990) was clipped
Feature 9991 (9991) was clipped
Feature 9992 (9992) was clipped
Feature 9993 (9993) was clipped
Feature 9994 (9994) was clipped
Feature 9995 (9995) was clipped
Layer 11: 6 features deleted, 15 clipped and 0 created

Summary: 15 features deleted, 31 clipped and 0 created

MESSAGES (SUCCESS)

These messages indicate that the program has completed successfully.

MAPINBDY, map is totally within window, no output

Explanation: As it says.

User action: Check that the correct clipping bounds were specified.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). Those referring to the opening and closing of IFF files, and the notification of layers, will only appear if /LOG has been specified on the command line. The messages giving boundary information are continuation messages and will appear regardless of /LOG.

CLIPBOUNDS, Clip boundary: 'real' 'real' 'real' 'real'

Explanation: The supplied clipping boundaries are out of range, described in a further message.

User action: Check the supplied boundaries in the command line.

DEFAULTMDORG, Default MD origin of (0.0, 0.0) used

Explanation: There was an error finding the IFF file map descriptor origin, and a default value of (0,0) is used.

User action: Check the IFF file for a valid MD2 entry.

FRTREAD, reading FRT file 'file-spec'

Explanation: This message indicates that the specified FRT file is being read.

User action: None

LAYER, layer 'integer' found

Explanation: The program has reached the given layer in the input file, and will commence clipping if appropriate.

User action: None.

EGTW, East boundary should be greater than West

Explanation: This is a continuation message, and indicates that the particular combination of qualifiers used to specify the clipping bounds has resulted in the eastern boundary being less than the western one.

User action: Reissue the IWINDOW command and qualifiers, taking care that the new combination will result in a valid window.

NGTS, North boundary should be greater than South

Explanation: This is a continuation message, and indicates that the particular combination of qualifiers used to specify the clipping bounds has resulted in the northern boundary being less than the southern one.

User action: Reissue the IWINDOW command and qualifiers, taking care that the new combination will result in a valid window.

NOLCM, no LITES2 guidance file (no reentrant features)

Explanation: This message appears when /LITES2 was specified and no guidance file. This will occur when there are no reentrant features.

User action: None.

OSMH, OS map header

Explanation: The IFF file has an OS map header.

User action: None.

MESSAGES (WARNING)

These messages indicate that IWINDOW has been unable to guarantee that features can be correctly clipped.

BADMHBGI, invalid basic grid interval 'integer' in OS MH entry

Explanation: An invalid value for the basic grid interval has been found in an OS map header entry. This means that the default clipping bounds will not have been properly set up, and the program may subsequently fail on that account. If it does not fail, the window may well differ from the expected dimensions.

User action: Investigate the cause of the invalid basic grid interval by running OMHED and examining the map header. If the error can be easily traced and corrected, run IWINDOW again on the corrected input file; otherwise, consult your supervisor.

BADMHMCE, inconsistency in MCE MH entry

Explanation: The latitude and longitude values for the map corners do not agree with those for the map extent. This means that the default clipping bounds will not have been properly set up, and the program may subsequently fail on that account. If it does not fail, the window may well differ from the expected dimensions.

User action: Investigate the cause of the inconsistency by running MCEHED and examining the map header. If the error can be easily traced and corrected, run IWINDOW again on the corrected input file; otherwise, consult your supervisor.

BADEBOUND, East boundary not in expected range ('real' to 'real')

Explanation: The value given with /EAST is not within the default window as defined by the map header, control points or range.

User action: Check that the clipping bounds used are the desired ones.

BADNBOUND, North boundary not in expected range ('real' to 'real')

Explanation: The value given with /NORTH is not within the default window as defined by the map header, control points or range.

User action: Check that the clipping bounds used are the desired ones.

BADSBOUND, South boundary not in expected range ('real' to 'value')

Explanation: The value given with /SOUTH is not within the default window as defined by the map header, control points or range.

User action: Check that the clipping bounds used are the desired ones.

BADWBOUND, West boundary not in expected range ('real' to 'real')

Explanation: The value given with /WEST is not within the default window as defined by the map header, control points or range.

User action: Check that the clipping bounds used are the desired ones.

DELLCMFIL, LITES2 guidance file deleted

Explanation: This is normally a continuation message and indicates that the output LITES2 guidance file has been deleted.

User action: See the appropriate error.

DELOUTFIL, output IFF file deleted

Explanation: This is normally a continuation message and indicates that the output file has been deleted.

User action: See the appropriate error.

FCNOTFND, feature code 'integer' not defined in FRT

Explanation: The program has tried to look up the specified feature code in the FRT file specified, and has not found a definition.

User action: If the feature code in the IFF data is correct, then either correct the FRT or use a different FRT file.

FEATINCOMP, feature 'integer' ('integer') incomplete

Explanation: This is normally a continuation message and indicates that a feature in the file is incomplete.

User action: Depends on the main error which caused this action. The input file should be examined using IPATCH. LITES2 will repair incomplete IFF features.

ILLEGLCP, illegal values in CP entry

Explanation: The control point values result in either the eastern boundary being less than the western, or the northern less than the southern. This means that the default clipping bounds will not have been properly set up, and the program may subsequently fail on that account. If it does not fail, the window may well differ from the expected dimensions.

User action: Investigate the cause of the problem by running IPATCH and examining the control points. If the error can be easily traced and corrected, run IWINDOW again on the corrected input file; otherwise, consult your supervisor.

ILLEGLMH, illegal values in MH entry

Explanation: The latitude and longitude values for the map corners agree with those for the map extent, but result in either the eastern boundary being less than the western, or the northern less than the southern. This means that the default clipping bounds will not have been properly set up, and the program may subsequently fail on that account. If it does not fail, the window may well differ from the expected dimensions.

User action: Investigate the cause of the problem by running MCEHED and examining the map header. If the error can be easily traced and corrected, run IWINDOW again on the corrected input file; otherwise, consult your supervisor.

MHMDDIFF, conflicting Map Header and Map Descriptor entries

Explanation: One or more of the map header (MH) entries do not match the map descriptor (MD) entries. Specifically the origin Eastings and Northings are tested against the values held in the MD entry of a type 3 IFF OS Map Header.

User action: The values for the Eastings and Northings held in the MH should be kept up to date with those in the MD. You may need to consult your system manager if the MH has been altered using in-house editors. To change the values in the MD, use ITRANS/DESCRIPTOR.

NOEJ, no EJ entry in input IFF file

Explanation: The EJ entry was missing from the end of the input file.

User action: If the input file is to be used again, run IMEND to insert an EJ entry.

NOEM, missing EM entry in input IFF file

Explanation: An EM entry was not found where expected in the input file.

User action: If the input file is to be used again, use IPATCH to determine where in the file the EM entry is missing. If it is the final EM entry in the file, run IMEND to insert an EM entry. If it is anywhere else in the file run LITES2 to insert the missing EM entry.

NOEO, missing EO entry in input IFF file

Explanation: An EO entry was not found where expected in the input file.

User action: If the input file is to be used again, run LITES2 on the file and then immediately exit from LITES2 to insert the missing EO entry.

NOMD, No MD entry in input IFF file

Explanation: No MD entry was not found in the input file and the /ABSOLUTE qualifier will not be valid. **User action:** If this is not expected then examine the IFF file with IPATCH to determine the cause of the problem as all IFF files should have a MD entry.

NOTOSMODE, OS map not running in OS mode

Explanation: The /OS qualifier was not given on the command line, although the input file has an OS map header.

User action: Run IWINDOW again, if necessary, with the /OS qualifier.

NOV2MD, MD entry in input IFF file is not version 2

Explanation: The MD entry found in the input file is version 1 and the /ABSOLUTE qualifier will not be valid.

User action: None

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. Some of the messages may indicate a programming error. All result in the deletion of the output IFF file.

BADCP, CP entry is not rectangular

Explanation: The control points in the input file do not define a rectangle.

User action: Edit the input file using IPATCH, ensuring the corner points are correctly set.

BADRA, invalid RA entry in input IFF file

Explanation: One, or both, of the minimum range values is greater than or equal to the corresponding maximum range value.

User action: Edit the input file using IPATCH, ensuring the range are correctly set.

CMDLINERR, Error reading command line

Explanation: There was an error reading the supplied command line.

User action: Check the command line, and resubmit.

EOFINFTR, unexpected end of file whilst updating feature

Explanation: While updating the NF entries of features with new serial numbers, the end of the output file was found where a NF entry was expected. This is almost certainly a programming error.

User action: A note should be made of the qualifiers used, the input file should be preserved, and the error should be reported to Laser-Scan. IPATCH can be used to update all missing serial numbers in the output file manually. At the same time, the file should be checked for any other inconsistencies if it is to be used again.

EOICPYFEAT, unexpected end of input IFF file copying feature

Explanation: Before outputting a feature, its details are copied from the input file. The end of the input file was found within a feature when an NF, TC, or CH entry was expected. This could be due either to a corrupt IFF file or a programming error.

User action: Examine the input file contents using IPATCH and if it is the cause of the problem, edit it. If no inconsistencies are found, make a note of the qualifiers used, preserve the input file, and report the error to Laser-Scan.

EQUALBDY, boundary lines should not be the same

Explanation: This message indicates that the combination of qualifiers used to specify the clipping bounds has resulted in either the eastern and western or the northern and southern boundaries being the same.

User action: Reissue the IWINDOW command and qualifiers, taking care that the new combination will result in a valid window.

ERRCLOLCM, error closing LITES2 guidance file on logical unit 'integer'

Explanation: The LITES2 guidance file could not be closed. This message should be followed by other supplementary messages which should give an indication of the source of the error.

User Action: Consult your system manager, and if necessary Laser-Scan.

ERRCLOLOG, error closing log file on logical unit 'integer'

Explanation: The log file could not be closed - note that this is SYS\$OUTPUT if /OUTPUT was not specified with /LOG. A continuation message immediately follows giving the LSLLIB error which gave rise to the problem, and if applicable a further continuation message gives the system error. These messages will give some indication as to whether the error is due to system problems or programming errors.

User action: Consult your supervisor or system manager, and if necessary Laser-Scan.

ERRDELLCM, cannot delete LITES2 guidance file on unit 'integer'

Explanation: If /LITES2 is specified and no reentrant features are found the LITES2 file is normally deleted. This error occurs if for some reason this cannot be deleted.

User action: Consult your supervisor or system manager, and if necessary Laser-Scan.

ERROPNLOG, error opening 'file-spec' as logfile on logical unit 'integer'

Explanation: If /OUTPUT was not specified with /LOG, the program attempts to open SYS\$OUTPUT as the log file. A continuation message immediately follows giving the LSLLIB error which gave rise to the problem, and if applicable a further continuation message gives the system error. These messages will give some indication as to whether any of the more serious errors are due to system problems or programming errors.

User action: If the problem is due to a simple mistake, reissue the IWINDOW command and qualifiers with the correct file specification. Otherwise, consult your supervisor or system manager, and if necessary Laser-Scan.

ERRPOSNO, error repositioning to NO (posn 'address')

Explanation: An NO entry in the output file was about to be updated with a pointer to the EO prior to writing the EO entry itself. This is almost certainly a programming error.

User action: A note should be made of the qualifiers used, the input file should be preserved, and the error should be reported to Laser-Scan.

ERRPOSTC, error repositioning to TC/CH (position 'address')

Explanation: A TC or CH entry in the input file was to be copied to the output file, but one address used is incorrect. This is almost certainly a programming error.

User action: Make a note of the command qualifiers used, preserve the input file, and report the error to Laser-Scan.

ERRWRTLOG, error writing to log file

Explanation: The log file could not be written to - note that this is SYS\$OUTPUT if /OUTPUT was not specified with /LOG. A continuation message immediately follows giving the LSLIB error, and a further continuation message gives the system error. The error is most likely to be due to system problems and this will be clarified by the messages.

User action: Consult your supervisor or system manager.

FRTFAIL, error reading FRT file 'file-spec'

Explanation: An error has occurred reading the specified FRT file. Additional messages should accompany this message giving some indication of the problem.

User action: Consult your system manager and if necessary Laser-Scan.

GOTHI, more than one HI entry in input IFF file

Explanation: More than one history entry was found in the input IFF file.

User action: Preserve the input file and consult Laser-Scan.

HIWRONG, HI is not the second entry in input IFF file

Explanation: The history entry should always be the second entry and in this case is not.

User action: Preserve the input file and consult Laser-Scan.

IFFCREATERR, Error creating output IFF file 'filename'

Explanation: There was an error creating the output IFF file.

User action: Check the requested IFF file name, or command line.

IFFOPNERR, Error opening input IFF file 'filename'

Explanation: There was an error opening the input IFF file.

User action: Check the requested IFF file, or command line.

ILLEGFC, illegal feature code 'integer' with '/'qualifier'

Explanation: The feature code specified with the given qualifier does not lie in the range 0 to 32767.

User action: Respecify the command line, taking care that feature codes fall within the allowed range.

ILLEGLAY, illegal IFF layer number 'integer' with 'qualifier'

Explanation: The layer number specified with the given qualifier does not lie in the range 0 to 32767.

User action: Respecify the command line, taking care that layer numbers fall within the allowed range.

NOBGIINMH4, no BGI in type 4 Map Header (MH) entry.

Explanation: An attempt was made to process an IFF file with a type 4 (CITF) MH entry which does not contain a Basic Grid Interval.

User action: At present IWINDOW cannot process IFF files containing type 4 Map Headers which are only intended for use by Ordnance Survey Great Britain. If IWINDOW is run without the /OS qualifier, the Map Header will be ignored.

NOCP, no CP entry in input IFF file

Explanation: A control point entry could not be found in the input file.

User action: Run IPATCH on the input file to ascertain the reason for this.

NOINPNF, cannot find NF entry in input IFF file where expected

Explanation: Before outputting a feature, its details are copied from the input file. The NF was not found in the input file where expected. This is almost certainly a programming error.

User action: A note should be made of the qualifiers used, the input file should be preserved, and the error should be reported to Laser-Scan.

NOMH, no MH entry in input IFF file

Explanation: A map header entry could not be found in the input file.

User action: Run IPATCH on the input file to ascertain the reason for this.

NOOUTNF, cannot find NF entry to update

Explanation: While updating the NF entries of features with new serial numbers, an NF was not found where expected. This is almost certainly a programming error.

User action: A note should be made of the qualifiers used, the input file should be preserved, and the error should be reported to Laser-Scan.

NOTANAREA, feature 'integer' has only 'integer' point(s), cannot be an area

Explanation: The specified feature cannot be an area as has less than the required number of points.

User action: Check that the features specified with the /AREA qualifier are correct.

NORA, no RA entry in input IFF file

Explanation: A range entry could not be found in the input file.

User action: Run IPATCH on the input file to ascertain the reason for this.

RANOTFND, cannot find RA entry in output IFF file

Explanation: Having finished going through the input file, with all the clipping done, the range entry in the output file has to be updated. This entry cannot be found, indicating a probable programming error.

User action: Make a note of the command qualifiers used, preserve the input file, and report the error to Laser-Scan.

TOOBIGFEAT, too many points ('integer') in feature 'integer' ('integer')

Explanation: There are too many points in the specified feature, the maximum possible is 5000.

User action: If it is not possible to reduce the number of points in the feature by splitting or filtering it then please submit an SPR.

TOOMANYFC, 'integer' feature codes with /'qualifier', maximum is 'integer'

Explanation: More than the maximum allowed number of feature codes have been specified with the given qualifier.

User action: Reissue the IWINDOW command and qualifiers, taking care that the number of feature codes specified does not exceed the maximum. If a larger number of feature codes is an ongoing requirement, please submit a software modification request to Laser-Scan.

TOOMANYFEAT, too many features created (max 'integer')

Explanation: More than the maximum allowed number of new features have been created.

User action: One way round this problem would be to suppress some of the layers, perform the clipping, and then clip the output file, this time suppressing the layers which have already been clipped. If this problem is likely to occur frequently, please submit a software modification request to Laser-Scan.

TOOMANYLAY, 'integer' layers with /'qualifier', maximum is 'number'

Explanation: More than the maximum allowed number of layers have been specified with the given qualifier.

User action: Reissue the IWINDOW command and qualifiers, taking care that the number of layers specified does not exceed the maximum. If a larger number of layers is an ongoing requirement, please submit a software modification request to Laser-Scan.

TOOMANYPTS, too many points ('integer') in ST entry, feature FSN 'integer' ('integer')

Explanation: There are too many points in the ST or CB entry read from the input file. This is probably due to an error in the input file.

User action: A note should be made of the qualifiers used, the input file should be preserved, and the error should be reported to Laser-Scan. IPATCH or ITOTEXT should also be used to find where the error occurs in the input file. The output file will almost certainly be incomplete and should either be edited with IPATCH or deleted.

UNEXPEOI, unexpected end of input IFF file

Explanation: The end of the input file was found within a feature - an EF, ST or ZS or CB entry was expected. This could be due either to a corrupt IFF file or a programming error.

User action: Examine the input file contents using IPATCH and if it is the cause of the problem, edit it. If no inconsistencies are found, make a note of the qualifiers used, preserve the input file, and report the error to Laser-Scan. The output file will almost certainly be incomplete and should either be edited with IPATCH or deleted.

UNSETMH, cannot derive boundaries with unset map header

Explanation: The /MAP_HEADER qualifier was specified, and since the map header is neither OS nor MCE, the default clipping bounds cannot be derived.

User action: Reissue the IWINDOW command with /CONTROL_POINTS to obtain the default clipping bounds, or specify some or all of the bounds explicitly with the boundary qualifiers. Note that if all four boundary qualifiers are given, default clipping bounds need not be specified.

WRONGBDY, boundaries were specified the wrong way round

Explanation: This message indicates that the combination of qualifiers used to specify the clipping bounds has resulted in the boundaries being the wrong way round. A further continuation message explains which boundaries are in error.

User action: Check for any additional messages which may help to diagnose the problem. Reissue the IWINDOW command and qualifiers, taking care that the new combination will result in a valid window.

WRONGCUST, map is not OS, cannot run in OS mode

Explanation: The /OS qualifier was given on the command line, but the input file does not have an OS map header.

User action: Run IWINDOW again, without the /OS qualifier.

MESSAGES (OTHER)

In addition to the above messages which are generated by IWINDOW itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users' guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

CHAPTER 22

MODULE ICUT

MODULE **ICUT**

REPLACES None, ICUT is a completely new IMP utility.

FUNCTION

ICUT takes existing features from an input IFF file and creates a new IFF file containing features generated from the input features. The generated features are subject to constraints of length and number of points per feature. Valid command line switches are:

- o /AC Selects attributes to be attached to output features,
- o /FC Only select particular FCs,
- o /FCP Specify file of feature code pairs,
- o /INVISIBLES Cut features at invisible line breaks,
- o /LAYER Select features from a particular IFF layer,
- o /LENGTH Specify the maximum length of an output feature,
- o /LOG Output additional information to SYS\$OUTPUT,
- o /NEWFC Sets the output FC of the output features,
- o /POINTS Sets the maximum number of points per output feature,
- o /XYONLY Disables height coordinates from being output.

If more than one criterion for selection is specified then all specified criteria have to match for a feature to be processed. Features not matching the selection criteria are not processed, but simply copied to the output file.

FORMAT

\$ ICUT input-file-spec output-file-spec

Command qualifiers**Defaults**

/AC=(FSN , FC , Z)

No ACs are added

/FC=integer[,...]

All features selected

/FCP=fcpfilename

No file

/[NO]INVISIBLES

/NOINVISIBLES

/LAYER=integer[,...]

All layers selected

/[NO]LOG

/NOLOG

/LENGTH = integer

No length restriction

/NEWFC=integer

keep input feature codes

/POINTS=integer

features contain one point only

/XYONLY

height coordinates retained

PROMPTS

_From: input-file-spec
_To: output-file-spec

PARAMETERS

input-file-spec

- specifies the IFF file from which selections are to be made. Any part of the file specification which is not supplied will be taken from the default specification 'LSL\$IF:IFF.IFF'.

output-file-spec

- specifies the IFF file which is to be created. Any part of the file specification which is not supplied will be taken from the parsed input file specification. Note that a version number must **not** be specified for the output file. If a file with the specified name already exists, a new file will be created with the version number incremented by one.

COMMAND QUALIFIERS**/AC=KEYWORD**

- Where KEYWORD is one or more of: FC (feature code), FSN (feature serial number) or Z (height).
- Enables the ancillary codes which are to be attached to the output features. By default no ancillary codes are attached to the output features. If qualifier arguments are supplied then only the specified ancillary codes will be attached.
- If /AC=Z is specified, the program checks that the output file is restricted to containing only single point features (/POINTS=1). If the value specified to /POINTS is not 1, then ICUT issues an error message.

/FC=integer[,...]

- Enables selection of all features which have a FC of the specified numbers. Selected features will be processed before corresponding features are written to the output file. Features that are not selected will be copied to the output file. Ranges of FCs may be specified. IFF feature codes must lie in the range 0 to 32767. A maximum of 1024 feature codes may be specified.

/FCP=fcpfilename

- Specifies the file which contains pairs of feature codes (input FC, output FC). The specified output FCs will be generated in the output file if the specified input FCs are detected in the input file. IFF feature codes must lie in the range 0 to 32767. a maximum of 1024 feature code pairs may be specified.

/INVISIBLES**/NOINVISIBLES**

- Enables cutting of features at invisible (pen-up) steps on a feature.

/LAYER=integer[,...]

- Enables selection of all features which lie in the layers with the specified values. Selected features will be processed before corresponding features are written to the output file. Features that are not selected will be copied to the output file. Ranges of layers may be specified. IFF layer numbers must lie in the range 1 to 32767. A maximum of 1024 layer numbers may be specified.

/LENGTH = real

- enables dissection of the linear features in the input file to features in the output file which have lengths less than the specified length. If the specified length is smaller than the length of an input feature, then the feature will not be dissected.
- Note that the length of a 2-Dimensional feature is its length along the ground. For a 3-Dimensional line feature, the length is its length in space.

/LOG

/NOLOG

- this will result in supplementary messages being sent to SYS\$OUTPUT. Supplementary messages are generated when files are opened or closed, and at important stages of processing.

/NEWFC=integer

- Specifies the global feature code for features to be written to the output file. The specified FC will be used for all selected features from the input file. Input features which are not selected will retain their feature codes. The IFF feature code must lie in the range 0 to 32767.

/POINTS=integer

- Specifies the number of points per feature in the output file. The output features will be created by bisecting the input features at the specified interval. The specified interval must be greater than 0.

/XYONLY

- Enables discarding of height coordinates from the output file.

RESTRICTIONS

- o The /POINTS and the /LENGTH qualifiers can not both be specified.
- o The AC code for height (/AC =Z) will only be written to the output file if the output features are restricted to one point per feature (/POINTS=1).
- o ICUT will not simply copy the entire contents of the input file to the output file. To prevent this, if neither of the /POINTS or /LENGTH qualifiers are specified, the program issues a warning message and sets the /POINTS=1 qualifier.
- o When the /FCP qualifier has been issued, the /NEWFC qualifier cannot be issued

DESCRIPTION

ICUT creates a new IFF file containing features cut (dissected) from features from an existing IFF file selected on the basis of attributes specified by qualifiers on the command line. Features may be selected on the basis of:

- o IFF layer,
- o feature code, (the first word of an IFF FS (Feature Status) entry),
- o any combination of the above.

The characteristics of the output features may be selected on the basis of:

- o ancillary code, which ACs will be set for the output features,
- o feature code, the output feature code for all selected input features,
- o points, the maximum number of points in an output feature,
- o length, the maximum length (in space) of an output feature.

If more than one criterion for selection is specified then all specified criteria have to match for a feature to be processed and corresponding point features written to the output file.

In the event of a feature not matching all the selection criteria, the feature will be copied unmodified to the output file.

In the event of no features matching the selection criteria, ICUT issues an error message and deletes the output file.

An IFF HI (HISTORY) entry is written to the output file only if the input IFF file contains an HI entry. Similarly, a type 2 MD entry is only written to the output file if the input file contains a type 2 MD entry.

EXAMPLES**\$ ICUT INPUT OUTPUT /POINTS=5 <CR>**

ELAPSED: 0 00:00:03.51 CPU: 0:00:00.39 BUFIO: 6 DIRIO: 29 FAULTS: 195

\$

In this example, ICUT takes input from the IFF file "INPUT.IFF" and produces the output IFF file "OUTPUT.IFF".

The /POINTS=5 qualifier specifies that the output file will contain features which contain no more than 5 points each.

\$ ICUT INPUT OUTPUT /LENGTH=300 <CR>

ELAPSED: 0 00:00:04.05 CPU: 0:00:00.46 BUFIO: 6 DIRIO: 22 FAULTS: 203

In this example, ICUT takes input from the IFF file "INPUT.IFF" and produces the output IFF file "OUTPUT.IFF".

The /LENGTH=300 qualifier specifies that the output file will contain features which are no longer than 500m each.

\$ ICUT INPUT OUTPUT /POINTS=5 /LOG <CR>

%ICUT-I-CMDQUAL, The following command qualifiers were detected on the command line:

/LOG /POINTS .

%LSLLIB-I-IFFOPENED, LSL\$DATAROOT:[LSL.IFF]INPUT.IFF;1 opened for read

%LSLLIB-I-IFFOPENED, LSL\$DATAROOT:[LSL.IFF]OUTPUT.IFF;5 opened for write

%ICUT-I-HIGHINFSN, highest FSN from input IFF file is 1

%ICUT-I-OUTFFT, 40 features were written to the output IFF file

ELAPSED: 0 00:00:04.35 CPU: 0:00:00.45 BUFIO: 6 DIRIO: 20 FAULTS: 261

\$

This example shows that when the /LOG qualifier is specified, additional diagnostic messages are output to SYS\$OUTPUT.

The command line is parsed and any qualifiers detected are output as informational messages. In this case /LOG and /POINTS have been detected (as expected).

ICUT then opens the IFF files and performs basic checks on the input file. If the input file passes the checks, ICUT shows the highest feature serial number (FSN) it encountered, and proceeds to process the file.

At the end of processing, ICUT shows how many features it wrote to the output file.

\$ ICUT INPUT OUTPUT /POINTS=5 /FCP=FRED.PAIR <CR>

ELAPSED: 0 00:00:02.93 CPU: 0:00:00.20 BUFIO: 10 DIRIO: 21 FAULTS: 200

\$

This is an example of the use of the /FCP qualifier. The file "FRED.PAIR" contains information on which feature codes ICUT will operate on, and their corresponding output feature codes. The /POINTS qualifier specifies that the output features will contain no more than 5 points each.

\$ ICUT INPUT OUTPUT /POINTS=5 /AC=(FSN,FC,Z) /LOG <CR>

%ICUT-W-INAPAZ, /AC=Z inappropriate without /POINTS=1

%ICUT-I-CANAZ, cancelling /AC =Z

%ICUT-I-CMDQUAL, The following command qualifiers were detected on the command line:

/AC /AC=FSN /AC=FC /LOG /POINTS .

%LSLLIB-I-IFFOPENED, LSL\$DATAROOT:[LSL.IFF]INPUT.IFF;1 opened for read

%LSLLIB-I-IFFOPENED, LSL\$DATAROOT:[LSL.IFF]OUTPUT.IFF;4 opened for write

%ICUT-I-HIGHINFSN, highest FSN from input IFF file is 1

%ICUT-I-OUTFFT, 40 features were written to the output IFF file

ELAPSED: 0 00:00:05.38 CPU: 0:00:00.38 BUFIO: 6 DIRIO: 22 FAULTS: 220

\$

This example shows the use of the /AC qualifier.

As before, ICUT parses the command line. In this case it has detected incompatible qualifiers "/AC=Z" and "/POINTS=5". ICUT assumes that output features which consist of more than 1 point (i.e. a line feature) can have separate heights for each of the points contained in them. These features can not be have an overall height associated with them and ICUT cancels the /AC=Z qualifier.

ICUT then proceeds to open the IFF files and after checking the input IFF file, processes it.

The /AC=FC and /AC=FSN qualifiers cause the FC and FSN attributes of the input features to be attached to the output features as ancillary codes.

EXAMPLES - OF BADLY FORMED COMMAND LINE SPECIFICATION

The following examples illustrate badly formed command line specification to ICUT. They illustrate the types of error messages that are likely to be encountered when using ICUT. They will cause an error condition to be generated and will cause ICUT to terminate processing.

\$ ICUT <CR>

From: <CTRL Z>

%CLI-W-NOCOMD, no command on line - reenter with alphabetic first character

ELAPSED: 0 00:00:00.35 CPU: 0:00:00.02 BUFIO: 0 DIRIO: 0 FAULTS: 86

Here the CLI routines have detected a premature termination of the command line and issue a warning message. ICUT then terminates. The minimum information that is required on the command line for ICUT to run is the input and output IFF filenames.

This error can also occur while batch processing if the line being submitted does not contain both the input and output IFF filenames.

\$ ICUT NONEXISTENT OUTPUT <CR>

%ICUT-W-WARNINSUF SWITCH, neither /POINTS or /LENGTH specified

%ICUT-I-SETPTS, defaulting to /POINTS=1

%LSLLIB-E-IFFOPEN, IFF error opening file "LSL\$IF:NONEXISTENT.IFF;0" for read

%IFF-E-OPEN, error opening IFF file

-RMS-E-FNF, file not found

ELAPSED: 0 00:00:01.77 CPU: 0:00:00.20 BUFIO: 6 DIRIO: 9 FAULTS: 161

In this example, ICUT has failed to detect the presence of either the /POINTS or the /LENGTH qualifiers. ICUT assumes as a default that the output file will consist of single point features and sets /POINTS=1. and displays warning messages.

ICUT then attempts to open the IFF file "NONEXISTENT.IFF" and fails. This generates an error and ICUT terminates with an error condition.

\$ ICUT INPUT OUTPUT /POINTS=5 /LENGTH=300 <CR>

%CLI-W-CONFLICT, illegal combination of command elements - check documentation
\POINTS\

ELAPSED: 0 00:00:00.64 CPU: 0:00:00.03 BUFIO: 2 DIRIO: 0 FAULTS: 107

Here ICUT has detected incompatible qualifiers /POINTS and /LENGTH. Since only one of these must be specified, ICUT terminates with an error condition.

\$ ICUT INPUT OUTPUT /POINTS=5 /FCP=FRED.PAIR /NEWFC=300 <CR>

%CLI-W-CONFLICT, illegal combination of command elements - check documentation
\FCP\
ELAPSED: 0 00:00:00.86 CPU: 0:00:00.03 BUFIO: 2 DIRIO: 0 FAULTS: 106

In this example ICUT has detected incompatible qualifiers /FCP and /NEWFC.

Since /FCP defines an input file which contains details on the features to cut, and the feature codes of the resulting output features, /NEWFC is redundant. ICUT terminates with an error condition since it has no way of knowing which of the two qualifiers was actually wanted.

MESSAGES (INFORMATIONAL)

These messages give information only, and require no immediate action by the user (except perhaps to verify that the correct options have been selected). Generally they will only appear if /LOG has been specified on the command line.

DELETE, deleting output file 'file-spec'

Explanation: ICUT is to delete the specified output IFF file because an error occurred while processing the input IFF file. The output IFF file is therefore incomplete and useless.

User action: Determine why the input IFF file is corrupt and attempt to fix the problem (e.g. using IMEND, IPATCH). Re-run ICUT with corrected IFF file or use a different input iff file.

LAYER, layer 'integer' found

Explanation: The specified layer has been found in the input file and is being processed.

User action: None.

LINESKIP, skipping over line

Explanation: An error occurred while processing a line from pair-file. ICUT is ignoring the offending line and continuing processing.

User action: None.

SETPTS, defaulting to /POINTS=1

Explanation: Neither the /POINTS nor the /LENGTH qualifiers were specified on the command line. ICUT is assuming that the output features will be restricted to single point features.

User action: None.

CANAZ, cancelling /AC=Z

Explanation: The /AC=Z was specified, but /POINTS was specified to be a value other than 1. ICUT assumes that the /AC=Z qualifier is incorrect and sets it accordingly.

User action: None.

HIGHINFSN, highest FSN from input IFF file is 'integer'

Explanation: This is a diagnostic message and is output when the input IFF file has been checked for consistency.

User action: None.

OUTFFT, 'integer' features were written to the output IFF file

Explanation: This message is diagnostic and is output at the end of processing.

User action: None.

CMDQUAL, the following command qualifiers were detected on the command line:

Explanation: This message is diagnostic and confirms that ICUT has correctly analysed the command line arguments.

User action: None.

UNIMPLE, Processing of entry code 'string' not implemented in this version'

Explanation: This indicates that ICUT is unable to process the specified entry as it has not been implemented in the current version of the software.

User action: None.

MESSAGES (WARNING)

These messages indicate that an error has occurred from which processing is continued. These are accompanied by messages which indicate how the program is dealing with the error. Processing will continue until normal completion or a severe error is detected (see next section). No user action is required until the termination of the program.

WARNNOCMA, expected comma not found

Explanation: The comma which separates the integer pairs in the pair file was not found.

WARNNOINT, expected integer not found

Explanation: a comma was found as the first entry of a line in the pair file, or a comma was found but no integer.

UNEXPEOL, unexpected end of line

Explanation: Self-explanatory

WARNINSUFSWITCH, neither /POINTS or /LENGTH specified

Explanation: neither the /POINTS or /LENGTH qualifiers were specified on the command line.

UNEXPENTRY, unexpected entry found

Explanation: while processing a feature, ICUT has found an entry which it did not expect.

MESSAGES (ERROR)

These messages indicate an error in processing which has caused the program to terminate. The most likely causes are a corrupt or otherwise invalid input IFF file, or an error related to command line processing and file manipulation. With the exception of the IFF file closure errors, no output file will be produced if one of these error messages is output.

ERRCLINFL, error closing input IFF file 'file-spec'

Explanation: The specified file cannot be closed. This message will be accompanied by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. If the problem continues, notify the system manager.

ERRCLOUTFL, error closing output IFF file 'file-spec'

Explanation: The specified file cannot be closed. This message will be accompanied by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. If the problem continues, notify the system manager.

ERROPN, error opening input file 'file-spec'

Explanation: The specified file cannot be opened. This message will be accompanied by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the error. If the problem continues, notify the system manager.

INVALFC, invalid /FC or /NEWFC argument - "'integer'" is outside range 0 to 32767

Explanation: IFF feature codes lie in the range 0 to 32767.

User action: Respecify the ICUT command line and ensure that the arguments for the /FC and /NEWFC qualifier are in the range 0 to 32767.

INVLAPT, invalid /POINTS argument - "'integer'" is less than 1

Explanation: The smallest number of points in an IFF feature is 1.

User action: Respecify the ICUT command line and ensure that the value for the /POINTS qualifier is greater than 1.

INVALLEN, invalid /LENGTH argument - "'number'" is too small

Explanation: Self-explanatory.

User action: Respecify the ICUT command line and ensure that the value for the /LENGTH qualifier is valid.

INVALLAY, invalid /LAYER argument - "'integer'" is outside range 0 to 32767

Explanation: IFF layer numbers lie in the range 0 to 32767.

User action: Respecify the ICUT command line and ensure that the arguments for the /LAYER qualifier lies within the specified range.

UNEXPEOF, unexpected end of IFF file

Explanation: The input IFF file terminated before an IFF EJ entry was encountered.

User action: Use IPATCH to determine whether the file has an EJ entry. If the file lacks an EJ entry use IMEND to correctly terminate the file. Try re-running the IMP utility which failed. If the problem persists then the internal structure of the IFF file is probably corrupt. Try reading it into LITES2 and then EXIT from LITES2. Re-run the IMP utility.

UNEXPERR, unexpected error while reading from a file

Explanation: An error has occurred while reading a non IFF file.

User action: Check the any non-IFF files specified on the command line. Check that the specified pair file contains only text entries and it is free from control characters. If the problem persists, the pair file is probably corrupted and must be repaired or re-written before re-running the IMP utility.

The following errors indicate that the structure of the input IFF file is corrupt. Use IPATCH to determine where the problem is and IMEND to attempt to fix the problem.

- o TOOMANYMH, MH (map header) entry found before closing of previous MH entry
- o UNEXPEM EM (end map) entry found but no corresponding MH found
- o TOOMANYNO, NO (map header) entry found before closing of previous MH entry
- o UNEXPEO EO (end overlay) entry found but no corresponding NO found
- o TOOMANYNF, NF (new feature) entry found before closing of previous EF entry

- o UNEXPEF EF (end feature) entry found but no corresponding NF found

MESSAGES (OTHER)

In addition to the above messages which are generated by ICUT itself, other messages may be produced by the command line interpreter (CLI) and by Laser-Scan libraries. In particular, messages may be generated by the IFF library. These are introduced by '%IFF' and are documented in the IFF library users guide. In most cases IFF errors will be due to a corrupt input file, and this should be the first area of investigation. If the cause of the error cannot be traced by the user and Laser-Scan are consulted, then the input file should be preserved to facilitate diagnosis.

APPENDIX A

ITRANS - SPHEROIDS OFFERED

Data about spheroids is stored within ITRANS as the length of the equatorial (a) and polar (b) axes. The flattening (f) and the eccentricity (e) are derived from these by the formulae:-

$$f = (a-b)/a$$
$$e^{**2} = (a^{**2} - b^{**2})/a^{**2}$$

NOTE

This is not necessarily the way in which the spheroids are formally defined.

0 Clarke 1866

Equatorial Radius:	6378206.4
Polar Radius:	6356583.8
Flattening:	1/294.9786982
Eccentricity squared:	0.006768658
Eccentricity:	0.08227185422

1 Clarke 1880

Equatorial Radius:	6378249.145
Polar Radius:	6356514.86955
Flattening:	1/293.465
Eccentricity squared:	0.00680351128
Eccentricity:	0.08248340004

2 Bessel

Equatorial Radius:	6377397.155
Polar Radius:	6356078.96284
Flattening:	1/299.1528131
Eccentricity squared:	0.00667437222
Eccentricity:	0.08169683118

3 New International 1967

Equatorial Radius:	6378157.5
Polar Radius:	6356772.2
Flattening:	1/298.2496154
Eccentricity squared:	0.00669455047
Eccentricity:	0.08182023266

4 International 1924 (Hayford)

Equatorial Radius:	6378388.0
Polar Radius:	6356911.94613
Flattening:	1/297.0
Eccentricity squared:	0.00672267002
Eccentricity:	0.08199188998

5 World Geodetic System 72 (WGS 72)

Equatorial Radius: 6378135.0
Polar Radius: 6356750.519915
Flattening: 1/298.2599986
Eccentricity squared: 0.00669431781
Eccentricity: 0.08181881086

6 Everest

Equatorial Radius: 6377276.3452
Polar Radius: 6356075.4133
Flattening: 1/300.8016994
Eccentricity squared: 0.00663784664
Eccentricity: 0.08147298106

7 World Geodetic System 66 (WGS 66)

Equatorial Radius: 6378145.0
Polar Radius: 6356759.769356
Flattening: 1/298.2499981
Eccentricity squared: 0.0066945419
Eccentricity: 0.08182018025

8 Geodetic Reference System 1980

Equatorial Radius: 6378137.0
Polar Radius: 6356752.31414
Flattening: 1/298.2572221
Eccentricity squared: 0.00669438002
Eccentricity: 0.08181919104

9 Airy

Equatorial Radius: 6377563.396
Polar Radius: 6356256.91
Flattening: 1/299.3249753
Eccentricity squared: 0.00667053976
Eccentricity: 0.08167337241

10 Modified Everest

Equatorial Radius: 6377304.063
Polar Radius: 6356103.039
Flattening: 1/300.8017001
Eccentricity squared: 0.00663784663
Eccentricity: 0.08147298097

11 Modified Airy

Equatorial Radius: 6377340.189
Polar Radius: 6356034.448
Flattening: 1/299.3249655
Eccentricity squared: 0.00667053998
Eccentricity: 0.08167337376

12 Walbeck

Equatorial Radius: 6376896.0
Polar Radius: 6355834.8467
Flattening: 1/302.7800002
Eccentricity squared: 0.00659454809
Eccentricity: 0.08120682293

13 Southeast Asia

Equatorial Radius: 6378155.0
Polar Radius: 6356773.3205
Flattening: 1/298.3000002
Eccentricity squared: 0.00669342162
Eccentricity: 0.08181333398

14 Australian National

Equatorial Radius: 6378160.0
Polar Radius: 6356774.719
Flattening: 1/298.2499973
Eccentricity squared: 0.00669454192
Eccentricity: 0.08182018037

15 Krassovsky

Equatorial Radius: 6378245.0
Polar Radius: 6356863.0188
Flattening: 1/298.3000004
Eccentricity squared: 0.00669342161
Eccentricity: 0.08181333397

16 Hough

Equatorial Radius: 6378270.0
Polar Radius: 6356794.343479
Flattening: 1/297.0000006
Eccentricity squared: 0.00672267001
Eccentricity: 0.08199188989

17 Mercury 1960

Equatorial Radius: 6378166.0
Polar Radius: 6356784.283666
Flattening: 1/298.3000008
Eccentricity squared: 0.0066934216
Eccentricity: 0.0818133339

18 Modified Mercury 1968

Equatorial Radius: 6378150.0
Polar Radius: 6356768.337303
Flattening: 1/298.3000008
Eccentricity squared: 0.0066934216
Eccentricity: 0.0818133339

19 Sphere of radius 6370997.0M

Radius: 6370997.0

20 Sphere of radius 6371229.3M

Radius: 6371229.3

21 Clarke 1880 (IGN)

Equatorial Radius: 6378249.2

Polar Radius: 6356515.0

Flattening: $1/293.4660213$

Eccentricity squared: 0.00680348765

Eccentricity: 0.08248325676

22 World Geodetic System 84 (WGS 84)

Equatorial Radius: 6378137.0

Polar Radius: 6356752.314

Flattening: $1/298.2572201$

Eccentricity squared: 0.00669438007

Eccentricity: 0.08181919131

APPENDIX B

ITRANS - PROJECTIONS OFFERED

For more details and formulae see "Map Projections Used by the U.S. Geological Survey" by John P. Snyder, published by the U.S. Geological Survey. Most of the information in this appendix is taken from that publication.

Projections marked with a "*" are only defined in their spherical form.

0 Unset

This code is set in the map descriptor if the projection data is not set up. It is not applicable to ITRANS

1 Universal Transverse Mercator

This is a particular case of the Transverse Mercator projection (code 9 - see below) that is used world wide for large scale mapping. As it is so widely used, the projection parameters are predefined.

The earth is divided into 60 zones, each 6 degrees wide in longitude. Bounding meridians are evenly divisible by 6 degrees, and zones are numbered from 1 to 60 proceeding east from the 180th meridian. Each of these zones is used to define a Transverse Mercator projection, using the meridian in the centre of each zone as the central meridian and reducing its scale to 0.9996 of true scale (a 1:2500 reduction). This reduction was chosen to minimise scale variation in a given zone; the variation reaching 1 part in 1000 from true scale at the Equator. The lines of true scale are approximately parallel to and approximately 180km east and west of the central meridian. Between them the scale is too small; beyond them it is too great.

In the Northern Hemisphere, the Equator at the central meridian is considered the origin, with an X coordinate of 500000.000 m and a Y coordinate of 0.0 m.

For the Southern hemisphere, the same point is the origin, but, while X remains 500000.000 m, Y is 10000000.00 m. In each case, numbers increase towards the east and north. Negative coordinates are thus avoided.

Additional information required

1. The UTM zone.

Negative zone numbers indicate that the southern hemisphere is being considered.

If a zone number of 0 is input, ITRANS will work out the zone for itself, but will require two additional pieces of information:

2. The latitude of any point in the zone

3. The longitude of any point in the zone

2 State Plane Coordinates

The State Plane Projection System is the system used for the mapping of the states of the USA. The projections used are the Transverse Mercator, Lambert Conformal Conic and Oblique Mercator (Hotine) depending on the extent, shape and geographical location of the state.

The system is based on the CLARKE 1866 spheroid.

Each state has its own projection system (or systems) and its own projection parameters. Information for each state should be sought in Snyder.

Within ITRANS this information is stored as an unique zone number. The available zone codes, and the state and area they cover are as follows:

101 ALABAMA	EAST	102 ALABAMA	WEST
201 ARIZONA	EAST	202 ARIZONA	CENTRAL
203 ARIZONA	WEST	301 ARKANSAS	NORTH
302 ARKANSAS	SOUTH	401 CALIFORNIA	I
402 CALIFORNIA	II	403 CALIFORNIA	III
404 CALIFORNIA	IV	405 CALIFORNIA	V
406 CALIFORNIA	VI	407 CALIFORNIA	VII
501 COLORADO	NORTH	502 COLORADO	CENTRAL
503 COLORADO	SOUTH	600 CONNECTICUT	---
700 DELAWARE	---	901 FLORIDA	EAST
902 FLORIDA	WEST	903 FLORIDA	NORTH
1001 GEORGIA	EAST	1002 GEORGIA	WEST
5101 HAWAII	1	5102 HAWAII	2
5103 HAWAII	3	5104 HAWAII	4
5105 HAWAII	5	1101 IDAHO	EAST
1102 IDAHO	CENTRAL	1103 IDAHO	WEST
1201 ILLINOIS	EAST	1202 ILLINOIS	WEST
1301 INDIANA	EAST	1302 INDIANA	WEST
1401 IOWA	NORTH	1402 IOWA	SOUTH
1501 KANSAS	NORTH	1502 KANSAS	SOUTH
1601 KENTUCKY	NORTH	1602 KENTUCKY	SOUTH
1701 LOUISIANA	NORTH	1702 LOUISIANA	SOUTH
1703 LOUISIANA	OFFSHORE	1801 MAINE	EAST
1802 MAINE	WEST	1900 MARYLAND	---
2001 MASSACHUSETTS	MAINLAND	2002 MASSACHUSETTS	ISLAND
2101 MICHIGAN	EAST	2102 MICHIGAN	CENTRAL
2103 MICHIGAN	WEST	2111 MICHIGAN	NORTH
2112 MICHIGAN	CENTRAL	2113 MICHIGAN	SOUTH
2201 MINNESOTA	NORTH	2202 MINNESOTA	CENTRAL
2203 MINNESOTA	SOUTH	2301 MISSISSIPPI	EAST
2302 MISSISSIPPI	WEST	2401 MISSOURI	EAST
2402 MISSOURI	CENTRAL	2403 MISSOURI	WEST
2501 MONTANA	NORTH	2502 MONTANA	CENTRAL
2503 MONTANA	SOUTH	2601 NEBRASKA	NORTH
2602 NEBRASKA	SOUTH	2701 NEVADA	EAST
2702 NEVADA	CENTRAL	2703 NEVADA	WEST
2800 NEW HAMPSHIRE	---	2900 NEW JERSEY	---
3001 NEW MEXICO	EAST	3002 NEW MEXICO	CENTRAL
3003 NEW MEXICO	WEST	3101 NEW YORK	EAST
3102 NEW YORK	CENTRAL	3103 NEW YORK	WEST
3104 NEW YORK	LONG ISLAND	3200 NORTH CAROLINA	---

3301 NORTH DAKOTA	NORTH	3302 NORTH DAKOTA	SOUTH
3401 OHIO	NORTH	3402 OHIO	SOUTH
3501 OKLAHOMA	NORTH	3502 OKLAHOMA	SOUTH
3601 OREGON	NORTH	3602 OREGON	SOUTH
3701 PENNSYLVANIA	NORTH	3702 PENNSYLVANIA	SOUTH
3800 RHODE ISLAND	---	3901 SOUTH CAROLINA	NORTH
3902 SOUTH CAROLINA	SOUTH	4001 SOUTH DAKOTA	NORTH
4002 SOUTH DAKOTA	SOUTH	4100 TENNESSEE	---
4201 TEXAS	NORTH	4202 TEXAS	NORTH CENTRAL
4203 TEXAS	CENTRAL	4204 TEXAS	SOUTH CENTRAL
4205 TEXAS	SOUTH	4301 UTAH	NORTH
4302 UTAH	CENTRAL	4303 UTAH	SOUTH
4400 VERMONT	---	4501 VIRGINIA	NORTH
4502 VIRGINIA	SOUTH	4601 WASHINGTON	NORTH
4602 WASHINGTON	SOUTH	4701 WEST VIRGINIA	NORTH
4702 WEST VIRGINIA	SOUTH	4801 WISCONSIN	NORTH
4802 WISCONSIN	CENTRAL	4803 WISCONSIN	SOUTH
4901 WYOMING	EAST	4902 WYOMING	EAST CENTRAL
4903 WYOMING	WEST CENTRAL	4904 WYOMING	WEST
5001 ALASKA	ZONE NO. 1	5002 ALASKA	ZONE NO. 2
5003 ALASKA	ZONE NO. 3	5004 ALASKA	ZONE NO. 4
5005 ALASKA	ZONE NO. 5	5006 ALASKA	ZONE NO. 6
5007 ALASKA	ZONE NO. 7	5008 ALASKA	ZONE NO. 8
5009 ALASKA	ZONE NO. 9	5010 ALASKA	ZONE NO. 10

Additional information required

1. The SPS zone code

3 Albers Conical Equal Area

- o Conic projection
- o Equal Area
- o Not a perspective projection
- o Parallels are unequally spaced arcs of concentric circles, more closely spaced at the north and south edges of the map
- o Meridians are equally spaced radii of the same circles, cutting parallel at right angles
- o There is no distortion in scale or shape along two standard parallels, normally, or along just one
- o Poles are arcs of circles
- o Used for equal area maps of regions with predominant east-west expanse, especially the conterminous United States
- o Presented by Albers in 1805

Additional information required

1. Latitude of first standard parallel
2. Latitude of second standard parallel

NOTE

It is not possible to have the same absolute latitude on opposite sides of the equator for these two standard parallels. In that case a cylindrical projection should be used

3. Longitude of the central meridian
4. Latitude of the projection origin
5. Easting coordinate of the origin
6. Northing coordinate of the origin

4 *Lambert Conformal Conic*

- o Conic projection
- o Conformal
- o Not a perspective projection
- o Parallels are unequally spaced arcs of concentric circles, more closely spaced near the centre of the map
- o Meridians are equally spaced radii of the same circles, cutting parallels at right angles
- o Scale is true along two standard parallels, normally, or along just one
- o Pole in same hemisphere as standard parallels is a point; other pole is at infinity
- o Used for maps of regions with predominant east-west expanse
- o Presented by Lambert in 1772

Additional information required

1. Latitude of first standard parallel
2. Latitude of second standard parallel

NOTE

It is not possible to have the same absolute latitude on opposite sides of the equator for these two standard parallels. In that case a cylindrical projection should be used

3. Longitude of the central meridian
4. Latitude of the projection origin
5. Easting coordinate of the origin
6. Northing coordinate of the origin

5 *Mercator*

- o Cylindrical projection
- o Conformal
- o Not a perspective projection
- o Meridians are equally spaced straight lines
- o Parallels are unequally spaced straight lines, closest near the equator, cutting meridians at right angles
- o Scale is true along the Equator, or along two parallels equidistant from the Equator
- o Loxodromes (rhumb lines) are straight lines
- o Not perspective
- o Poles are at infinity; great distortion in area of polar regions
- o Used for navigation
- o Presented by Mercator in 1569

Additional information required

1. Latitude that has true scale
2. Longitude of central meridian
3. Easting coordinate of the origin
4. Northing coordinate of the origin

6 *Polar Stereographic*

- o Azimuthal
- o Conformal
- o All parallels are circles
- o All meridians are straight lines
- o A perspective projection (only for sphere, not for spheroid)
Projection is onto a plane, tangential to one pole, from the opposite pole
- o Directions from the centre of the projection are true
- o Scale increases away from the centre of the projection
- o Pole opposite the centre of projection cannot be projected
- o Used for polar maps and miscellaneous special maps
- o Apparently invented by Hipparchus (2nd century B.C.)

Additional information required

1. Latitude that has true scale
The hemisphere of this latitude determines the pole that is used as the centre of the projection
2. Longitude straight down from North Pole of map, up from South Pole
3. Easting coordinate of the origin
4. Northing coordinate of the origin

7 *Polyconic*

- o Neither conformal or equal area
- o Not a perspective projection
- o Parallels of latitude (except for Equator) are arcs of circles, but are not concentric
- o Central meridian and Equator are straight lines; all other meridians are complex curves
- o Scale is true along each parallel and along the central meridian, but no parallel is "standard"
- o Free of distortion only along the central meridian

- o Used almost exclusively in slightly modified form for large scale mapping in the U.S. until the 1950's
- o Was apparently originated about 1820 by Hassler

Additional information required

1. Latitude of origin of projection
2. Longitude of central meridian
3. Easting coordinate of the origin
4. Northing coordinate of the origin

8 *Equidistant Conic*

- o Conic
- o Not a perspective projection
- o Meridians are straight lines converging on a polar axis but not on the pole
- o Parallels are arcs of concentric circles
- o Scale is true along all meridians and along the standard parallels (or parallel)
- o Meridian spacing is true on the standard parallels and decreases toward the pole
- o Parallels are spaced at true scale along the meridians
- o Meridians and parallels intersect each other at right angles
- o Often called the 'Simple Conic' or just 'Conic'.
- o Used for atlas maps portraying mid-latitude areas

NOTE

It is possible to define this projection with one or two standard parallels.
These are termed "Format A" and "Format B" within ITRANS.

Additional information required

1. Latitude of first standard parallel

2. Latitude of second standard parallel (Format B only)
3. Longitude of the central meridian
4. Latitude of the projection origin
5. Easting coordinate of the origin
6. Northing coordinate of the origin

9 *Transverse Mercator*

- o Cylindrical (transverse)
- o Conformal
- o Not a perspective projection
- o Central meridian, each meridian 90 degrees from central meridian and the Equator are straight lines
- o Other meridians and parallels are complex curves
- o Scale is true along central meridian, or along two straight lines equidistant from and parallel to central meridian. (These lines are only approximately straight for the ellipsoid.)
- o Scale becomes infinite 90 degrees from central meridian
- o Used extensively for large scale mapping all over the world, particularly as the UTM projection, but also by national mapping agencies (e.g. the State Plane Coordinate System in the USA uses it for some states and it is used for the U.K. national grid)
- o The formulae used for the ellipsoidal form are only accurate for zones of up to 4 degrees from the central meridian.
- o Presented by Lambert in 1772 and further analysed (in its ellipsoidal form) by Gauss in 1822 and by Krüger in 1912 and 1919.
- o Sometimes known in its ellipsoidal form as 'Gauss-Krüger' or 'Gauss Conformal'.

Additional information required

1. Scale factor at central meridian
2. Latitude of origin
3. Longitude of central meridian
4. Easting coordinate of the origin

5. Northing coordinate of the origin

10 * *Stereographic*

- o Azimuthal
- o Conformal
- o A perspective projection where the point of projection is the point opposite the point of tangency
- o Central meridian and a particular parallel (if shown) are straight lines
- o All meridians on the polar aspect and the Equator on the equatorial aspect are straight lines
- o All other meridians and parallels are shown as arcs of circles
- o Directions from the centre of the projection are true
- o Scale increases away from the centre of the projection
- o Point opposite the centre of the projection cannot be plotted
- o Used for polar maps and miscellaneous special maps
- o Apparently invented by Hipparchus (2nd century B.C.)

Additional information required

1. Latitude of centre of projection
2. Longitude of centre of projection
3. Easting coordinate of the origin
4. Northing coordinate of the origin

11 * *Lambert Azimuthal Equal Area*

- o Azimuthal
- o Equal Area
- o A perspective projection
- o All meridians on the polar aspect, the Central meridian in other aspects and the Equator on the equatorial aspect are straight lines
- o The outer meridian of a hemisphere in the equatorial aspect and the parallels in the polar aspect are circles

- o All other meridians and parallels are shown as arcs of circles
- o Directions from the centre of the projection are true
- o Scale decreases radially as the distance increases from the centre, the only point without distortion
- o Scale increases in the direction perpendicular to radii as the distance increases from the centre
- o Point opposite the centre of the projection is shown as a circle surrounding the map
- o Used for maps of continents and hemispheres
- o Presented by Lambert in 1772

Additional information required

1. Latitude of centre of projection
2. Longitude of centre of projection
3. Easting coordinate of the origin
4. Northing coordinate of the origin

12 * *Azimuthal Equidistant*

- o Azimuthal
- o Neither equal area nor conformal
- o Not a perspective projection
- o Distances measured from the centre are true
- o Distances not measured along radii from the centre are not correct
- o The centre of projection is the only point without distortion
- o Directions from the centre are true
- o All meridians on the polar aspect, the central meridian on other aspects, and the Equator on the equatorial aspect are straight lines
- o Parallels on the polar projection are circles spaced at true intervals and are equidistant
- o The outer meridian of a hemisphere on the equatorial aspect (for the sphere) is a circle

- o All other meridians and parallels are complex curves
- o Point opposite the centre is shown as a circle surrounding the map
- o Used in the polar aspect for world maps and maps of both polar hemispheres
- o Used in the oblique aspect for atlas maps of continents and world maps for aviation and radio use
- o Known for many centuries in the polar aspect

Additional information required

1. Latitude of centre of projection
2. Longitude of centre of projection
3. Easting coordinate of the origin
4. Northing coordinate of the origin

13 * *Gnomonic*

- o Azimuthal
- o In the polar aspect meridians are straight lines radiating from the point of tangency; on other aspects the meridians are straight lines
- o The parallels are concentric circles in the polar aspect; on other aspects they are complex curves
- o A perspective projection (projection is from centre of the earth)
- o Directions from the centre of the projection are true
- o Scale, angular and areal deformations are extreme, rapidly increasing away from the centre of the projection
- o Cannot show a full hemisphere
- o All great circles are straight lines on projection, and thus the shortest distance between two points is shown as a straight line
- o Used for seismic work and navigation charts

Additional information required

1. Latitude of centre of projection
2. Longitude of centre of projection

3. Easting coordinate of the origin

4. Northing coordinate of the origin

14 * *Orthographic*

- o Azimuthal
- o Neither conformal nor equal area
- o A perspective projection where the point of projection is at infinity
- o All meridians and parallels are ellipses, circles or straight lines
- o Only one hemisphere can be shown at a time
- o Much distortion near the edge of the hemisphere; only the centre is distortion free
- o Directions from the centre are true
- o Scale in the direction of the lines of latitude is true in the polar aspect
- o Used only in the spherical form
- o Known by Egyptians and Greeks 2000 years ago

Additional information required

1. Latitude of centre of projection
2. Longitude of centre of projection
3. Easting coordinate of the origin
4. Northing coordinate of the origin

15 * *General Vertical Near-Sided Perspective*

- o Azimuthal
- o A perspective projection where the point of projection is at any point in space, such as a satellite
- o Used for pictorial representations

Additional information required

1. Height of perspective point above sphere (in projection units)
2. Latitude of centre of projection
3. Longitude of centre of projection
4. Easting coordinate of the origin
5. Northing coordinate of the origin

16 * *Sinusoidal*

- o Pseudocylindrical
- o Equal area
- o Central meridian is a straight line; all other meridians are shown as sinusoidal curves
- o Parallels are equally spaced straight lines
- o Not a perspective projection
- o Scale is true along central meridian and all parallels
- o Used for world maps with single central meridian or in interrupted form with several central meridians
- o Used for maps of South America and Africa
- o Used since the mid-16th century

Additional information required

1. Longitude of central meridian
2. Easting coordinate of the origin
3. Northing coordinate of the origin

17 * *Equirectangular*

- o Cylindrical
- o Neither equal area nor conformal
- o Not a perspective projection
- o Meridians and parallels are equidistant straight lines, intersecting at right angles

- o Poles shown as lines
- o Very simple construction
- o Used for world or regional maps
- o Presented by Eratosthenes (B.C.) or Marinus (A.D. 100)

Additional information required

1. Latitude with true scale
2. Longitude of central meridian
3. Easting coordinate of the origin
4. Northing coordinate of the origin

18 * *Miller Cylindrical*

- o Cylindrical
- o Neither equal area nor conformal
- o Not a perspective projection
- o Meridians and parallels are straight lines, intersecting at right angles
- o Meridians are equidistant; parallels spaced further apart away from Equator
- o Poles shown as lines
- o Compromise between Mercator and other cylindrical projections
- o Used for world maps
- o Presented by Miller in 1942

Additional information required

1. Longitude of central meridian
2. Easting coordinate of the origin
3. Northing coordinate of the origin

19 * *Van der Grinten*

- o Neither equal area nor conformal
- o Not a perspective projection
- o Shows entire globe enclosed in a circle
- o Central meridian and Equator are straight lines
- o All other meridians and parallels are arcs of circles
- o A curved modification of the Mercator projection, with great distortion in the polar areas
- o Equator is true to scale
- o Used for world maps
- o Presented by van der Grinten in 1904

Additional information required

1. Longitude of central meridian
2. Easting coordinate of the origin
3. Northing coordinate of the origin

20 *Oblique Mercator (Hotine)*

- o Cylindrical
- o Conformal
- o Not a perspective projection
- o Two meridians 180 degrees apart are straight lines
- o Scale is (nearly) true along chosen central line, a great circle at an oblique angle or along two straight lines parallel to central line. It is true at the point selected to be the centre of the projection.
- o Scale becomes infinite 90 degrees from central lines
- o The Y axis of the coordinate system follows the meridian passing through the centre point of the projection, and increases northwards. The X axis lies east west through the same point
- o The origin of this projection is the point where the central line crosses the equator of the aposphere (a sphere whose curvature is the 'total' curvature at the central point of the projection).

- o Used for grids on maps of Alaska and mapping in Switzerland, Madagascar and Borneo
- o Developed 1900-50 by Rosenmund, Laborde, Hotine and others

NOTE

It is possible to define the central line of this projection by defining two points on it, or by defining one point on the line and the azimuth of the line through this point.

These are termed "Format A" and "Format B" within ITRANS.

Additional information required

Format A

1. Scale factor at centre of projection
2. Latitude of centre of projection
3. Easting coordinate of the origin
4. Northing coordinate of the origin
5. Latitude of first point on central line
6. Longitude of first point on central line
7. Latitude of second point on central line
8. Longitude of second point on central line

Format B

1. Scale factor at centre of projection
2. Azimuth of central line at origin of projection
3. Latitude of centre of projection
4. Longitude at which the azimuth of the central line is measured
5. Easting coordinate of the origin
6. Northing coordinate of the origin

21 * *Oblique Mercator (Spherical)*

- o Cylindrical
- o Conformal
- o Not a perspective projection
- o Two meridians 180 degrees apart are straight lines
- o Scale is true along chosen central line (false equator), a great circle at an oblique angle or along two straight lines parallel to central line
- o Scale becomes infinite 90 degrees from central lines
- o The X axis of the coordinate system lies along the false equator, and the Y axis is at right angles to this, where it cuts the true equator
- o The false origin of this spherical form of the projection is called the Datum. ITRANS expects this to be defined in geographical coordinates, and calculates the projection coordinates of the true origin from these.
- o Used for aeronautical charts
- o Developed 1900-50 by Rosenmund, Laborde, Hotine and others

NOTE

It is possible to define the false equator of this projection by defining two points on it, or by defining its vertex (highest point).

These are termed "Format A" and "Format B" within ITRANS.

Additional information required

Format A

1. Scale factor at centre of projection
2. Latitude of datum of projection (the datum is the false origin of the projection)
3. Longitude of datum of projection
4. Latitude of first point on central line
5. Longitude of first point on central line
6. Latitude of second point on central line

7. Longitude of second point on central line

Format B

1. Scale factor at centre of projection
2. Latitude of datum of projection (the datum is the false origin of the projection)
3. Longitude of datum of projection
4. Latitude of vertex of central line
5. Longitude of vertex of central line

22 *Stereographic (UKSL formulae)*

This is a projection used internally by Laser-Scan. It is based on a conformal earth.

This projection is based on the INTERNATIONAL spheroid.

See above for characteristics of the Stereographic projection.

Additional information required

1. Radius of Conformal earth
2. Latitude of centre of projection
3. Longitude of centre of projection
4. Easting coordinate of the origin
5. Northing coordinate of the origin

23 *Cassini*

- o Cylindrical
- o Neither equal-area or conformal, but a compromise between the two
- o Central meridian, meridians 90 degrees from the central meridian and the Equator are straight lines; other meridians and parallels are complex curves.
- o Scale is true along central meridian, and along lines perpendicular to central meridian. Scale is constant, but not true, along lines parallel to central meridian (or nearly so in spheroidal form).
- o Formerly used for topographic mapping in the United Kingdom where several different central meridians were used (the County Series). Also currently in a few other countries (Cyprus, Czechoslovakia,

Denmark, Germany and Malaysia).

- o The name 'Cassini-Soldner' is often used for the spheroidal form used for topographic mapping. In this case the formulae used are only suitable for use within a few (say 2) degrees on either side of the central meridian.

Additional information required

1. Latitude of origin
2. Longitude of central meridian
3. Easting coordinate of the origin
4. Northing coordinate of the origin

24 Krovak

- o Oblique Conic projection.
- o Conformal
- o Not a perspective projection
- o Projects spheroidal coordinates conformally onto a sphere which are then projected to an oblique conformal conic projection with one standard parallel.
- o The scale of the standard parallel can be set so as to reduce the scale errors within the area of interest.
- o Suitable for areas whose axis is neither a constant latitude nor a constant longitude.
- o Parallels and Meridians are generally complex curves.
- o Used for topographic mapping in Czechoslovakia (where it is used with a left handed coordinate system)

Additional information required

To set up this projection it is necessary to select a spheroidal latitude for the conformal projection onto the sphere. This is the latitude where the mean radius of curvature of the spheroid equals the radius of the sphere.

The origin of the projection is a point on the spheroid that, having been projected onto the sphere, lies on the standard line of the oblique conical projection. The pole of the oblique sphere (and at the same time the angle of the cone of the conic projection) are defined by specifying what can be thought of as the latitude of the origin in terms of the oblique sphere.

To reduce the scale error (which increases with distance on either side of the standard line of the projection) throughout the area, the true scale of the standard line can be reduced by an appropriate factor.

1. Latitude for conformal projection to sphere
2. Latitude of origin point
3. Longitude of origin point
4. Latitude of origin on oblique sphere
5. Scale factor of standard line through the origin
6. Easting coordinate of the origin
7. Northing coordinate of the origin

Use in Czechoslovakia

In Czechoslovakia this projection is used to produce a left handed coordinate system with the apex of the oblique cone as origin. ITRANS produces a right handed coordinate system with the above mentioned origin point as origin. By supplying appropriate parameters (particularly the False Origin) the Czechoslovakian coordinate system can be reproduced, albeit with the signs of the coordinates reversed.

so to use this projection in Czechoslovakia:

Latitude for conformal projection to sphere	:	49 30 00.000N
Latitude of origin	:	48 15 00.000N
Longitude of origin	:	42 30 00.000E from Ferra
Latitude of origin on oblique sphere	:	78 30 00.000N
Scale factor on standard line through origin	:	0.9999
Easting of origin	:	0.000
Northings of origin	:	-1298039.005

Input Point		Output Point	
Latitude	Longitude	Easting	Northing
50 12 32.441N	34 30 59.179E	-568990.997	-1050538.649

25 *Bonne*

- o Pseudoconical
- o Equal area
- o Central meridian is a straight line; other meridians are complex curves
- o Parallels are concentric circular arcs, but the poles are points
- o Scale is true along the central meridian and along all parallels
- o Used for atlas maps of continents and for topographic mapping in some countries (parts of France, Ireland, Morocco and some countries in the Eastern Mediterranean).
- o The limiting case, when the pole is the equivalent of the standard parallel is sometimes called the 'Werner' projection. The opposite limiting case, when the equator is the standard parallel is the 'Sinusoidal' projection.
- o Used by Bonne in mid-18th century, but developed by others (including Werner) in early 16th century.

Additional information required

1. Latitude of standard parallel (which has its true curvature)
2. Longitude of the central meridian
3. Easting coordinate of the origin
4. Northing coordinate of the origin

26 * *Mollweide*

- o Pseudocylindrical
- o Equal Area
- o Central meridian is a straight line; 90th meridians are circular arcs; all other meridians are equally spaced elliptical arcs.
- o Parallels are unequally spaced straight lines, parallel to each other. Poles are points.
- o Scale is true along latitudes 40°44' North and South
- o Used for world maps
- o Inspiration for several world projections

- o Presented by Mollweide in 1805

Additional information required

1. Longitude of central meridian
2. Easting coordinate of the origin
3. Northing coordinate of the origin

27 * *Hammer-Aitoff*

- o Pseudocylindrical
- o Equal Area
- o Central meridian and equator are straight lines; 90th meridians are circular arcs; all other meridians and parallels are complex curves
- o Based on the azimuthal equal area projection with an equatorial aspect and projecting this orthogonally onto a plane through the equator at an angle of 60°.
- o Used for world maps
- o Devised by Hammer in 1892

Additional information required

1. Longitude of central meridian
2. Easting coordinate of the origin
3. Northing coordinate of the origin

28 * *Winkel-III*

- o A hybrid projection computed by taking the arithmetic mean of the Plate Carrée and the Hammer-Aitoff projection.
- o Winkel also produced other hybrid projections by taking the mean of the Plate Carrée and the Sinusoidal (Winkel I) and the mean of the Plate Carrée and Mollweide (Winkel II)
- o For world maps, avoiding the congestion in high latitudes existing in the original projections.

Additional information required

1. Longitude of central meridian

2. Easting coordinate of the origin

3. Northing coordinate of the origin

100 *Geographic (i.e. Lat and Long)*

This is not a projection.

Additional information required

None

101 *UK national grid*

This is a particular case of the Transverse Mercator projection (code 9 - see above) that is used for large scale mapping in the U.K. As it is so widely used in the U.K., the projection parameters are predefined within ITRANS.

The U.K. national grid is based on the AIRY spheroid.

The following projection parameters are used:

1. Scale factor on central meridian 0.9996012717

2. Latitude of origin 49° North

3. Central meridian 2° West

4. Easting coordinate of origin 400000.000

5. Northing coordinate of origin -100000.000

Additional information required

None

APPENDIX C

ITRANS - NUMBER FORMATS FOR PROMPTS

The type of number expected can be ascertained from the format of the default.

These are used when entering the value of a code (e.g. units or projections).

2. REAL numbers.

NOTE

A real number may be in any of the following forms

a	integer - coerced to a real
a.b	real format
a.bEc	exponential format
a.bDc	double precision exponential format
a.b/c.d	rational format

The syntax of a real number may be more formally defined as:

```
<real number> ::= <unsigned real number> |  
                  <sign><unsigned real number>
```

```
<unsigned real number> ::= <basic real number> |  
                             <basic real number><exponent> |  
                             <basic real number>'/'<basic real number>
```

$$\langle \text{sign} \rangle ::= '+' \mid '-'$$
[illegible]

```
<exponent> ::= 'E'<signed decimal number> |
               'D'<signed decimal number>
```

$$\langle \text{whole part} \rangle ::= \langle \text{decimal number} \rangle$$

```
<fraction part> ::= '.'<decimal number>
```

$$\langle \text{signed decimal number} \rangle ::= \langle \text{decimal number} \rangle \mid \langle \text{sign} \rangle \langle \text{decimal number} \rangle$$

```
<decimal number> ::= <digit> |  
                    <decimal number><digit>  
  
    <digit> ::= '0' | '1' | '2' | '3' | '4' |  
                '5' | '6' | '7' | '8' | '9'
```

3. ANGLES.

These are used when projections require additional information, and that information is an angular quantity (e.g. the latitude of the origin).

They are also used when the /POINT qualifier has been given and projection code 100 and unit code 200 specified.

Angles can be specified in the format sDDD MM SS.SSS or DDD MM SS.SSSh where:

s	is + , - or <blank>
h	is N, n, S, s, E, e, W or w; representing a hemisphere
DDD	is an integer value for the degrees, or a real number, if there are no minutes or seconds.
MM	is an integer value for the minutes, or a standard real number if there are no seconds
SS.SSS	is a standard real value for the seconds.

NOTES

- i) In angles the exponential and rational real number formats are not permitted.
- ii) Elements can be excluded from the left of an angle, but not from the right. That is the seconds (and minutes) need not be present, but the degrees must always be.
- iii) The value in the signed format must be less than 360 degrees. The maximum value of an angle in the E/W hemisphere is 180 degrees and in the N/S hemisphere is 90 degrees.
- iv) The value of the minutes and seconds must be less than 60.0.
- v) Sign and hemisphere are mutually exclusive. When the hemisphere form is used, south and west are taken as negative.

EXAMPLE

The following all represent the same angular value:

```
-00 00 30.00  
00 00 30.00 S  
00 00 30 s  
00 00.5 W  
0.00833333 w
```

INDEX

- Akima interpolation, 4-1
- Bezier interpolation, 4-1
- BUNCH filter, 4-1
- CLI, 1-5 to 1-6
- CLI (Command Line Interpreter), 4-32
- Creates a new IFF file from an existing IFF file, 22-1
- DAMP, 1-1 to 1-2, 2-1, 3-1, 5-1, 6-1, 7-1, 8-1, 9-1, 10-1, 11-1, 13-1, 15-1, 17-1, 19-1, 20-1, 21-1
 - and IMP, 1-2
 - CHKLP, 2-1
 - CMPIFF, 3-1
 - component modules, 1-2
 - functional correspondence with IMP, 1-2
 - I2TEXT, 19-1
 - ICLIP, 21-1
 - IED, 1-2, 11-1
 - IFINFO, 7-1
 - IFIX, 5-1
 - IFSN, 13-1
 - IFT, 9-1
 - IPR, 1-2, 20-1
 - IRN, 17-1
 - LIT, 8-1
 - MER, 10-1
 - MIF, 9-1
 - SCI, 7-1
 - SELAC, 15-1
 - SIF, 17-1
 - TEXT2I, 6-1
- DAMP IPR
 - incompatibility with ITRANS, 20-8
 - incompatibility with type 2 MD, 20-8
- DOUGLAS-PEUCKER filter, 4-1
- DUMP, 1-4
- Fig 1, 4-12
- Fig 10, 4-21
- Fig 11, 4-22
- Fig 12, 4-23
- Fig 2, 4-13
- Fig 3, 4-14
- Fig 4, 4-15
- Fig 5, 4-16
- Fig 6, 4-17
- Fig 7, 4-18
- Fig 8, 4-19
- Fig 9, 4-20
- Filters
 - BUNCH, 4-1, 4-3, 4-6
 - and smoothing, 4-7
 - description, 4-6
 - parameters, 4-7
 - DOUGLAS-PEUCKER, 4-1, 4-4, 4-6 to 4-7
 - description, 4-7
 - parameters, 4-7
- FRT, 18-2
- FRT (Feature Representation Table), 2-1, 2-3, 2-6 to 2-9, 21-4
- geometry checking, 2-6
- Getting started with IMP, 1-6
- HELP
 - on-line, 1-1
- HI entry, 1-4
- ICHECK, 2-1
 - and FRT, 2-1, 2-3, 2-6 to 2-9
 - and LITES2, 2-1 to 2-2, 2-4, 2-9
 - and LSL\$FRT:, 2-8
 - and LSL\$IF:, 2-1, 2-4, 2-9
 - available checks, 2-6
 - against FRT, 2-6
 - crossing segments, 2-6
 - default action, 2-6
 - for duplicate points, 2-6
 - for features outside neatlines, 2-6
 - for open features, 2-7
 - for open features in border zone, 2-6
 - neatline and line intersection, 2-7
 - relative speed, 2-6
 - subset definition, 2-6
 - touching segments, 2-6
- command qualifiers, 2-1 to 2-2
 - /BASE_FC, 2-1 to 2-2, 2-8
 - /BORDER, 2-1 to 2-2

- /CROSS, 2-1 to 2-2
- /DUPLICATES, 2-2, 2-7
 - options, 2-7
- DUPLICATES, 2-1
- /FC, 2-3
- FC, 2-1
- /FRT, 2-1, 2-3, 2-7
- /IGNORE, 2-3
- IGNORE, 2-1
- /LAYER, 2-4
- LAYER, 2-1
- /LITES2, 2-1, 2-4
 - example LCM, 2-4, 2-9 to 2-10
- /LOG, 2-1, 2-4
- /MARKERS, 2-1, 2-4, 2-9
 - box feature codes, 2-9
 - box positioning, 2-9
- /MARKERS file characteristics, 2-8
- /NEATLINE, 2-1, 2-4
- /OUTPUT, 2-1, 2-5
- /PARALLEL, 2-1, 2-5
- /PRINTER, 2-1, 2-5
- /TOUCH, 2-1, 2-5
 - default distance, 2-5
- default file-specs
 - FRT, 2-8
 - input file, 2-1
 - LITES2 command file, 2-4
 - /MARKERS IFF file, 2-9
- description, 2-6
- EXAMPLES, 2-11
- neatline definition, 2-2, 2-4, 2-7
- parameter, 2-1
- prompt, 2-1
- RESTRICTIONS, 2-5
- ICUT, 22-1
 - and IFF HI entry, 22-7
 - and LSL\$IF:, 22-3
 - and selection failure, 22-1, 22-7
- command format, 22-2
- command qualifiers, 22-2, 22-4
 - /AC, 22-2, 22-4
 - /FC, 22-2, 22-4
 - /FCP, 22-2, 22-4
 - /INVISIBLES, 22-2, 22-4
 - /LAYER, 22-2, 22-4
 - /LENGTH, 22-2, 22-5
 - /LOG, 22-2, 22-5
 - /NEWFC, 22-2, 22-5
 - /POINTS, 22-2, 22-5
 - /XYONLY, 22-2, 22-5
- default file-specs
 - input file, 22-3
 - output file, 22-3
- description, 22-7
- examples, 22-8
- examples of badly formed
 - commands, 22-10
- messages (error), 22-15
- messages (informational), 22-12
- messages (other), 22-18
- messages (warning), 22-14
- parameters, 22-3
- prompts, 22-3
- restrictions, 22-6
- IDIFFERENCE, 3-1
 - and 'file header' entry order, 3-6
 - and CC (Cubic Coefficients) entries, 3-8
 - and CP (Control Point) entries, 3-8
 - and HI (HISTORY) entries, 3-7
 - and IFF features, 3-8
 - and IFF layers, 3-8
 - and junction entries, 3-6
 - and LSL\$IF:, 3-1
 - and MD (Map Descriptor) entries, 3-7
 - and MD (Map Header) entries, 3-7
 - effect of /OS, 3-7
 - and NS (New Section) entries, 3-7
 - and RA (RANGE) entries, 3-7
 - and real value comparison, 3-6
 - default tolerance, 3-6
 - effect of /TOLERANCE, 3-6
 - factor tolerance, 3-6
 - standard tolerance, 3-6
 - and unconcatenated digitising files, 3-7
 - and voids, 3-6
- COMMAND QUALIFIERS, 3-2
- command qualifiers, 3-1
 - /DEBUG, 3-1 to 3-2
 - /FRT, 3-1 to 3-2
 - /IGNORE, 3-1 to 3-2
 - /LOG, 3-1, 3-3
 - /LOOKAHEAD, 3-1, 3-3
 - /OS, 3-1, 3-3
 - /OS qualifier, 3-7
 - /OUTPUT, 3-1, 3-4
 - /PLOT, 3-1, 3-4
 - file-specs used, 3-4
 - options, 3-4

- /PRINTER, 3-1, 3-5
- /SUMMARY, 3-1, 3-5
- /TOLERANCE, 3-1, 3-5
- comparing line strings, 3-5
- comparing real attributes, 3-5
- default real tolerance, 3-6
- definition of "master" file, 3-1
- definition of "revision" file, 3-2
- DESCRIPTION, 3-5
 - CC (Cubic Coefficients) entries, 3-8
 - Comparing real values, 3-6
 - CP (Control Point) entries, 3-8
 - General, 3-5
 - HI (HISTORY) entries, 3-7
 - IFF features, 3-8
 - IFF layers, 3-8
 - MD (Map Descriptor) entries, 3-7
 - MH (Map Header) entries, 3-7
 - NS (New Section) entries, 3-7
 - RA (Range) entries, 3-7
 - Summary of differences, 3-12
 - What is compared, 3-6
- entries it does not know, 3-6
- EXAMPLES, 3-14
- factor real tolerance, 3-6
- feature comparison, 3-9
 - AC (Ancillary Code), 3-9
 - CB (Coordinate Block), 3-11
 - FS (Feature Status), 3-9
 - RO (ROTation), 3-10
 - ST (coordinate STRing), 3-11
 - TH (THickness or Text Height), 3-10
 - TS (Text Status), 3-10
 - TX (TeXt), 3-10
 - ZS (3d coordinate string), 3-11
- FORMAT, 3-1
- FUNCTION, 3-1
- Ordnance Survey (U.K.) mode, 3-3
 - functions, 3-3
- PARAMETERS, 3-1
- PROMPTS, 3-1
- standard real tolerance, 3-6
- Summary of differences, 3-12
 - sample output, 3-12
- unrecognised entries, 3-6
- IFF
 - and IMP, 1-3
 - feature selection, 16-1
 - IFF conversion Utility, 22-1
- IFF library, 1-4
- IFF MD entry, 20-8
 - and ITRANS, 20-8
 - contents, 20-8
- IFFLIB, 1-4, 9-3
- IFILTER, 1-5
 - and IFF layer 0, 4-6
 - and LSL\$IF:, 4-2
 - and string length, 4-6
- command qualifiers, 4-1 to 4-2
 - /AKIMA, 4-1 to 4-2
 - /BEZIER, 4-1, 4-3
 - /BLANK, 4-4
 - /BUNCH, 4-1, 4-3
 - /DP, 4-1, 4-4
 - /FC, 4-1
 - /FILTER_FIRST, 4-4
 - /LAYER, 4-1, 4-4
 - /LINEAR, 4-2, 4-5
 - /LOG, 4-2, 4-5
 - /MCCONALOGUE, 4-2, 4-5
- command qualifiers/FILTER_FIRST, 4-1
- default file-specs
 - input file, 4-2
 - output file, 4-2
- description, 4-6
- EXAMPLES, 4-10
 - Akima interpolation, 4-10
 - Douglas-Peucker, 4-10
 - Douglas-Peucker filter, 4-11
 - McConalogue interpolation, 4-10
- Interpolation
 - qualifier arguments, 4-8
- measurement units, 4-3
- messages
 - other, 4-32
- parameters, 4-2
- prompt, 4-2
- restrictions, 4-6
- IFIXAREA, 5-1
 - and LSL\$IF:, 5-2
 - and unchecked features, 5-5
 - available checks, 5-1
- COMMAND QUALIFIERS, 5-2
- command qualifiers, 5-1
 - /AC, 5-1 to 5-2
 - /ANTI_CLOCKWISE, 5-1 to 5-2
 - /CLOCKWISE, 5-1 to 5-2
 - /CLOSE, 5-1, 5-3
 - /DELETE, 5-1, 5-3
 - /DFAD, 5-1, 5-3

- /FC, 5-1, 5-3
- /LAYER, 5-1, 5-3
- /LOG, 5-2, 5-4
- /NEAR, 5-2, 5-4
- /OUTPUT, 5-2, 5-4
- /PRINTER, 5-2, 5-4
- /REVERSE, 5-2, 5-4
- default action, 5-5
- DESCRIPTION, 5-5
- EXAMPLES, 5-6
- FORMAT, 5-1
- FUNCTION, 5-1
- PARAMETERS, 5-2
- PROMPT, 5-2
- restriction on number of
 - coordinates in feature, 5-5
- RESTRICTIONS, 5-4
- summary output, 5-5
- IFROMTEXT, 1-3, 6-1
 - action if EOF detected, 6-5
 - and AC entry, 6-4
 - and CB entry, 6-4
 - and CC entry, 6-3
 - and comments, 6-5
 - and CP entry, 6-3
 - and EJ entry, 6-4
 - and EM entry, 6-4
 - and FS entry, 6-4
 - and HI entry, 6-3
 - and IFF junction entries, 6-5
 - and incomplete text file, 6-5
 - and ISN, 6-3
 - and Map Descriptors, 6-7
 - and Map Headers, 6-5
 - and MD entry, 6-3
 - and NF entry, 6-3
 - and NO entry, 6-3
 - and NS entry, 6-3
 - and RA entry, 6-3
 - and ST entry, 6-4
 - and ZS entry, 6-4
- COMMAND QUALIFIERS, 6-2
- command qualifiers, 6-1
 - /ECHO, 6-1 to 6-2
 - /FRT, 6-1 to 6-2
 - /LOG, 6-1 to 6-2
 - /ORIGIN, 6-1, 6-3
- default procedures, 6-9
- DESCRIPTION, 6-3
 - CB entry, 6-8
- Default Procedures, 6-9
- General, 6-3
- IFROMTEXT and Map Descriptors, 6-7
- IFROMTEXT and Map Headers, 6-5
- IFROMTEXT and the /ORIGIN
 - qualifier, 6-8
- editing the text file, 6-5
- EXAMPLES, 6-11
- FORMAT, 6-1
- FUNCTION, 6-1
- IFROMTEXT and Map Descriptor LO
 - and SC values, 6-8
- input from terminal, 6-1
- PARAMETERS, 6-2
- prompt when reading from
 - terminal, 6-1
- PROMPTS, 6-1
- IINFO, 1-4, 7-1
- COMMAND QUALIFIER, 7-2
- command qualifiers, 7-1
 - /ABSOLUTE, 7-1 to 7-2
 - /BRIEF, 7-1 to 7-2
 - /EAST, 7-1 to 7-2
 - /HEIGHT, 7-1 to 7-2
 - /LAYER, 7-1 to 7-2
 - /MAP_DESCRIPTOR, 7-1, 7-3
 - /MH_TYPE, 7-1, 7-3
 - /NORTH, 7-1, 7-3
 - /OUTPUT, 7-1, 7-3
 - /PRINTER, 7-1, 7-3
 - /REVISION_LEVEL, 7-1, 7-3
 - /SCAN, 7-1, 7-4
 - /SOUTH, 7-1, 7-4
 - /SYMBOLS, 7-1, 7-4
 - /TIMINGS, 7-1, 7-4
 - /WEST, 7-1, 7-4
- DESCRIPTION, 7-5
- effect of qualifiers on output, 7-5
- EXAMPLES, 7-9
- FORMAT, 7-1
- FUNCTION, 7-1
- information output, 7-1
- PARAMETER, 7-2
- PROMPT, 7-2
- ILAYFC, 1-7, 8-1
 - command qualifiers, 8-1 to 8-2
 - /CODES, 8-1 to 8-2
 - /IN_SITU, 8-1 to 8-2
 - /LOG, 8-1 to 8-2
- DESCRIPTION, 8-2
- EXAMPLES, 8-3
- FORMAT, 8-1
- FUNCTION, 8-1
- PARAMETERS, 8-1
- PROMPTS, 8-1
- IMEND, 1-4, 9-1

- and LASERAID, 9-3
- command qualifiers, 9-1
 - /RANGE, 9-1
 - /TRUNCATE, 9-1 to 9-2
- DESCRIPTION, 9-2
- EXAMPLES, 9-4
- FUNCTION, 9-1
- PARAMETER, 9-1
- processing action, 9-2
- PROMPT, 9-1
- reasons for use, 9-2
- IMEND and DAMP, 9-1
- IMERGE, 3-7, 10-1
 - and CP entry default, 10-1
 - and the IFF HI entry, 10-10
 - and the IFF MD entry, 10-8
 - COMMAND QUALIFIERS, 10-3
 - command qualifiers, 10-1
 - /CP_EXPANSION, 10-1 to 10-3
 - restrictions, 10-3
 - /DEBUG, 10-1, 10-3
 - /LAYER, 10-1 to 10-3
 - positional dependency, 10-3
 - restrictions, 10-4
 - /LOG, 10-1, 10-4
 - /OUTPUT, 10-2, 10-4
 - /REST_LAYER, 10-2, 10-4
 - /SINGLE_LAYER, 10-2, 10-5
 - /SPLIT, 10-2, 10-5
 - argument, 10-5
 - /UNIQUE_FSN, 10-2, 10-5
 - DESCRIPTION, 10-5
 - file bifurcation, 10-8
 - file concatenation, 10-7
 - file mergeing, 10-5
 - general, 10-5
 - layer selection, 10-8
- Example formats, 10-2
- EXAMPLES, 10-10
- FORMAT, 10-1
- FUNCTION, 10-1
- input file command qualifiers
 - /LAYER, 10-2
- output file command qualifiers
 - /CP_EXPANSION, 10-2
 - /REST_LAYER, 10-2
 - /SINGLE_LAYER, 10-2
- parameter defaults, 10-3
- parameter restrictions, 10-3
- PARAMETERS, 10-3
- PROMPT, 10-3
- rules for IFF MD useage, 10-9
- uses
 - file bifurcation, 10-1
 - file concatenation, 10-1
 - file mergeing, 10-1
 - layer selection, 10-1
 - when to use IMERGE, 10-1
- IMP
 - and \$STATUS, 1-1
 - and /IN_SITU mode, 1-6
 - restrictions, 1-6
 - and DCL symbol \$STATUS, 1-5
 - and default IFF file extension, 1-3
 - and file management, 1-3
 - and IFF, 1-3
 - and IFF HI entry, 1-4
 - and new IFF entries, 1-1
 - and origin offset, 1-3, 1-5
 - and origin offsets, 1-1
 - and the DAMP package., 1-1
 - and the SI utility, 1-3
 - and three dimensional strings, 1-5
 - as a part of LAMPS, 1-1
 - compatibility with old IFF files, 1-3
 - definition of acronym, 1-1
 - features, 1-1
 - functional advantages over DAMP, 1-2
 - functional correspondence with DAMP, 1-2
 - general IFF file opening messages, 1-7
 - getting started, 1-6
 - message definitions, 1-6
 - module functions
 - change feature code by layer, 8-1
 - coordinate transformation, 20-1
 - feature renumbering, 13-1
 - feature selection, 16-1
 - feature selection by AC, 15-1
 - file recovery, 9-1
 - file summary, 7-1
 - filtering, 4-1
 - geometry checking, 2-1
 - IFF area feature check and repair, 5-1
 - IFF conversion, 22-1
 - IFF entry recoding, 12-1
 - IFF entry report, 14-1
 - IFF file comparison, 3-1
 - IFF to text expansion, 19-1
 - interpolation, 4-1
 - IPATCH, 11-1
 - sorting, 17-1

- windowing, 21-1
- new IFF entries, 1-3
 - and downwards compatibility, 1-3
 - HI, 1-3
 - MD, 1-3
 - TS, 1-3
 - ZS, 1-3
- on-line help, 1-1
- opening IFF files, 1-7
- role within automated mapping, 1-1
- specification of IMP command
 - lines, 1-9
 - values set in \$STATUS, 1-5
- IMP and logical name LSL\$IF:, 1-3
- IMP package initialisation, 1-6
- IMPINI, 1-6, 18-5 to 18-6
 - DCL symbols, 1-6
 - in login file, 1-6
- Interpolation
 - Akima, 4-1, 4-8
 - Bezier, 4-1, 4-8
 - linear, 4-1
 - McConalogue, 4-1, 4-8
- IPATCH, 11-1
- !
 - see comments
 - comment delimiter, 11-5
- *
 - as a command, 11-8
 - as a wildcard, 11-5
- AC entry, 11-42
 - editing, 11-43
 - searching, 11-42
- and LSL\$IF:, 11-1
- angles
 - conversion of, 11-37, 11-39
- CB entry, 11-45
 - editing, 11-45
 - searching, 11-45
- CC entry, 11-47
 - editing, 11-47
 - searching, 11-47
- CH entry, 11-49
 - editing, 11-49
 - searching, 11-49
- * command, 11-8
- / command, 11-10
 - see edit mode
- ? command, 11-9
- command qualifiers, 11-1 to 11-2
 - /COMMANDS, 11-1 to 11-2
 - /FRT, 11-1 to 11-2
 - /JOURNAL, 11-1 to 11-2
 - /LOG, 11-1 to 11-2
 - /REVISION_LEVEL, 11-1, 11-3
 - /STATUS, 11-1, 11-3
 - /WRITE, 11-1, 11-3
- comments, 11-5
 - delimiter, 11-5
 - journalling of, 11-6
- CP entry, 11-51
 - editing, 11-51
 - searching, 11-51
- CS entry, 11-52
 - editing, 11-52
 - searching, 11-52
- <CTRL/Z>
 - see EXIT command
- dangers of use, 11-4
- default file-specs
 - command file, 11-2
 - FRT file, 11-2
 - IFF file, 11-1
 - journal file, 11-2
- default mode, 11-1
- DELETE command, 11-12
 - and marks, 11-12
- DELETE MARK command, 11-13
 - and marks, 11-13
- description, 11-3
- display of values, 11-41
- DISPLAY OFF command, 11-14
 - and NEXT, 11-14
- DISPLAY ON command, 11-15
 - and NEXT, 11-15
- edit mode, 11-10
 - see / command
 - command mode, 11-11
 - forms of, 11-10
 - line mode, 11-10
 - prompt mode, 11-10
- editing
 - AC entry, 11-43
 - CB entry, 11-45
 - CC entry, 11-47
 - CH entry, 11-49
 - CP entry, 11-51
 - CS entry, 11-52
 - FS entry, 11-57
 - MH entry, 11-64
 - NF entry, 11-66
 - NO entry, 11-68
 - NS entry, 11-70
 - RO entry, 11-74
 - SL entry, 11-76
 - SS entry, 11-77
 - ST entry, 11-78

- TC entry, 11-80
- TH entry, 11-82
- TS entry, 11-84
- TX entry, 11-86
- ZS entry, 11-90
- EF entry, 11-53
 - searching, 11-53
- EJ entry, 11-54
 - searching, 11-54
- EM entry, 11-55
 - searching, 11-55
- EO entry, 11-56
 - searching, 11-56
- EOF command, 11-16
 - and PREVIOUS, 11-5, 11-16
- example session, 11-92
- EXIT command, 11-17
- EXPLAIN command, 11-18
- fatal messages, 11-118
- 'feature' and NFs, 11-4
- file scanning, 11-4
- FIND command, 11-19
 - see also file scanning
 - and NEXT, 11-19
 - and PREVIOUS, 11-5
 - problems with, 11-19
- FRT command, 11-21
- FS entry, 11-57
 - editing, 11-57
 - searching, 11-57
- function, 11-1
- HELP command, 11-22
- HI entry, 11-60
 - searching, 11-60
- informational messages, 11-97
- integers
 - conversion of, 11-36, 11-38
- JB entry, 11-61
 - searching, 11-61
- journalling of <CTRL/Z>, 11-2
- JP entry, 11-62
 - searching, 11-62
- limitations, 11-3
- LINE command, 11-23
- LIST command, 11-24
- MARK command, 11-25
 - see also DELETE
 - see also DELETE MARK
 - see also RETURN
- MD entry, 11-63
 - and ITRANS, 11-63
 - searching, 11-63
- messages
 - fatal, 11-118
 - informational, 11-97
 - other, 11-119
 - warning, 11-100
- MH entry, 11-64
 - editing, 11-64
 - searching, 11-64
- NEXT comamnd
 - see also PREVIOUS
- NEXT command, 11-26
 - see also DISPLAY ON
 - see also file scanning
 - and PREVIOUS, 11-4
- NF entry, 11-66
 - editing, 11-66
 - searching, 11-66
- NO entry, 11-68
 - editing, 11-68
 - searching, 11-68
- NS entry, 11-70
 - editing, 11-70
 - searching, 11-70
- other messages, 11-119
- parameter, 11-1
- PREVIOUS command, 11-28
 - see also file scanning
 - how it works, 11-4
- print current entry
 - see ? command
- prompts, 11-1
- RA entry, 11-72
 - searching, 11-72
- RETURN command, 11-29
 - see also MARK
- REVISION_LEVEL command, 11-30
- rewind file
 - see * command
- RO entry, 11-74
 - editing, 11-74
 - searching, 11-74
- searching, 11-6
 - AC entry, 11-42
 - CB entry, 11-45
 - CC entry, 11-47
 - CH entry, 11-49
 - CP entry, 11-51
 - CS entry, 11-52
 - EF entry, 11-53
 - EJ entry, 11-54
 - EM entry, 11-55
 - EO entry, 11-56
 - FS entry, 11-57
 - HI entry, 11-60
 - JB entry, 11-61
 - JP entry, 11-62
 - MD entry, 11-63
 - MH entry, 11-64

- NF entry, 11-66
- NO entry, 11-68
- NS entry, 11-70
- RA entry, 11-72
- RO entry, 11-74
- SH entry, 11-75
- SL entry, 11-76
- SS entry, 11-77
- ST entry, 11-78
- TC entry, 11-80
- TH entry, 11-82
- TS entry, 11-84
- TX entry, 11-86
- VO entry, 11-88
- ZS entry, 11-90
- SH entry, 11-75
 - searching, 11-75
- SL entry, 11-76
 - editing, 11-76
 - searching, 11-76
- SS entry, 11-77
 - editing, 11-77
 - searching, 11-77
- ST entry, 11-78
 - editing, 11-78
 - searching, 11-78
- STATUS command, 11-31
 - see also STATUS ON
- status lines, 11-3, 11-31 to 11-33
- STATUS OFF command, 11-32
- STATUS ON command, 11-3, 11-33
- SYMBOL command, 11-34
- TC entry, 11-80
 - editing, 11-80
 - searching, 11-80
- TEXT command, 11-35
- TH entry, 11-82
 - editing, 11-82
 - searching, 11-82
- TO DECIMAL command, 11-36
- TO DEGREES command, 11-37
- TO HEXADECIMAL command, 11-38
- TO RADIANS command, 11-39
- TS entry, 11-84
 - editing, 11-84
 - searching, 11-84
- TX entry, 11-86
 - editing, 11-86
 - searching, 11-86
- TYPE command, 11-40
 - and PREVIOUS, 11-4
- VO entry, 11-88
 - searching, 11-88
- warning messages, 11-100
- wildcards, 11-5
- XX entry, 11-89
- ZS entry, 11-90
 - editing, 11-90
 - searching, 11-90
- IRECODE, 1-7, 12-1
 - and ancillary codes, 12-5
 - and feature codes, 12-5
 - and layers, 12-5
 - and text component codes, 12-5
 - Command format, 12-1
 - command qualifiers, 12-1 to 12-2
 - /AC, 12-1, 12-3
 - WARNING, 12-3
 - /ALPHABETIC, 12-1 to 12-2
 - /FC, 12-1, 12-4
 - /IN_SITU, 12-1, 12-4
 - /LAYER, 12-1, 12-4
 - /LOG, 12-1, 12-5
 - /TCC, 12-1, 12-4
 - description, 12-5
 - examples, 12-8
 - implications of use, 12-6
 - on ACs, 12-6
 - on FCs, 12-6
 - on layers, 12-6
 - on TCCs, 12-7
 - Parameters, 12-2
 - Prompts, 12-2
- IRENUMBER, 1-7, 13-1
 - and FSN 0, 13-3
 - /BASE
 - a warning, 13-3
 - COMMAND QUALIFIERS, 13-2
 - command qualifiers, 13-1
 - /BASE, 13-1 to 13-2
 - /IN_SITU, 13-1 to 13-2
 - /INCREMENT, 13-1 to 13-2
 - /LOG, 13-1 to 13-2
 - /LOWER_LIMIT, 13-1 to 13-2
 - /UPPER_LIMIT, 13-1 to 13-2
 - default action, 13-3
 - if no command qualifiers, 13-3
 - DESCRIPTION, 13-3
 - EXAMPLES, 13-4
 - FORMAT, 13-1
 - FUNCTION, 13-1
 - PARAMETERS, 13-1
 - PROMPTS, 13-1
- IREPORT, 14-1
 - and LITES2, 14-11
 - and LITES2, 14-1, 14-3
 - and LSL\$IF:, 14-2

- COMMAND QUALIFIERS, 14-2
- command qualifiers, 14-1
 - /ABSENT, 14-1 to 14-2
 - /ADDRESS, 14-1 to 14-2
 - /FRT, 14-1 to 14-2
 - IFF entry level, 14-1
 - /IN_LAYER, 14-1 to 14-2
 - restrictions, 14-2
 - /LITES2, 14-1 to 14-2
 - restrictions, 14-3
 - /LOG, 14-1
 - /OUTPUT, 14-1, 14-3
 - /PRINTER, 14-1, 14-4
 - /REVISIONLEVEL, 14-1, 14-4
 - /WITH_FC, 14-1, 14-4
 - restrictions, 14-4
- default action, 14-1
- default output, 14-11
- DESCRIPTION, 14-11
- EXAMPLES, 14-12
- FORMAT, 14-1
- FUNCTION, 14-1
- IFF entry command qualifiers, 14-4
- IFF entry qualifiers, 14-1
 - /AC, 14-1, 14-4
 - /CB, 14-1, 14-5
 - /CC, 14-1, 14-5
 - /CP, 14-1, 14-6
 - /FC, 14-1, 14-6
 - /LAYER, 14-1, 14-6
 - /NF, 14-1, 14-7
 - /NS, 14-1, 14-7
 - /RO, 14-1, 14-7
 - /ST, 14-1, 14-7
 - /TC, 14-1, 14-8
 - /TH, 14-1, 14-8
 - /TS, 14-1, 14-9
 - /TX, 14-1
 - /ZS, 14-1, 14-9
- output to printer, 14-11
- PARAMETER, 14-2
- PROMPT, 14-2
- RESTRICTIONS, 14-10
- ISELAC, 15-1
 - AND command, 15-10
 - and structure of IFF AC entries, 15-1
 - ? command, 15-24
- COMMAND QUALIFIER, 15-1
- commands
 - ?, 15-24
 - AND, 15-10
 - EXIT, 15-11
 - FROM, 15-12
 - FSN, 15-13
 - GO, 15-15
 - HELP, 15-16
 - LAYER, 15-17
 - RESET, 15-19
 - SHOW, 15-20
 - SKELETON, 15-21
 - TO, 15-23
- comments in skeleton files, 15-4
- default skeleton file, 15-2
- DESCRIPTION, 15-1
 - abbreviations, 15-8
 - General, 15-1
 - IFF AC entries, 15-1
 - selecting AC names, 15-6
 - selecting attribute names, 15-6
 - skeleton files, 15-2
 - description, 15-2
- example skeleton file, 15-2
- EXAMPLES, 15-25
- EXIT command, 15-11
- FORMAT, 15-1
- FROM command, 15-12
- FSN command, 15-13
- FUNCTION, 15-1
- GO command, 15-15
- HELP command, 15-16
- LAYER command, 15-17
- prompt, 15-1
- RESET command, 15-19
- selecting AC names or attribute names, 15-6
- SHOW command, 15-20
- SKELETON command, 15-21
- skeleton file
 - comment delimiter, 15-4
 - default, 15-2
 - example, 15-2
 - format, 15-4
 - skeleton files, 15-1 to 15-2
- TO command, 15-23
- ISELECT, 16-1
 - and IFF HI entry, 16-5
 - and IFF layer 0, 16-5
 - and LSL\$IF:, 16-2
 - and selection failure, 16-1, 16-4
 - command format, 16-1
 - command qualifiers, 16-1 to 16-2
 - /AC, 16-1 to 16-2
 - /EXCLUDE_FC, 16-1 to 16-2
 - /FC, 16-1 to 16-2

- /FSN, 16-1, 16-3
- /INTEGER_HEIGHT, 16-1, 16-3
- /LAYER, 16-1, 16-3
- /LOG, 16-1, 16-3
- /MODULUS, 16-1, 16-3
- /REAL_HEIGHT, 16-1, 16-3
 - example, 16-3 to 16-4
- /TCC, 16-1, 16-4
- default file-specs
 - input file, 16-2
 - output file, 16-2
- description, 16-4
- examples, 16-5
- messages (error), 16-9
- messages (informational), 16-8
- messages (other), 16-12
- parameters, 16-2
- prompts, 16-2
- restrictions, 16-4
- ISORT, 17-1
 - and FSN 0, 17-3
 - and LITES2, 17-3
- COMMAND QUALIFIERS, 17-2
- command qualifiers, 17-1
 - /DUPLICATES, 17-1 to 17-2
 - /FC, 17-1 to 17-2
 - /INTERNAL, 17-1 to 17-2
 - /LOG, 17-1 to 17-2
 - /NODUPLICATES=FIRST, 17-1 to 17-2
 - /NODUPLICATES=LAST, 17-1 to 17-2
 - /OUTPUT, 17-1 to 17-2
 - /PRINTER, 17-1 to 17-2
- DESCRIPTION, 17-3
- EXAMPLES, 17-3
- FORMAT, 17-1
- PARAMETERS, 17-1
- PROMPTS, 17-1
- ISTART, 18-1
 - and /LASERAID qualifier
 - restrictions, 18-12
 - and direct table reads, 18-4
 - and FRT, 18-2, 18-4
 - and IMPINI, 18-5 to 18-6
 - and LASERAID, 18-2, 18-12
 - available modes, 18-12
 - IFF differences, 18-12
 - and TABLE_MONITOR, 18-4
- COMMAND QUALIFIERS, 18-2
- Command qualifiers, 18-1
 - /ADDRESSABILITY, 18-2, 18-5
 - /BUTTONS, 18-2, 18-6
 - /CENTRE, 18-2, 18-6
 - /CHECK, 18-1 to 18-2
 - /COUNTS, 18-2, 18-6
 - Digitising, 18-1, 18-5
 - /ADDRESSABILITY, 18-5
 - /BUTTONS, 18-6
 - /CENTRE, 18-6
 - /COUNTS, 18-6
 - /REPEAT, 18-6
 - /ROTATION, 18-7
 - /THOU, 18-7
 - /FC, 18-1 to 18-2
 - /LASERAID, 18-1 to 18-2
 - /LAYER, 18-1, 18-3
 - /LEARNER, 18-1, 18-3
 - /LOG, 18-1, 18-3
 - /MCE, 18-1, 18-3
 - /MH_TYPE, 18-1, 18-3
 - /MUART_TABLE, 18-1, 18-3
 - /OS, 18-1, 18-4
 - /REPEAT, 18-2, 18-6
 - /ROTATION, 18-2, 18-7
 - /SYMBOLS, 18-1, 18-4
 - /TABLE, 18-1, 18-4
 - /THOU, 18-2, 18-7
 - /TICKS, 18-1, 18-4
 - /USER_ORIGIN, 18-1, 18-5
 - restriction, 18-5
- default mode of operation, 18-8
- defining DCL symbols, 18-5 to 18-6
- DESCRIPTION, 18-8
 - digitising table
 - characteristics, 18-13
 - digitising table mode, 18-10
 - functions, 18-12
 - General, 18-8
 - ISTART and LASERAID, 18-12
 - logical names required, 18-11
 - OS mode, 18-9
 - template mode, 18-9
- digitising table mode, 18-1, 18-5, 18-8, 18-10
- Digitising table qualifiers, 18-5
- EXAMPLES, 18-13
- Function, 18-1
- functions, 18-12
- learner help, 18-12
- Logical names, 18-11
 - LSL\$MONITOR_TABLE, 18-11
 - LSL\$TK, 18-11
- modes of operation, 18-1
- Ordnance Survey BGI, 18-4
- OS mode, 18-1, 18-9
- Parameter, 18-2
- Prompt, 18-2

- RESTRICTIONS, 18-7
- table options, 18-1
 - direct table reads, 18-1, 18-8
 - MAURT, 18-8
 - MUART, 18-1
 - TABLE MONITOR, 18-1, 18-8
- template mode, 18-1, 18-9
- ITOTEXT, 19-1
 - and IFF junction entries, 19-8
 - and IFROMTEXT, 19-3
 - warning about /BRIEF, 19-3
 - and non-graphic edits, 19-9
- COMMAND QUALIFIERS, 19-2
- command qualifiers, 19-1
 - /ABSOLUTE, 19-1 to 19-2
 - /ADDRESS, 19-1 to 19-2
 - /BRIEF, 19-1, 19-3
 - /CUSTOMER, 19-1, 19-3
 - /DECIMAL_PLACES, 19-1, 19-3
 - /FRT, 19-1, 19-3
 - /KEEP, 19-1, 19-3
 - /LOG, 19-1, 19-3
 - /MH, 19-1, 19-4
 - /PRINTER, 19-1, 19-4
 - /QUOTED_TEXT, 19-1, 19-4
 - /REVISION_LEVEL, 19-1, 19-4
 - /WIDTH, 19-1, 19-4
- default file-spec extensions, 19-7
- DESCRIPTION, 19-5
- effect of /CUSTOMER and /NOMH qualifiers, 19-7
- example output, 19-9
- EXAMPLES, 19-12
- FORMAT, 19-1
- FUNCTION, 19-1
- intended use, 19-5
- ITOTEXT and the /ABSOLUTE qualifier, 19-8
- ITOTEXT and the /BRIEF qualifier, 19-7
- ITOTEXT and the HI entry., 19-5
- ITOTEXT and the MD entry., 19-6
- ITOTEXT and the MH entry., 19-5
- output options, 19-1
- PARAMETERS, 19-2
- PROMPTS, 19-2
- RESTRICTIONS, 19-4
- ITRANS, 1-5, 20-1
 - and Lat-Long coordinates, 20-13
 - and the MD entry, 20-8
 - incompatibility with DAMP IPR, 20-8
 - available spheroids, 20-9
 - carrying out datum shifts, 20-14
 - carrying out projection transformations, 20-12
- COMMAND QUALIFIERS, 20-3
- command qualifiers, 20-1
 - /ABSOLUTE, 20-1, 20-3
 - /CONTROL, 20-1, 20-3
 - /DATUM_SHIFT, 20-1, 20-4, 20-14 to 20-15
 - /DESCRIPTOR, 20-1, 20-4
 - a warning, 20-4
 - /FOUR, 20-1, 20-5
 - /HEIGHT_ADJUST, 20-1, 20-5
 - /LEARNER, 20-1, 20-5
 - /LOG, 20-1, 20-5
 - /OUTPUT, 20-1, 20-6
 - /POINT, 20-1, 20-6
 - /PROJECTIVE, 20-1, 20-6
 - /RETAIN_CP, 20-1, 20-6
 - /SIX, 20-1, 20-7
 - /THREE, 20-1, 20-7
 - /TWO, 20-1, 20-7
- datum shift calculations on individual points, 20-20
- datum shifts, 20-14
- DESCRIPTION, 20-8
 - Carrying out datum shifts, 20-14
 - Carrying out empirical transformations, 20-16
 - Carrying out projection transformations, 20-12
 - datum shift calculations on individual points, 20-20
 - projection calculations on individual points, 20-20
 - the MD entry, 20-8
- Editing the map descriptor, 20-21
- empirical transformations, 20-16
- EXAMPLES, 20-22
- FORMAT, 20-1
- FUNCTION, 20-1
- geodetic datum, 20-9, 20-14
 - origin, 20-14
- IFF MD entry
 - contents, 20-8
- PARAMETERS, 20-2
- projection calculations on individual points, 20-20
- projection transformations, 20-12
- PROMPTS, 20-1

- RESTRICTIONS, 20-2
 - spheroid codes, 20-9
 - see Appendix A
 - spheroid shape, 20-9
 - spheroid size, 20-9
 - spheroids, 20-9
 - supported measurement units, 20-12
 - supported projections, 20-10 to 20-11
 - see Appendix A
 - supported spheroids, 20-9
- IWINDOW, 1-5, 21-1
 - and 3d strings, 21-8
 - and CB entries, 21-8
 - and FRT, 21-4
 - and reentrant features, 21-8
- COMMAND QUALIFIERS, 21-3
- command qualifiers, 21-1
 - /ABSOLUTE, 21-1, 21-3
 - /AREA, 21-1, 21-3
 - /CLIP_GRID, 21-1, 21-6
 - /CONTROL_POINTS, 21-1, 21-3
 - /DEBUG, 21-1, 21-3
 - /EAST, 21-1, 21-3
 - /FC_IGNORE, 21-1, 21-3
 - /FRT, 21-1, 21-4
 - /FSN_BY_LAYER, 21-1, 21-4
 - /KEEP_TEXT, 21-1, 21-6
 - /LAYER_IGNORE, 21-1, 21-4
 - /LINES2, 21-1
 - /LITES2, 21-4
 - /LOG, 21-1, 21-4
 - /MAP_HEADER, 21-1, 21-5
 - /NORTH, 21-1, 21-5
 - /OFFSET, 21-1, 21-5
 - /ORIGIN, 21-1, 21-5
 - /OS, 21-1, 21-5
 - /OUTPUT, 21-1, 21-5
 - /RESET_CP, 21-1
 - /RETAIN_FSN, 21-1, 21-6
 - /SOUTH, 21-1, 21-6
 - /VOID, 21-1, 21-6
 - /WEST, 21-1, 21-6
- command qualifiers in OS mode, 21-1
- command qualifliers
 - /RESET_CP, 21-6
- default action, 21-7
- DESCRIPTION, 21-6
- EXAMPLES, 21-10
- FUNCTION, 21-1
- map header option, 21-7
- OS mode command qualifiers, 21-6
 - /CLIP_GRID, 21-1, 21-6
 - /KEEP_TEXT, 21-1, 21-6
- PARAMETERS, 21-2
- PROGRAM OPERATION, 21-8
- PROMPTS, 21-2
- RESTRICTONS, 21-2
- LAMPS, 1-1
- LAPROCESS, 3-7
- LASERAID, 3-7, 18-2, 18-12
 - and ISTART, 18-12
- LAT-LONG and ITRANS, 20-13
- Linear interpolation, 4-1
- LITES2, 2-1 to 2-2, 2-4, 2-9, 3-7, 14-1, 14-3, 14-11, 17-3
 - and IREPORT, 14-1, 14-3
 - and ISORT, 17-3
- LITES2 and IREPORT, 14-11
- logical name LSL\$IF:, 1-3
- Logical names
 - ISTART, 18-11
 - LSL\$MONITOR_TABLE, 18-11
 - LSL\$TK, 18-11
- LSL\$IF:, 1-3, 1-6
 - and IMP, 1-3
- McConalogue interpolation, 4-1
- MD entry, 20-8
 - and ITRANS, 20-8
- MUART, 18-3
- MUART (Multi-UART), 18-1
- Origin offset, 1-3, 10-8
- OSCLIP, 21-1
- Registration marks, 18-2
- Selection of IFF features, 16-1
- SI
 - example, 1-6
- SI command, 1-6
- SI utility, 1-3
- \$STATUS, 1-1, 1-5 to 1-6