

AVEVA ImPLANT-I

Installation and User Guide

Disclaimer

Information of a technical nature, and particulars of the product and its use, is given by AVEVA Solutions Ltd and its subsidiaries without warranty. AVEVA Solutions Ltd and its subsidiaries disclaim any and all warranties and conditions, expressed or implied, to the fullest extent permitted by law.

Neither the author nor AVEVA Solutions Ltd, or any of its subsidiaries, shall be liable to any person or entity for any actions, claims, loss or damage arising from the use or possession of any information, particulars, or errors in this publication, or any incorrect use of the product, whatsoever.

Copyright

Copyright and all other intellectual property rights in this manual and the associated software, and every part of it (including source code, object code, any data contained in it, the manual and any other documentation supplied with it) belongs to AVEVA Solutions Ltd or its subsidiaries.

All other rights are reserved to AVEVA Solutions Ltd and its subsidiaries. The information contained in this document is commercially sensitive, and shall not be copied, reproduced, stored in a retrieval system, or transmitted without the prior written permission of AVEVA Solutions Ltd. Where such permission is granted, it expressly requires that this Disclaimer and Copyright notice is prominently displayed at the beginning of every copy that is made.

The manual and associated documentation may not be adapted, reproduced, or copied, in any material or electronic form, without the prior written permission of AVEVA Solutions Ltd. The user may also not reverse engineer, decompile, copy, or adapt the associated software. Neither the whole, nor part of the product described in this publication may be incorporated into any third-party software, product, machine, or system without the prior written permission of AVEVA Solutions Ltd, save as permitted by law. Any such unauthorised action is strictly prohibited, and may give rise to civil liabilities and criminal prosecution.

The AVEVA products described in this guide are to be installed and operated strictly in accordance with the terms and conditions of the respective licence agreements, and in accordance with the relevant User Documentation. Unauthorised or unlicensed use of the product is strictly prohibited.

First published June 2008

© AVEVA Solutions Ltd, and its subsidiaries 2008

AVEVA Solutions Ltd, High Cross, Madingley Road, Cambridge, CB3 0HB, United Kingdom

Trademarks

AVEVA and Tribon are registered trademarks of AVEVA Solutions Ltd or its subsidiaries. Unauthorised use of the AVEVA or Tribon trademarks is strictly forbidden.

AVEVA product names are trademarks or registered trademarks of AVEVA Solutions Ltd or its subsidiaries, registered in the UK, Europe and other countries (worldwide).

The copyright, trade mark rights, or other intellectual property rights in any other product, its name or logo belongs to its respective owner.

AVEVA ImPLANT-I Installation and User Guide

Contents	Page
Introduction	1:1
How to Use This Manual	1:1
Input to ImPLANT-I.	1:2
Output From ImPLANT-I	1:2
Modes of Operation	1:2
Installation	2:1
Basic Installation	2:1
.NET Framework Installation.	2:1
Flexman License Manager	2:2
Running in a Command Window	2:2
Electronic Manuals	2:2
Testing the Installation	2:2
Test Files	2:3
Running the Tests	2:3
Checking the Macro Files.	2:3
Checking the Review Model Files	2:3
ImPLANT-I Dialog.	3:1
Overview	3:1
Graphical User Interface	3:2
Adding the Input Files.	3:2
Setting the Units.	3:2
Setting the Output File	3:2

Output Messages, Input Command Files and Specifying Extra Commands	3:2
Defaults and Persistence of Settings.	3:2
Using ImPLANT-I via the Command Processor Interface	4:1
Overview	4:1
Default Operation.	4:2
Units in ImPLANT-I.	4:2
Example ImPLANT-I Session	4:3
Notes on the Example Session	4:5
Steel Structures	4:7
ImPLANT-I Profile Macro	4:7
ADDSTEEL Command.	4:8
REMSTEEL Command	4:9
REPORTSTEEL Command	4:9
STEELFILTER Command	4:9
Recognising Profiles.	4:9
ISFF Extensions	4:10
ImPLANT-I via the Command Line.	5:1
Command Line Arguments	5:1
Limitations, Warnings and Error Messages	6:1
Geometry Transfer Limitations.	6:1
Using MicroStation to Improve the Translation	6:2
Specific Review or PDMS/Outfitting Limitations.	6:2
Intelligent Transfer Limitations	6:3
Not in the Scope of ImPLANT-I	6:3
Other Limitations	6:3
Warning Messages.	6:3
Error Messages	6:5
Other Messages	6:5
Appendix: Command Syntax	A:1
Appendix: Attribute Translation	B:1

1 Introduction

AVEVA ImPLANT-I is a translator for exporting Intergraph's 3D graphics (held in the Intergraph Standard File Format, or ISFF), as used by MicroStation (.dgn), to either AVEVA PDMS or AVEVA Marine Outfitting via DESIGN macro files, or to Review via model files. It can also translate any associated Intergraph DESIGN Review (.drv) files into the AVEVA Attribute File Format (ASCII), which can then be accessed from within Review.

Before you can use AVEVA ImPLANT-I, it must have been installed and tested, as described in [Installation](#).

Note: ImPLANT-I Version 1.6 does not handle MicroStation DGN V8 files. This feature is planned to be included in a future release. Refer to [Geometry Transfer Limitations](#) for details of other restrictions on what DGN objects can be translated, and [Using MicroStation to Improve the Translation](#) for advice on how to work round some of the limitations.

1.1 How to Use This Manual

[Introduction](#), is the chapter you are reading now, and tells you what AVEVA ImPLANT-I, does and some of its background details.

[Installation](#), provides the installation instructions.

[ImPLANT-I Dialog](#), tells you how to use AVEVA ImPLANT-I interactively.

[Using ImPLANT-I via the Command Processor Interface](#), tells you how you to use AVEVA ImPLANT-I with typed commands.

[ImPLANT-I via the Command Line](#), provides a brief description of how to call AVEVA ImPLANT-I as a batch process.

[Limitations, Warnings and Error Messages](#), tells you about the limitations of AVEVA ImPLANT-I, how you can work round some of them, and explains the warning and error messages you may see when using AVEVA ImPLANT-I.

Appendix A is the [Appendix: Command Syntax](#) section.

Appendix B, [Appendix: Attribute Translation](#), describes the file format required for translating attributes.

1.2 Input to ImPLANT-I

The input to ImPLANT-I consists of ISFF DESIGN Files, which are generated from MicroStation or compatible products. ISFF files are binary files, which normally have the file extension **.dgn**.

Note: ImPLANT-I Version 1.6 does not handle MicroStation DGN V8 files. This feature is planned to be included in a future release. Refer to [Geometry Transfer Limitations](#) for details of other restrictions on what DGN objects can be translated, and [Using MicroStation to Improve the Translation](#) for advice on how to work round some of the limitations.

The interface works in read-only mode and so ISFF data will not be at risk.

1.3 Output From ImPLANT-I

The output of the program is either an ASCII macro file (suitable for input to PDMS or Outfitting), as or a Review model file. No databases are created in PDMS.

If attributes associated with the design files are to be output to Review in AVEVA Attribute File Format, then the associated DESIGN Review files (**.drv**) attribute files must be present.

1.4 Modes of Operation

ImPLANT-I can be run in three ways:

- The Graphical User Interface (Section 3). This provides a Windows-based graphical interface to set up everything required for a translation. This then calls into the command processor interface with the relevant settings applied automatically. This is the recommended approach for new or occasional users.
- The Command Processor Interface (Section 4). This is the command window interface that appears when running ImPLANT-I.exe. It is the traditional mode of working.
- Using the command line (Section 5). An input macro file (see section 4.5.1) can be specified in the command line arguments containing a list of everything that needs to be done. This approach is often used when automating translation tasks.

2 Installation

2.1 Basic Installation

ImPLANT-I is supplied on DVD or CD. The disk contains the installer, and an installable version of Microsoft's .NET 2.0. Also on the disk is a folder containing the Installation instructions and user documentation, and it is recommended that you carefully read all of the Installation instructions before installing the software.

To install ImPLANT-I first insert the disk in your Workstation's DVD drive, click for Contents, then select a product, and then follow the Installer instructions.

If the disk does not start, click **Start>Run**, and then browse for the Start.htm file.

During the installation sequence, follow the on-screen instructions as they appear. You will be given a choice between two levels of installation:

- **Standard installation:**
Installs all available product components in the default location.
- **Advanced installation:**
Presents you with a list of all available product components from which you can select those to be installed, and allows you to choose the Installation folder.

It is recommended that you use the default settings for folder paths etc. unless you have good reasons for doing otherwise.

Note: The default installation disk is the largest disk, and this is not always appropriate. This can be checked and if necessary changed by clicking the **Advanced** button on the installer.

The process needs at least 7Mb of free disk space for a full installation of all options.

When the process is complete, select **Start>All Programs>AVEVA>AVEVA ImPLANT-I1.6** to reach the program and this documentation.

Note: If the installer detects an existing ImPLANT-I installation, it will display a set of options for modifying/repairing/removing these existing files in place of the standard installation options.

2.2 .NET Framework Installation

ImPLANT-I requires that Microsoft's .NET Framework 2.0 be installed. If the installer is started from setup.exe, then the .NET Framework is installed automatically, if required.

2.3 Flexman License Manager

You must install and set up the Flexman license system before launching the installed software. This will usually be done only on the server. Please see the *Flexman 3.1.SP1 Installation and Configuration Guide*. News and update information is also available for Flexman on the AVEVA support web-site.

2.4 Running in a Command Window

In order to make ImPLANT-I available as a command in a command window, the ImPLANT-I installation folder may be added to the **PATH** environment variable. For detailed instructions on how to do this refer to Windows Help.

2.5 Electronic Manuals

Electronic versions of the ImPLANT-I Manuals are available on the CD-ROM/DVD as **.pdf** files. In order to display these Manuals you must have the Adobe Acrobat Reader software installed on your PC. This software is also included on the CD-ROM/DVD. Installing this software will enable you to view and print the ImPLANT-I Manuals.

2.6 Testing the Installation

The following tests can be performed after installing ImPLANT-I to make sure that the installation is complete and correct. You will find all the required testing material below the test subfolder of the installation folder (by default C:\AVEVA\ImPLANT-I1.6).

There are three **.dgn** files for three models: **tank1**, **tank2** and **tank3**.

The tests described below create three Review 2 model files and three PDMS/Outfitting DESIGN macro files. When running the tests, certain warning messages will be generated. These have been recorded in a file called **test_report.txt**, which is provided for comparison, e.g.

Warning - Line Type Not Implemented

Element 4 at File position 2048

Warning - Gap in loop of projected surface or surface of revolution

```
[ -125814.785,    74066.400,    4419.600] to
[ -125809.391,    74071.823,    4419.600] differences are:
[      -5.394,      -5.423,      0.000]
```

Element 1027 at File position 226062

Warning - Complex String Type Not Implemented

Element 1432 at File position 434790

These warnings are due to known limitations and geometrical oddities in the supplied DGN files. (Refer to [Limitations, Warnings and Error Messages](#)).

2.6.1 Test Files

To test ImPLANT-I export to PDMS/Outfitting and Review the following scripts are provided:

- **test_pdms.bat** (which uses the ImPLANT-I command file E1_pdms.txt and the sample DGN files tank1.dgn, tank2.dgn and tank3.dgn. The script creates tank1.mac, tank2.mac, tank3.mac and E1_pdms_log.txt)
- **test_review.bat** (which uses the ImPLANT-I command file E1_review.txt and the sample DGN files tank1.dgn, tank2.dgn and tank3.dgn. The script creates tank1.model, tank2.model, tank3.model and E1_review_log.txt)

To import the PDMS DESIGN macro files to PDMS the following macro is provided:

- **input_test_to_PDMS.txt**

2.6.2 Running the Tests

Shortcuts to run the PDMS and Review tests are created in the AVEVA ImPLANT-I program-group accessed from the start-menu (**Start>All Programs>AVEVA>AVEVA ImPLANT-I 1.6**).

Select **Test for PDMS** to create the PDMS macro files **tank1.mac**, **tank2.mac** and **tank3.mac**, together with the log file **E1_pdms_log.txt**, in the test sub-folder of the AVEVA ImPLANT-I installation folder (by default **C:\AVEVA\ImPLANT-I1.6**).

Select **Test for Review** to create the PDMS macro files **tank1.model**, **tank2.model** and **tank3.model**, together with the log file **E1_review_log.txt**, in the test sub-folder of the AVEVA ImPLANT-I installation folder (by default **C:\AVEVA\AVEVA ImPLANT-I1.6**).

To compare the warnings issued with those expected compare **E1_pdms_log.txt** and **E1_review_log.txt** with the relevant parts of **test_report.txt**.

2.6.3 Checking the Macro Files

- Enter the MONITOR module of PDMS or Outfitting
- Select the MDB. If this test has been run before, ensure that site **/test_implanti** is deleted before continuing.
- From MONITOR type:

```
$m /input_test_to_PDMS.txt
```
- Enter the DESIGN module of PDMS or Outfitting and visually check that the newly created site **/test_implanti** corresponds to the constituent **.dgn files** (as displayed in MicroStation).

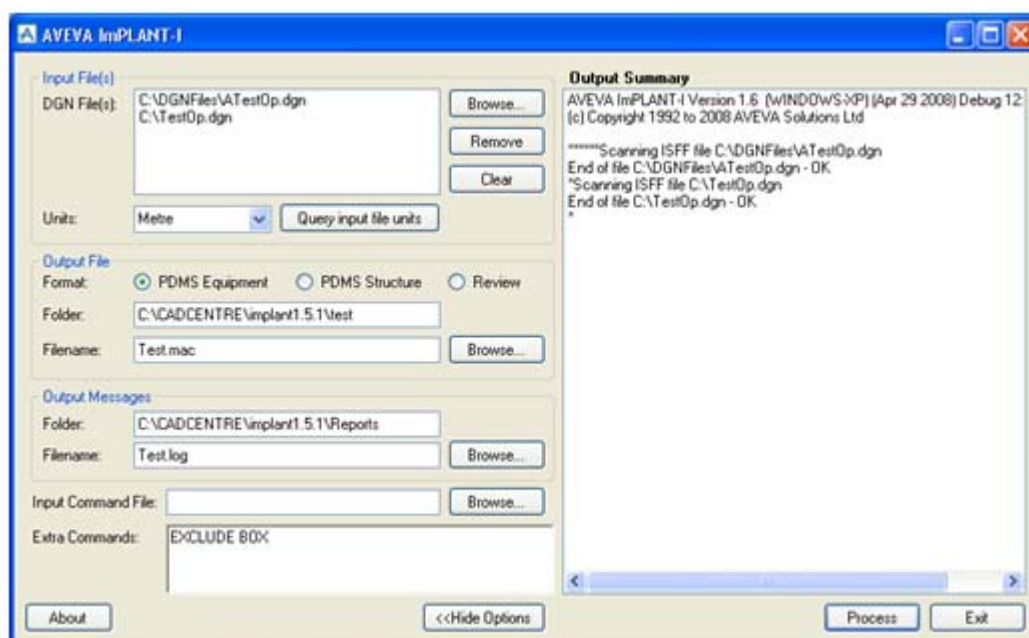
2.6.4 Checking the Review Model Files

Display **tank1.model**, **tank2.model** and **tank3.model** in Review and visually check against the corresponding **.dgn** models as displayed in **MicroStation**.

3 ImPLANT-I Dialog

3.1 Overview

For users who prefer using a graphical interface to a command-line, or only need to perform a limited number of translations, there is a separate executable `ImPLANT_I_ui.exe` that may be used to collect all the information needed and automatically run an ImPLANT-I session.



This program is started by the **ImPLANT-I Dialog** start menu item.

In order to use the GUI effectively, it is still necessary to understand how files are processed via the Command Processor so it is strongly recommended to also read Section 4 of this manual as well, even if you only intend to use the ImPLANT-I GUI to perform the translations.

The GUI can process the full set of commands that ImPLANT-I is capable of handling via the command line processor.

3.2 Graphical User Interface

3.2.1 Adding the Input Files

It is first necessary to choose the DGN files that are to be translated. Use the **Browse** button in the Input File(s) section to select all the files to be processed. Use the **Remove** button to remove any selected files from the DGN File(s) list and the **Clear** button to clear the list completely.

3.2.2 Setting the Units

Use the Units dropdown menu to select which units are to be used for the processing. Use the **Query input file units** button to display the units specified in any of the input DGN files listed.

3.2.3 Setting the Output File

First, select the output format needed; PDMS Equipment, PDMS Volume, PDMS Structure or Review, then use the **Browse** button to choose the output filename.

3.2.4 Output Messages, Input Command Files and Specifying Extra Commands

Clicking on the **Show Options** button displays the optional settings.

The Output Messages section allows the user to choose where to generate a log file containing any messages generated by the command processor whilst performing the translation.

An input command file (macro file) may be specified that contains further commands to be processed.

Any additional commands can be added in the extra commands section if needed, e.g

\$m /profiles_british.imp

REPORTSTEEL ON

3.3 Defaults and Persistence of Settings

Initially the only default settings are that Units are set to Metres and the Output Format is set to PDMS Equipment.

After being used, the ImPLANT-I GUI will remember the directories used for Input and Output Files and Output Messages but will not retain the names that were used. It will store the Input Command File (if specified) and any extra commands that were entered as well as the most recent state of the Units and Output Format.

4 Using ImPLANT-I via the Command Processor Interface

When using ImPLANT-I it is driven via the AVEVA Command Processor. You must specify the type of file to be generated (PDMS or Review - for Marine Outfitting select PDMS) and the names of the DGN Files whose graphics are to be translated.

A number of engineering items may be stored in one DGN file. When transferring data to PDMS/Outfitting, AVEVA ImPLANT-I will transfer all the engineering items into a single database element. The database element will typically either be a ZONE (containing a number of EQUIPMENT or VOLuME elements) or a STRUCTURE element. ImPLANT-I will allow you to specify the PDMS element type (EQUIPMENT, VOLM or STRUCTURE) and its name (if any).

More than one DGN file can be input for each PDMS/Outfitting or Review output file, and more than one PDMS/Outfitting or Review file can be generated within one ImPLANT-I session.

ImPLANT-I has a macro-processing facility that enables you to create a file of commands (called a macro) using a standard editor, and then ask ImPLANT-I to process the commands in that file. Therefore, you do not have to enter all commands interactively every time.

An alternative approach is to generate the ImPLANT-I script automatically with, for example, a Visual Basic program, or a Perl script. ImPLANT-I includes several commands designed for use in such scripts.

4.1 Overview

A typical session using ImPLANT-I can be divided into the following stages:

1. Start up ImPLANT-I.
2. You must specify the following:
 - The distance units used in the DGN file.
3. You can optionally specify the following:
 - How the MicroStation primitives are going to be grouped in PDMS/Outfitting/Review.
 - The curvature coarseness (or arc tolerance).
 - A shift factor for the model, in millimetres.
 - Which parts of the DGN model to exclude from the transfer.
 - How surfaces will be imported into PDMS/Outfitting or Review (that is, hollow or solid).
 - Steel profile parameters.
 - Whether you want commands and/or messages stored in a file.
 - Whether a separate Equipment or VOLM is to be generated for each primitive.

4. You specify the Attribute Filename for output (if required). The output of attributes can be toggled on and off for each DGN file processed.
5. You specify either the Design macro or the Review model filename.
6. You specify the DGN filename whose 3D graphics are to be translated. ImPLANT-I reads the DGN file and appends the appropriate data to the current output file (that is the Design macro file or the Review model file), according to the parameters set at Stage 2.
7. You can either finish the session or repeat the process from Stage 2 to export more files.

4.2 Default Operation

By default ImPLANT-I does the following:

1. By default, all the primitives generated will be under an EQUIPMENT element. Normally, all DGN entities are grouped under the same EQUI. However, when a CELL is encountered in the DGN file, a new EQUI is generated. When the import of the CELL is complete, the elements that follow in the DGN file are grouped below another new EQUI, and so on. To generate a separate equipment element for each primitive, you can use the **GROUPPRIMITIVES OFF** command. You may also use the **GROUPPRIMITIVES** command to limit the number of primitives in each EQUIPMENT, VOLM or STRUCTURE.
2. You can specify the name of the PDMS/Outfitting element to create using the ISFF command. If you do not specify a name, the filename is used instead. You may alternatively indicate that the element should be anonymous, or that no top-level element should be created.
3. The curved lines in the MicroStation surfaces are approximated into straight segments and the curved surfaces around a projected curved surface are approximated to convex planar faces. You can control the way the curved parts are approximated into a number of straight segments or planar faces using the APPROX value which ranges from 1 (very coarse, less faces/segments) to 10 (very smooth, more faces/segments). The default is 5.
4. ImPLANT-I will output all DGN surfaces as SOLID primitives by default.
5. The default is to INCLUDE ALL the primitives for export.
6. Attributes will not be output by default.
7. ImPLANT-I has no default units.

4.3 Units in ImPLANT-I

MicroStation works internally in terms of *position units* (also known as *units of resolution*), but also displays *master units* and *sub-units* (also known as *working units*). PDMS and Review work internally in millimetres, but can interact with the user in many other units.

In order to successfully translate a DGN file for PDMS or Review it is necessary to convert the units in the DGN file into millimetres. The DGN file contains two-letter abbreviations of the names of the *master unit* and the *sub-unit*, but these abbreviations are not standardised, and so the translation is not automatic.

To display the ISFF unit information from a file, use the **Q UNITS** command. For example, if the file you are working on is called **example.dgn** you should type:

```
Q UNITS /example.dgn
```

This will produce a message like:

```
Master Units : ME
Sub-Units : MM
Resolution:
1000      : MM per ME
80        : Pos Units Per MM
Global Origin : 0.000000 0.000000 0.000000
```

In this case, the MicroStation user has chosen to use metres as the *master unit*, millimetres as the *sub-unit*, and has one eightieth of a millimetre as the *position unit*.

The recommended way to set the units is with the **UNITS** command. If the master unit is one of **METRE** (or **METER**), **CM**, **MM**, **YARD**, **FOOT** or **INCH** you should use that keyword. In this case you would type:

```
UNITS METRE
```

If the master unit is not one of these, you should supply the length of a master unit in millimetres. In this case you could type:

```
UNITS 1000
```

as a metre is one thousand millimetres, but **UNITS METRE** is recommended.

4.4 Example ImPLANT-I Session

The following example shows a typical session using ImPLANT-I. References are made to the stages described in Section 2.1 at the appropriate points. Some of the commands in the example are described in detail in Section 4.4.1.

Text in *italic* indicates what the user is typing.

```

ImPLANTI.exe                                     (stage 1)
AVEVA ImPLANT-I Version 1.6 (WINDOWS-XP) (May 30 2008)
(c) Copyright 1992 to 2008 AVEVA Solutions Ltd AVEVA

* Q UNITS /turb1.dgn                             (stage 2)
Master Units : M
  Sub Units : MM
Resolution:
1000: MM per M
80      : Pos Units per MM
Global Origin: 0,0 0,0 0,0

*UNITS METRE                                     (stage 2)
*GENERATE PDMS /file1.model                      (stage 4)
*ISFF /turb1.dgn                                (stage 5)
*ISFF /turb2.dgn                                (stage 5)
*STRUC                                           (stage 2)
*GENERATE PDMS 'file2.model'                    (stage 4)
*SOLID                                           (stage 2)
*ISFF /pipe1.dgn                                (stage 4)
*MESSAGE /message                              (stage 2)
*GENERATE REVIEW /file3.model                  (stage 4)
*APPROX 3                                       (stage 2)
*HOLLOW                                         (stage 2)
*EXCL BOX                                       (stage 2)
*EXCL COL 0                                    (stage 2)
*SHIFT 2000.0 3000.0 0.0                      (stage 2)
*GROUPPRIMITIVES OFF                          (stage 2)
*ATTRIBUTES /file3.att                        (stage 3)
*ISFF /*.str /BUILD                           (stage 5)
*MES END                                       (stage 2)
*FINISH                                       (stage 6)

```

4.4.1 Notes on the Example Session

The commands used in the example are described in detail below:

GENERATE PDMS /file1

Open a PDMS DESIGN macro file called **file1.mac** which will hold the PDMS/Outfitting commands for importing 3D data taken from the DGN files specified by the next ISFF commands. Every **GENERATE** command closes down any previous opened files.

ISFF /turb1.dgn

Read the file **turb1.dgn** for 3D data and generate the PDMS commands for import to PDMS DESIGN. The PDMS primitives generated will be under the PDMS ZONE element called / **turb1.dgn** as no zone name was specified.

SOLID

After giving this command, any DGN surfaces will be output as solids. This command should be used, for example, when exporting pipes, so that tubes or bends are exported as cylinders and circular toruses and not as faceted models. The **SOLID** command is the default and produces solid models.

UNITS METRE

Units in ImPLANT-I

STRUC (or EQUIP or VOLM)

By default, all generated PDMS/Outfitting elements will be of type EQUIPMENT. To change this to STRUCTURE use the **STRUC** command and the **VOLM** command to change to VOLUME elements.

The **STRUC** command sets the mode so that the ISFF command generates the PDMS/Outfitting hierarchy STRUC/SUBST/Geometric primitives (BOX etc.). The **EQUIP** command sets the mode so that the ISFF command generates the hierarchy ZONE/EQUIPMENT/Geometric primitives (BOX etc.). In EQUIPMENT mode, the primitives are packed into the same EQUI until a MicroStation CELL entity is encountered. Then a new EQUI is generated. If GROUP ON is set in addition to EQUIPMENT mode then a new EQUI is generated for each micro station primitive.

Note that the STRU and ZONE PDMS elements take the name of the DGN file supplied, unless otherwise specified.

The **VOLM** command does exactly the same as the **EQUIP** command but replaces all EQUIPMENT elements with VOLUME elements.

APPROX 3

Change the curvature approximation to 3 (more coarse than the default value of 5).

GROUPPRIMITIVES

Stops primitives being grouped into arbitrary hierarchies. This is important if attributes are being output.

ATTRIBUTES /file3.att

For all geometry files processed by following ISFF commands, process the corresponding attribute files with the same names and output the results in an AVEVA Attributes file named **file3.att**. The attribute files normally have the file extension **.drv**, and their format is shown in Appendix B.

The **ATTRIBUTES filename** command opens a new attributes file and switches attribute processing on. Opening an attributes file will close any attributes file that is already open. **ATTRIBUTES END** closes the attributes file. The output of attributes can be toggled on and off.

Note: ImPLANT-I maintains *cell* ownership and Review segments within a *cell* segment inherit the attributes of the *cell* segment. In other words, the segments within a *cell* do not have attribute entries within the AVEVA Attributes file, only significant segments do.

ISFF /*.str /BUILD

Read all files with extension **.str** which have been generated by MicroStation and append the primitive records to the Review file **file3.model**. Standard windows wild-card characters may be used.

If you wish to use file names which include spaces, or certain other characters, you must use single-quotes around the name.

EXCLUDE BOX

From now on do not export boxes. To include them back again in this example the commands **INCLUDE ALL** or **INCLUDE BOX** can be used.

EXCLUDE COL 0

After this command has been given, primitives with colour 0 will not be exported. To include primitives with colour 0 again, the commands **INCLUDE ALL** or **INCLUDE COL 0** can be used. Note that sometimes MicroStation users use colour 0 to create a face that they subsequently use to create a projected surface, or surface of revolution object with a different colour. In this case there will be a clash of colour between the construction face colour (that is 0) and the object's colour as they occur in the same place. To avoid this clash of colours, in Review export only, exclude colour 0 from the model.

Most of the **EXCLUDE/INCLUDE** options are self-explanatory. However there are some options that describe MicroStation type of primitives/records and others describe 3D type of models as generated by ImPLANT-I.

The MicroStation types of primitives/records are described in the MicroStation documentation and these are: CELL, TRUNCATED CONE, CAPPED SURFACE, SURFACE, SURFACE OF REVOLUTION.

Other element types referenced by the **EXCLUDE** command:

CURVED SOLID Any solid that has been recognised as curved and which does not fit into the standard curved primitives such as circular torus, rectangular torus etc.

PLANAR SOLID Any solid that does not have curved parts and does not fit into the standard primitives such as box, pyramid etc.

SHIFT 2000.0 3000.0 0.0

The SHIFT command specifies a shift in millimetres for the model.

MESSAGE /message

MES END

The **MESSAGE** command specifies an output file where all user commands and output messages can be stored. The **MES END** command closes the file.

4.5 Steel Structures

ImPLANT-I comes with a facility for transferring steel structures from MicroStation Design files to PDMS/Outfitting through the PDMS catalogue mechanism, using the Structure/Framework/ Section hierarchy of PDMS. This involves the following stages (note that Steps 2 and 3 are not necessary if you know which profiles are used):

1. ImPLANT-I is supplied with two macro files defining the British Standard and DIN profiles included in the standard PDMS Catalogue Database:

profiles_british.imp

profiles_din.imp

These may be passed to ImPLANT-I by the usual command processor **\$m** mechanism; for example:

\$m/profiles_british.imp

2. Run ImPLANT-I using the **REPORTSTEEL ON** command for discovering any possible steel structures (include profiles_british.imp and/or profiles_din.imp if appropriate). ImPLANT-I responds with one or more possible Steel Structure profiles.
3. Look at the shapes of the reported profiles and decide if they match existing profiles held in the catalogue. If some profiles do not exist, you may create corresponding Catalogue components in the Catalogue database by, for example, using the PARAGON module. If you choose not to create corresponding Catalogue components, ImPLANT-I will still import the objects, but not as Sections.
4. Create a macro containing the corresponding data that defines the profiles included in the Catalogue.
5. Rerun ImPLANT-I and this time run the macro created by Step 4 (and any standard macro required). For PDMS transfers this will generate less data and so will speed up the process.
6. The Design macro file created can be entered into PDMS/Outfitting through the DESIGN module.

Note that you will usually need to create the Catalogue components and the corresponding Profile macro file once only. You can then use this file again when necessary.

4.5.1 ImPLANT-I Profile Macro

Profiles of sections are entered in ImPLANT-I through a series of commands, together with the necessary data. These commands are in standard Command Processor format.

The typical way of entering these commands is through the macro facility; that is, you prepare a macro file corresponding to the necessary commands, and then run the macro file to process the commands. For example:

\$m /<filename>

\$m '<filename>'

This will run the commands you have entered into the macro called filename.

There are three commands dealing with profiles: **ADDSTEEL**, **REMSTEEL** and **REPORTSTEEL**.

4.5.2 ADDSTEEL Command

The **ADDSTEEL** command has the format:

```
ADDsteel spec_name
n
x y          (times number of points)
PARAM/ETERS END
```

where:

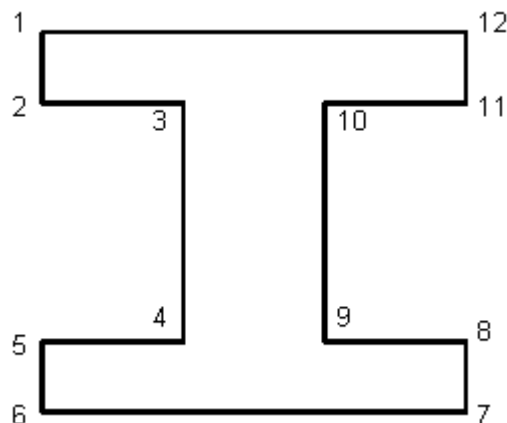
spec_name is the database name of the corresponding SPCO/CATR catalogue element

n is the number of points defining the 2D profile

x y are the co-ordinates of the points defining the 2D profile

(Some additional options are permitted in the command, for compatibility with previous versions of ImPLANT-I, but these are now ignored.)

For example, consider the following I shape:



The following sequence of commands is an example describing the I-shape. For this, the program computes the origin to be the average of all the points.

```
ADD /GENP1
12
92550.00 16450.00
92550.00 16415.00
92690.74 16415.00
92690.74 15585.00
92550.00 15585.00
92550.00 15550.00
92850.00 15550.00
92850.00 15585.00
92709.24 15585.00
92709.24 16415.00
92850.00 16415.00
92850.00 16450.00
PARAM END
```

The corresponding I-shape catalogue component has to be defined.

4.5.3 REMSTEEL Command

To disable the matching of profiles while in ImPLANT-I, use the following command:

REMsteel

This command removes the matching of all previously defined profiles.

4.5.4 REPORTSTEEL Command

To control reporting on possible profiles, use the following commands:

REPortsteel ON report on possible profiles

REPortsteel OFF do not report on possible profiles (default)

4.5.5 STEELFILTER Command

To control whether elements recognised as Sections should be included in the output file, use the following commands:

STEELFILTER SECTIONS ON

include recognised Sections in the output (default)

STEELFILTER SECTIONS OFF

do not include recognised Sections in the output

To control whether elements not recognised as Sections should be included in the output file, use the following commands:

STEELFILTER NONSECTIONS ON

include non-Sections in the output (default)

STEELFILTER NONSECTIONS OFF

do not include non-Sections in the output

Note that element types that are never Sections, such as cylinders and boxes, are not affected by these commands.

(You might use these commands to see how much of a model is being transferred as identified Sections and how much as geometrical primitives).

4.5.6 Recognising Profiles

The ImPLANT-I program stores the definition of the profiles as read by the program through the Profile macro file as described in a previous section.

For each DGN projected surface, the program will compare the profile of the surface with the profiles stored. If it finds a match, the program then generates Framework and Section elements. The Section element provides the SPRE, POSS, POSE, BANG, DRNS and DRNE attributes (see the *Design Reference Manual* for definitions of these attributes). The attribute SPRE is set to the *spec_name* as given in the Profile file.

To match a profile of a surface with a predefined profile the program checks:

1. That the number of points in each profile is the same.
2. That the total length of each profile is the same (within a tolerance).
3. That the area of each profile is the same (within a tolerance).
4. That the lengths of each span of each profile match (within a tolerance).
5. That the angles between each span of each profile match.

Note: Circular or curved steel sections are not considered for matching.

Note: There may be circumstances in which a profile is matched even though the projected surface did not represent a steel section.

The **STEELTOLERANCE** command may be used to change the tolerance used in this matching process.

4.6 ISFF Extensions

Some programs create DGN files that are not standard. ImPLANT-I can be configured to accept some of these extensions.

ImPLANT-I normally requires that surfaces made by using rules, such as capped surfaces, shall have their caps classified as **PRIMARY** and their rules as **PRIMARY RULES**.

To allow ImPLANT-I to accept such surfaces, in which there is always exactly one element in each cap and both the caps and the rules are classified as **PRIMARY**, type the command:

EXTENSION RULES PRIMARY ON

To allow ImPLANT-I to accept surfaces where the rules are classified as **CONSTRUCTION RULES**, type the command:

EXTENSION RULES CONSTRUCTION ON

Some previous versions had the first extension always enabled for certain surfaces, but not the second. The default behaviour is now for neither extension to be enabled.

To restore this default behaviour, type the command:

EXTENSION OFF

5 ImPLANT-I via the Command Line

ImPLANT-I can be driven via the command line if required.

5.1 Command Line Arguments

- f <macro-filename>** Process the specified macro file (Section 4.5.1)
- c <macro-filename>** Process the specified macro file, suppressing the console banner
- o <filename>** Redirect the standard output to the specified file, i.e. create a message file.
- w lines** Set the output window size. If the number of lines is negative the window will appear minimised. If *lines* equals 0 the window will not appear.

Example:

```
Macro_file.mac
units metre
STRUC
$m/scriptfiles/profiles_din.imp
gen pdms /results/example.mac
isff /dgnfiles/inputfile1.dgn
isff /dgnfiles/inputfile2.dgn
fin
```

From the command line:

```
ImPLANT-I.exe -f 'Macro_file.mac' -o 'std_output.txt'
```

This will generate a .mac file called example.mac that can be loaded into DESIGN and a report file called std_output.txt that can be read into any text editor, e.g. notepad.

6 Limitations, Warnings and Error Messages

6.1 Geometry Transfer Limitations

1. The following elements have not been implemented:
 - B-spline Pole (Type 21)
 - B-spline Surface Header (Type 24)
 - B-spline Surface Boundary (Type 25)
 - B-spline Knot (Type 26)
 - B-spline Curve Header (Type 27)
 - B-spline Weight Factor (Type 28)
2. The Curve element (Type 11) and Conic element (Type 13) are treated as a series of straight segments and not as curves.
3. The translator does not deal with warped surface (double curved surfaces, typically defined by B-Splines or other curves) and volumes defined by boundary elements within the Surface Header (Type 18) or Solid Header (Type 19).

Type of Surface

0=Surface of projection	implemented
1=Bounded Plane	non implemented
2=Unbounded Plane	not implemented
3=Right Circular cylinder	implemented
4=Right Circular cone	implemented
5=Tabulated cylinder	not implemented
6=Tabulated cone	not implemented
7=Convolute	implemented
8=Surface of revolution	implemented
9=Warped surface	not implemented

Type of Solid

0=Volume of projection	implemented
1=Volume of revolution	implemented
2=Volume defined by boundary elements	implemented as volume of projection

4. 3D Lines are ignored.
5. Holes defined independently and not as part of a cell are ignored.
6. Holes made from negative cones are ignored.
7. Some wedges have been implemented using a different method; instead of two 3-sided shapes and four ruled lines, a rectangle and line can be used with four ruled lines.
8. Shared Cell Definition and Shared Cell Instances have not been implemented.
9. Smart Solids and Smart Surfaces are not supported (but see section 6.2).

6.2 Using MicroStation to Improve the Translation

It is often possible to use MicroStation to improve the quality of a translation, by "dropping" unsupported Smart Solids and Shared Cells to simpler elements which can be translated.

- Using the "Drop Element" tool drop all "Shared Cells" to "Geometry" (This may need to be done more than once, in the case where shared cells are nested).
- Using the "Drop Element" tool drop all "Solids" to "Surfaces".
- If you are using MicroStation V8 or newer, use "Save As" on the file menu to save in "MicroStation V7 DGN files" format.

6.3 Specific Review or PDMS/Outfitting Limitations

1. In Review the surfaces of an open surface of revolution are not smooth shaded.
2. Large amounts of data are generated for both PDMS/Outfitting and Review. This is because neither PDMS/Outfitting nor Review can store non-standard primitive shapes economically.

This, for example, may cause problems with the DRAFT module. This module can not cope with a large number of primitives under a significant element (such as an Equipment, Volume, Structure, Polyhedron etc). There is a limit of 4096, but performance will be severely degraded with more than 500. You may use the GROUPP command to limit the number of primitives in an element in PDMS.

Specifically, you are advised to read large Design macros by entering DESIGN in **dev** **ty** mode and then use the **TRACE OFF** command before you run the design macro.

3. Steel Structures limitations:
 - Circular or curved steel sections are not considered for matching.
 - There may be circumstances that a profile is matched although that model was not meant to be a steel structure.
4. The data generated in PDMS can be clash-checked in DESIGN. However, the DGN data cannot be thoroughly checked for clashes within itself unless you create a separate EQUIPMENT, VOLuMe or STRUcture for each primitive.

6.3.1 Intelligent Transfer Limitations

1. Cell libraries, Shared Cell Definitions (Type 34) and Shared Cell Instances (Type 35) are not transferred.
2. Attribute data is not transferred to PDMS.
3. AVEVA ImPLANT-I ignores the definition of Group Data.
4. AVEVA ImPLANT-I ignores the definition of Levels (layers).
5. Cell Hierarchy is not transferred in Review.
6. Element Names are not transferred.

6.3.2 Not in the Scope of ImPLANT-I

1. Font Libraries, text and 2D data.
2. Dimension (Type 33), Multi-line (Type 36).
3. The class types of 1 Pattern component, 2 Construction element, 3 Dimension element, 5 Linear Patterned element and 6 Construction rule element are not taken into account.
4. Raster Header (Type 87), Raster Data (Type 88) are ignored.
5. Boundaries, holes or a combination of boundaries with holes, with lines intersecting produce a warning and the model is exported.
6. Point String (Type 22) is ignored.

6.3.3 Other Limitations

Any messages produced by the Command Processor are not stored in the message file, e.g. syntax errors.

6.4 Warning Messages

This section notes the warnings that could occur when running ImPLANT-I. Further information is provided with some of the warnings.

Depending on the warning, you are advised to try to correct the inconsistencies in the DGN model and rerun ImPLANT-I. ImPLANT-I will continue running when a warning is issued, but the object that generated the warning will not be exported.

Most of the expected warnings will be produced on ill-defined faces:

- A face has been defined with duplicate points.
- A line in a face intersects or touches another line or point in the same face.
- A face defined as a number of points makes a single line, and therefore is not a correct face.

The following warnings may occur in the above category:

Warning - Straight line found

Warning - External Boundary 'n' degenerate

Warning - External Boundary 'n' self intersecting

Warning - External Boundary 'n' self touching

Warning - Gap in loop of projected surface or surface of revolution

Warning - Hole Boundary 'n' is degenerate

Warning - Hole Boundary 'n' self intersecting

Warning - Hole Boundary 'n' self touching

Warning - Hole is degenerate

Warning - Start and End Coordinates do not match

Warning - Shape forced onto a plane

The following warnings may occur when certain complex elements are incomplete, or are not defined in the conventional manner:

Warning - No rules found for capped surface or surface

Warning - No boundaries found for capped surface or surface

Warning - The capped surface or surface has not the same number of boundary records or is ill defined

Warning - Nested Complex Shape found (but accepted)

Other warnings will be produced when some special cases have not been implemented in this version of ImPLANT-I:

Warning - B-spline type found, not implemented

Warning - Not implemented solid/surface type 'n'

Warning - A planar solid has been found that has not been implemented

Warning - Not implemented surface type 'n'

Warning - Not implemented Volume Boundaries Type

Warning - Not implemented Surface Boundaries Type

Warning - Cell Library Type Not Implemented

Warning - Line Type Not Implemented

Warning - Text Node Type Not Implemented

Warning - Complex String Type Not Implemented

Warning - Text Type Not Implemented

Warning - Point String Type Not Implemented

Warning - B-Spline Type Not Implemented

Warning - Shared Cell Definition Type Not Implemented

Warning - Shared Cell Instance Type Not Implemented

Warning - Group Data Not Implemented

Warnings will be produced when there is something wrong with the construction of the model:

Warning - The hole does not follow an external boundary

Warning - The construction of the element has not been implemented

Warning - Invalid type found, type = 'n'

Warning - Two shapes in projected or surface of revolution do not have the same number of vertices

Warning - A solid was found in a hole surface

Warning - This construction cannot produce a primitive

Warning - The following hole plane is not on the same plane

Finally, warnings will be produced when ImPLANT-I is used in an invalid manner:

Warning - A file for output has not been defined

Warning - The millimetre factor has not been set

6.5 Error Messages

This section lists errors that could occur when running ImPLANT-I.

The program stops processing after an error is reported.

Error - Unexpected End of File was found

Error - Unexpected record for a complex shape hole

Error - Cannot open file

Error - The read from the file failed error 'n'

Error while closing ... file (may be incomplete):

If the following errors occur, please contact the AVEVA Customer Support desk:

Error - Not all the allocated memory has been cleared

Error - Insufficient memory space

Error - Pointer was not found in buffer store

6.6 Other Messages

Under some circumstances ImPLANT-I may write other messages to the standard output, and/or the message file. These may provide extra information to help you or AVEVA Customer Support to identify the causes of any problems. If you receive one of these messages and need more information about it, please contact AVEVA Customer Support, sending a complete message file.

A Appendix: Command Syntax

The following graphs describe the sequence of commands that you can use with ImPLANT-1.

Refer to [Example ImPLANT-I Session](#), for examples of using the main commands.

ADDSTEEL

The **ADDSTEEL** command defines a steel profile in ImPLANT-I. The corresponding profile must be defined in the PDMS/Outfitting Catalogue.

```

>-- ADDsteel - SPEC NAME -- N --+-- X -- F --
|
|-- PARAMETERS -- END -->

```

where:

spec_name is the name of the corresponding SPCO/CATR catalogue element

n the number of points defining the 2D profile

x y the co-ordinates of the points defining the 2D profile

(Some extra options are accepted by the program, for compatibility with previous versions, but are now ignored.)

For more information, see Section 4.5, *Steel Structures*.

APPROX

The **APPROX** command sets the curvature approximation (chord tolerance). Values between 1 and 10 are allowed. The default is 5.

```
>-- APProximate -- val -->
```

ATTRIBUTES

The **ATTRIBUTES** name command opens a new attributes file and switches attribute processing on. Use the **ATTRIBUTES** text variant if you want to use a filename containing spaces. Opening an attributes file will close one that is already open. **ATTRIBUTES END** closes the attribute file. The output of attributes can be toggled on and off. The **ATTRIBUTES EXTENSION** command allows you to change the file extension of the associated file (by default, **drv**).

```

>-- ATtributes --+-- name -----,
|
|
|-- text -----|
|
|
|-- ON -----|
|
|
|-- OFF -----|
|
|
|-- END -----|
|
|-- EXTension --+-- name --|
|
|
|-- text --+-->

```

EQUIP

The **EQUIP** command sets the mode so that the **ISFF** command generates the hierarchy Zone/Equipment/Geometric primitives (Box etc.). In EQUIPMENT mode, the primitives are packed into the same EQUI until a MicroStation CELL entity is encountered. Then a new EQUI is generated. If GROUP ON is set in addition to EQUIPMENT mode, a new EQUI is generated for each micro station primitive.

The alternatives are the **VOLM** and **STRUC** commands.

> - - EQUIPment - - >

EXCLUDE

The **EXCLUDE** command can be used to control which primitives are translated. You can exclude primitives by type, by colour or by level. See also the **INCLUDE** command.

The **EXCLUDE LEVEL** command has an extended form not shown in the picture to exclude several levels in one command; for example:

EXCLUDE LEVEL 3, 4-6, 8

```

>-- EXclude --*-- ALL -----
-- CELL -----
-- TRUNcatedcone -----
-- CYLinder -----
-- CONe -----
-- SNOut -----
-- CAPnedsurface -----
-- CURvedsolid -----
-- CIRcularatorus -----
-- RECTangularatorus -----
-- SPHERE -----
-- DISh -----
-- PIANarsolid -----
-- BOX -----
-- PYRamid -----
-- SURFACE -----
-- OTRuncatedcone -----
-- OCurvedsolid -----
-- OCircularatorus -----
-- ORECTangularattorus -----
-- ODISH -----
-- OPLANarsolid -----
-- SURfaceofrevolution -----
-- COLOUR --+-- ALL --+--
-- -- 153.7 --+--
-- LEVEL --+-- ALL --+--
-- -- 153.7 --+--

```

EXTENSIONS

The **EXTENSIONS** command tells ImPLANT-I how to deal with certain non-standard constructions found in some DGN files (see Section 4.6).

```
>-- EXTensions --+-- OFF -----
|
|  -- DEFault -----
|
|  ' -- RULEs --+-- CONstruction --+-- ON -----
|
|
|
|
|
|
|  ' --PRImary -----+-- ON -----
|
|
|  -- OFF -----
|
|  -- DEFault --
|
|  -- DEFault --+-->
```

FINISH

The **FINISH** command exits from AVEVA ImPLANT-I.

```
>-- FINish -->
```

GENERATE

The **GENERATE** command opens a named file for input into Review or PDMS. Any previously opened files will be closed. The filename can be given as a name (with a leading /) or as text (enclosed in single quotes). If the filename contains spaces you must use the text form. If the filename you supply does not have an extension, AVEVA ImPLANT-I will add **.mac** for a PDMS/Outfitting macro file, or **.rvm** for a Review model file.

```
>-- GENerate --+-- PDMS -----
|
|  ' -- REVIEW --+-- filename -->
```

GROUPPRIMITIVES

The **GROUPPRIMITIVES** command affects how ImPLANT-I groups the Review and PDMS primitives. **GROUPPRIMITIVES** should be switched off if attributes are to be output.

For import to PDMS/Outfitting, all the primitives generated will be under a EQUIPMENT element. Normally, all DGN entities are grouped under the same EQUI. However, when a CELL is encountered in the DGN file, a new EQUI is generated. When the import of the CELL is complete, the elements that follow in the DGN file are grouped below another new EQUI, and so on. To generate a separate equipment for each primitive, you can use the **GROUPPRIMITIVES OFF** command.

You may alternatively supply a maximum number of primitives to group under an equipment. Under normal conditions hundreds of primitives per equipment will give acceptable performance, but thousands of primitives per equipment may not. This does however depend on the precise geometry involved.

```
>-- GROUPPrimitives --+-- ON ---.
      |               |
      |-- OFF --+
      |               |
      '-- int --+-->
```

HOLLOW

The **HOLLOW** command specifies that DGN surfaces will be output as surfaces, and not solids, which is the default. Note that drawing will be slower than for solids. To change back to importing solids, use the **SOLID** command.

```
>-- HOLLOW -->
```

INCLUDE

The **INCLUDE** command is the opposite of **EXCLUDE**; that is, excluded options are included for export again. See the notes on **EXCLUDE** and the example in [Example ImPLANT-I Session](#), Example ImPLANT-I Session for more information on primitive types.

```

>-- INCLUDE --*-- ALL -----
-- CELL -----
-- TRUNcatedcone -----
-- CYLinder -----
-- CONe -----
-- SNOuT -----I-----
-- CAPpedsurface -----
-- CURvedsolid -----
-- CIRculartorus -----
-- RECTangulartorus -----
-- SPHere -----
-- DISh -----
-- PLANarsolid -----
-- BOX -----
-- PYRamid -----
-- SURFACE -----
-- OTRuncatedcone -----
-- OCURvedsolid -----
-- OCirculartorus -----
-- OREctangulartorus -----
-- ODISH -----
-- OPLANarsolid -----
-- SURfaceofrevolution --
-- COLOUR --+-- ALL --.
-- val --+--
-- LEVel --+-- ALL --.
-- val --+--
-->

```

ISFF

The **ISFF** command reads a named input file and generates a Review model file or a PDMS/Outfitting command file. Any previously opened files will be closed. The filename can be given as a *name* (with a leading */*) or as *text* (enclosed in single-quotes). If the filename contains spaces you must use the *text* form. If the filename does not contain spaces then you may use wildcards to read in several files at once. For example, on Windows:

```
ISFF 'someFolder\*.dgn'
```

You can specify the name of the PDMS Zone as a *name* (with a leading */*). If you do not want a Zone element to be created use the **CONTINUE** option to continue with the current zone. If you want a Zone without a name use the **ANONYMOUS** option. If no Zone name is given and neither of the other options is specified, then the Zone name will be created from the input model filename.

```
>-- ISFF -- filename --+-- zonename ---.
|                                     |
| -- CONTinue ---|
| -- ANOnymous ---|
|                                     |
|'-----+-->
```

MESFIL

The **MESFIL** command controls the message file and related facilities.

To send messages only to a message file use **MESSAGEFILE** *filename*.

To send messages both to the standard output and to a message file use **MESSAGEFILE COPY** *filename*.

To send messages only to the standard output use **MESSAGEFILE END**.

The filename (if required) can be given as a *name* (with a leading */*) or as *text* (enclosed in single quotes). If the filename contains spaces you must use the *text* form.

To control echoing of commands from a macro to the message file use the **ECHO ON** and **ECHO OFF** commands. If you use a lengthy standard macro, you may wish to include an **ECHO OFF** at the start and an **ECHO ON** at the end of the macro. This is done, for example, in the supplied steel section macros.

To write a message directly to the message file use the **MESSAGEFILE PRINT** *message* command. The message must be enclosed in single quotes.

```
>-- Messagefile --+-- filename -----
|                                     |
| -- COpy -- filename --|
| -- END -----|
| -- ECHO --+-- ON -----|
|                                     |
|                                     |
|                                     |
|'-----+-->
| -- Print-- message ----+-->
```

PDMSCOMMAND

The **PDMSCOMMAND** command allows you write a PDMS command directly to the PDMS macro file, either at once or at the start of each Equipment. Each command must be enclosed in single quotes. Use two single quotes to represent a single quote within the command. For example:

```
PDMSCOMMAND 'NEW SITE \mySite'
PDMSCOMMAND EACHEQUIPMENT 'DESCRIPTION ''July 2008'''
>-- PDMScommand --+-- command -----,
|
|  -- EAChequipment --+-- command --|
|                               |
|                               |  -- OFF -----+-->
```

QUERY

The **QUERY** command allows you to query the units used in a MicroStation file. The master units, sub-units and unit-of-resolution will be output. The filename can be given as a *name* (with a leading /) or as *text* (enclosed in single quotes). If the filename contains spaces you must use the *text* form.

You can also query the current ImPLANT-I settings and the current tolerances.

```
>-- Query --+-- units -- filename --,
|
|  -- SETtings -----|
|
|  -- TOLerances -----+-->
```

REMSTEEL

The **REMSTEEL** command disables the matching of profiles while in ImPLANT-I. This command removes the matching of all previously defined profiles. See the **ADDSTEEL** command.

```
>-- REMsteel -->
```

REPORTSTEEL

The **REPSTEEL** command controls reporting on possible profiles. Reporting can be restricted to profiles with fixed length.

```
>-- REPortsteel --+-- ON --+-----,
|
|  -- FIXEdlength --|
|
|  -- OFF -----+-->
```

SHIFT

The **SHIFT** command specifies a shift in millimetres for the model.

```
>-- SHIfT -- x -- y -- z -->
```

SOLID

The **SOLID** command specifies that ISFF surfaces will be output as solids, which is the default. To specify that surfaces should be used to produce hollow primitives, use the **HOLLOW** command, but note that drawing will be slower than for solids.

```
>-- SOLId -->
```

STEELFILTER

The **STEELFILTER** command allows you to selectively include or omit elements recognised, or not recognised, as catalogue sections (see Section 4.5.5).

```
>-- STEELFILTER --+-- SECTIONS -----+-- ON ---|  
                    |                      |-- OFF --+  
                    |                      |          |  
                    '-- NONSECTIONS --+-- ON ---|  
                                    |-- OFF ---->
```

STEELTOLERANCE

The **STEELTOLERANCE** command lets you set the tolerance (in mm) used when matching steel sections.

```
>-- STEELTOLerance -- tolerance -->
```

STRUC

The **STRUC** command sets the mode so that the **ISFF** command generates the hierarchy STRUC/SUBST/Geometric primitives (Box etc.). The alternatives are the **EQUIP** and **VOLM** commands.

```
>-- STRucture -->
```

TOLERANCE

AVEVA ImPLANT-I uses several geometric tolerances when translating primitives; for example, to decide whether two points are coincident. We recommend that under most circumstances you use the default settings, as these have been chosen to be suitable for a wide range of DGN files.

The **TOLERANCE** command is provided for the rare cases when altering the tolerances can improve the translation.

TOLERANCE DEFAULT restores the recommended values.

TOLERANCE QUERY displays the current tolerance values (equivalent to **QUERY TOLERANCE**).

TOLERANCE FACTOR scales all the tolerances in proportion, and should be the first option to try if you want to adjust the tolerances. A factor of 1.0 is the same as **DEFAULT**, a factor of 2.0 loosens the tolerances, and a factor of 0.5 tightens them.

The other tolerance options affect individual tolerances used in various circumstances. These should very rarely need individual adjustment, and will not be further explained here.

```
>-- TOLerance --+-- DEFault -----
|
| -- Query -----|
| -- FACTor -- factor --|
| -- NUMber -- value ---|
| -- ANGLE -- value ----|
| -- VECTor -- value ---|
| -- POInt -- value ----|
| -- POLygon -- value --|
|
| -- NORmal -- value ----+-->
```

UNITS

Specifies the size of a MicroStation master unit, by name or in millimetres, refer to [Units in ImPLANT-I](#).

```
>-- UNITS --+-- MM -----
|
| -- CM -----|
| -- METre --|
| -- METEr --|
| -- INCh ---|
| -- F0ot ---|
| -- YArD ---|
|
| -- val ----+-->
```

VOLM

The **VOLM** command sets the mode so that the **ISFF** command generates the hierarchy Zone/VOLuMe/Geometric primitives (Box etc.). In VOLuMe mode, the primitives are packed into the same VOLM until a MicroStation CELL entity is encountered, when a new VOLM is generated. If GROUP ON is set in addition to VOLuMe mode, a new VOLM is generated for each micro station primitive.

Other alternatives are the **EQUIP** and **STRUC** commands.

```
>-- VOLM -->
```

B Appendix: Attribute Translation

ImPLANT-I assumes the following format for the attribute files that it reads in using the **ATTRIBUTE** command:

```
keyword <blank>{<blank><attLink><blank>text<blank>{<attVal><attVal>
```

```
...
```

```
...
```

```
<attVal><blank>}<blank>}
```

or

```
<blank>}<blank>}
```

where:

<blank> is zero or more spaces.

<attLink> is four integer values separated by a space.

<attVal> is an attribute pair separated by a colon : . Any attribute pair that does not contain a : is ignored and a warning given.

Index

A

Adding the input files	3:2
Appendix	
Attribute Translation	B:1
Command Syntax	A:1

B

Basic Installation	2:1
B-splines	6:1

C

Cell libraries	6:3
Checking the Macro Files	2:3
Checking the Review Model Files	2:3
Command line arguments	5:1
Command Processor Interface	4:1
Command Processor overview	4:1
Commands	
ADDSTEEL	4:8, A:1
APPROX	A:1
ATTRIBUTES	A:2
ATTRIBUTES EXTENSION	A:2
EQUIP	A:2
EXCLUDE	A:2
EXCLUDE LEVEL	A:2
EXTENSIONS	A:4
FINISH	A:4
GENERATE	A:4
GROUPPRIMITIVES	A:4
HOLLOW	A:5
INCLUDE	A:5
ISFF	A:7

MESFIL	A:7
PDMSCOMMAND	A:8
QUERY	A:8
REMSTEEL	4:9, A:8
REPORTSTEEL	4:9, A:8
SHIFT	A:8
SOLID	A:8
STEELFILTER	4:9, A:9
STEELTOLERANCE	A:9
STRUC	A:9
TOLERANCE	A:9
UNITS	A:10
VOLM	A:10

D

Default settings	3:2
------------------------	-----

E

Electronic Manuals	2:2
Error Messages	6:1, 6:5
Example ImPLANT-I Session	4:3

F

Flexman License Manager	2:2
-------------------------------	-----

G

Geometry Transfer Limitations	6:1
Graphical User Interface	1:2, 3:2

H

How to Use this Manual 1:1

I

ImPLANT-I Default Operation 4:2
 ImPLANT-I Dialog 3:1
 ImPLANT-I Dialog Overview 3:1
 ImPLANT-I Profile Macro 4:7
 ImPLANT-I via the Command Line 5:1
 Input command files 3:2
 Input to ImPLANT-I 1:2
 Installation 2:1
 Intelligent Transfer Limitations 6:3
 Intergraph Design Review 1:1
 Introducing AVEVA ImPLANT-I 1:1
 ISFF Extensions 4:10

M

Microsoft .NET Framework Installation .. 2:1
 MicroStation 1:2
 MicroStation DGN V8 1:1
 Modes of operation 1:2

N

Not in the Scope of AVEVA ImPLANT-I . 6:3

O

Optional Settings 3:2
 Other Limitations in AVEVA ImPLANT-I . 6:3
 Other Messages 6:5
 Output from ImPLANT-I 1:2
 Output messages 3:2

P

PDMS Equipment 3:2
 PDMS Structure 3:2
 PDMS Volume 3:2
 PDMS/Outfitting Limitations 6:2
 PRIMARY RULES 4:10

R

Recognising Profiles 4:9
 Running in a command Window 2:2

S

Setting the output file 3:2

Setting the units 3:2
 Shared Cell Definitions 6:3
 Shared Cell Instances 6:3
 Smart Solids 6:2
 Steel Structures 4:7

T

Test Files 2:3
 Testing the installation 2:2

U

Units in AVEVA ImPLANT-I 4:2
 Using MicroStation to Improve the Translation
 6:2

W

Warning Messages 6:3
 Warnings 6:1