

Agile

Version e6.1

ORACLE

# **Oracle Agile Engineering Data Management**

MCAD Connector for Catia V5 - Version 6.5.3.0  
Installation and Administration Manual

Part No. E14022-02

September 2009



## Copyright and Trademarks

*Copyright © 1995, 2009, Oracle and/or its affiliates. All rights reserved.*

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

### U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

# CONTENTS

---

Copyright and Trademarks.....	iii
Preface .....	v
<b>Introduction.....</b>	<b>1</b>
Overview.....	1
Architecture .....	1
<b>Installation and Configuration.....</b>	<b>3</b>
Prerequisites.....	3
Checking the Prerequisites .....	3
Installing the Software .....	3
Adapting the Startup Script .....	4
Customizing Vault Options .....	5
<b>Configuration .....</b>	<b>7</b>
Agile e6.1.....	7
CATIA .....	7
Document Folder.....	7
Integration Environment .....	8
Adapting the File CATIA_ECC.txt.....	9
Connection Configuration Files .....	9
Master Configuration File Ecc6Initialize.ini .....	10
Options for EdbUser Settings.....	17
Internal Structure of Mapping Files.....	18
Basic Configuration File Ecc6.ini.....	19
Customer Configuration File Ecc6Customer.ini.....	19
<b>Parameter Transfer .....</b>	<b>21</b>
Transfer from CATIA to Agile e6.1 .....	21
Frame Mapping .....	21
Drawing Sheets .....	22
Baselines .....	23
Comparing Assembly Versions – EccCompareVersion .....	24
<b>Debugging.....</b>	<b>25</b>

# Preface

The Oracle documentation set includes Adobe® Acrobat™ PDF files. The [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technology/documentation/agile.html) (<http://www.oracle.com/technology/documentation/agile.html>) contains the latest versions of the Oracle Agile EDM PDF files. You can view or download these manuals from the Web site, or you can ask your Agile administrator if there is an Oracle Documentation folder available on your network from which you can access the documentation (PDF) files.

---

**Note** To read the PDF files, you must use the free Adobe Acrobat Reader™ version 7.0 or later. This program can be downloaded from the [Adobe Web site](http://www.adobe.com) (<http://www.adobe.com>).

---

---

**Note** Before calling Agile Support about a problem with an Oracle Agile EDM manual, please have the full part number, which is located on the title page.

---

## TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

## Readme

Any last-minute information about Oracle Agile EDM can be found in the Release Notes file on the [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technology/documentation/agile_eseries.html) ([http://www.oracle.com/technology/documentation/agile\\_eseries.html](http://www.oracle.com/technology/documentation/agile_eseries.html))

## Agile Training Aids

Go to the [Oracle University Web page](http://www.oracle.com/education/chooser/selectcountry_new.html) ([http://www.oracle.com/education/chooser/selectcountry\\_new.html](http://www.oracle.com/education/chooser/selectcountry_new.html)) for more information on Agile Training offerings.

## Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

## Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the

accessibility of these Web sites.

## Introduction

### Overview

The information in this document is based on a standard installation and the actual appearance of a specific installation may vary depending on the local environment and custom configuration realized for the individual requirements of the customer.

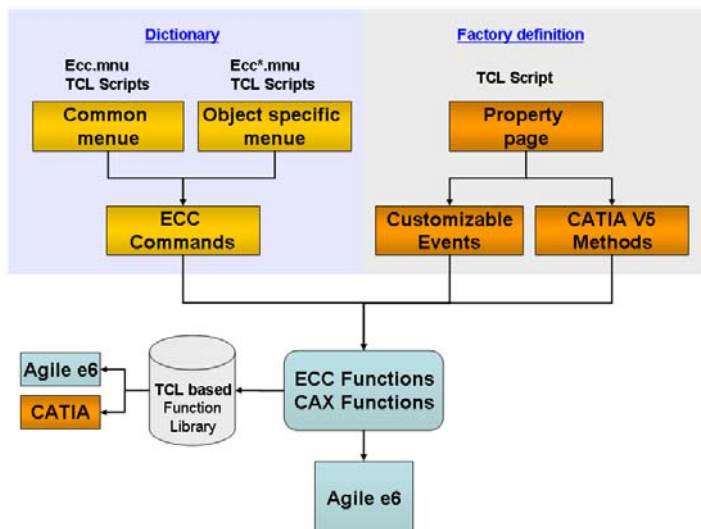
The integration functions are provided with additional and expanded menus in Agile e6.1 and CATIA.

### Architecture

The Agile e6.1 - CATIA integration was created using the development tools delivered by Dassault Systèmes. It represents an enhancement to the operations of CATIA. The Agile e6.1 - CATIA integration (ECC) is an add-on module to Agile e6.1.

Agile offers a uniform GUI which operates with different operating systems throughout the company. This facilitates the operation of the integration despite the use of different operating systems. Many customizing requests can be solved using the script language Tcl. Tcl is also available on different platforms.

The architecture of the integration between Agile e6.1 and CATIA is illustrated in the following graphic:







## Chapter 2

# Installation and Configuration

The installation of the Agile e6.1-CATIA integration (ECC) is divided into several steps. Performing the steps in the proper order is important for correct operation.

The following steps have to be carried out:

1. Prerequisites
2. Checking the Prerequisites
3. Installing the software
4. Adapting the Startup Script
5. Customizing Vault Options

## Prerequisites

The Agile e6.1-CATIA integration is currently available for several hardware platforms and operating systems. For an overview of platforms refer to the Release Notes for Catia V5 – Version 6.5.3.0.

## Checking the Prerequisites

- ORACLE is installed and running.
- Agile e6.1 is installed and running.
- A designated user environment is already set up in Agile e6.1.
- In Agile e6.1 a test user with a valid password exists and is authorized to start the Agile e6.1 session.
- If the Agile e6.1 File Server should be used check whether it is working.
- CATIA is installed and can be launched by the test user in the home directory.
- The latest version of the integration that is compatible with the operating system and the CAD system should be available.

## Installing the Software

To install the ECC Integration extract the delivered zip file (e.g. ecc6530.zip) to any path on your hard drive (e.g. C:\AgilePLM\ecc6)

The installation creates the following directory structure:

Main Directory	Sub-directory	Sub-directory	Sub-directory	Description
\bin				Program libraries
	\aix_a			ECC libraries for AIX
	\hpux_b			ECC libraries for HP-UX
	\Intel_a			ECC libraries for NT
	\solaris_a			ECC libraries for SUN
		\plm61		Agile e6 EDB system
			\rXsp0	ECC binary directory for CATIA V5 Release, X = CATIA Release
\com				Configuration files
\dic				ECC function definition
\Frames				Drawing frames
\ini				Initialization files and info text
\install				Post-install routines
\Menu				Definition files of the menu structure
	\icons			Icons
	\msgcatalog			Message catalog
\tcl				TCL start procedures
\tcllib				TCL library files
\EccUser*				
	\bck			Activates the cache mechanism
	\tmp			Temp directory
	\work			Local working directory of the user

\* The “ECC-User” directory contains working areas with 3 sub-directories for each user. Reference to the individual user-specific directories can be changed.

## Adapting the Startup Script

To set the customer specific variables the respective startup file needs to be adapted.

- ecc6\com\cv5.cmd (MS Windows)
- ecc6/com/cv5.ksh (Unix)

The following table refers to the MS Windows Startup Script

<b>Variable</b>	<b>Description</b>	<b>Valid Content (default setting)</b>
EccRoot	Root directory of integration	D:\ecc6
EccUserHome	User directory	D:\%EccUserRoot%\%USERNAME%
EccWorkPath	Working directory of the user	%EccUserHome%\work\
EccTempPath	Temp directory of the user	%EccUserHome%\tmp\
EccBckPath	Cache mechanism	%EccUserHome%\bck\
EccTclScriptDir	Directory of TCL scripts	%EccRoot%\tcl\
TCL_LIBRARY	Directory of TCL libraries	%EccRoot%\tcllib\
CaxConfigDir	Directory of configuration files	%EccRoot%\ini\

## Customizing Vault Options

The CAD files are stored in a central vault when using the Agile e6 FMS (FMS = File Management Server). In the FMS the filenames are created in an encoded form when the data are transferred to the FMS. Working with the file server permits central, and protected data storage.

The default name of the used vault is “CAX” (capitalized). If you want to use another name, you need to change it in Agile e6.1.

---

**Note** How to set up a vault can be found in the Agile e6.1 documentation.

---



## Chapter 3

# Configuration

## Agile e6.1

The following data models are required for running the connector. It is already implemented in the standard dump, and is only shown here to compare it with older customized dumps:

Document types for drawings and 3D models:	DRAWING and 3D_MODEL
Masks for document types:	EDB-DOC-DRW-TFR / TLI EDB-DOC-3DM-TFR / TLI
Document loading list (master list):	EDB-DOC-SLI

The following fields must be verified and added irrespective from the standard data model fields:

Database Field	Type
T_DOC_DAT.CAX_FIL_NAME	S255
T_DOC_DAT.CAX_FIL_OLD_NAME	S255
T_DOC_DAT.CAX_FIL_PATH	S255
T_DOC_DAT.CAX_LOCAL	L
T_DOC_DAT.CAX_CRE_SYSTEM	S20
T_DOC_DAT.CAX_TYPE	S20
T_DOC_DAT.CAX_FRAME_ID	S20
T_DOC_DAT.CAX_TIMESTAMP	S20
T_DOC_DAT.CAX_SUBTYPE	S20

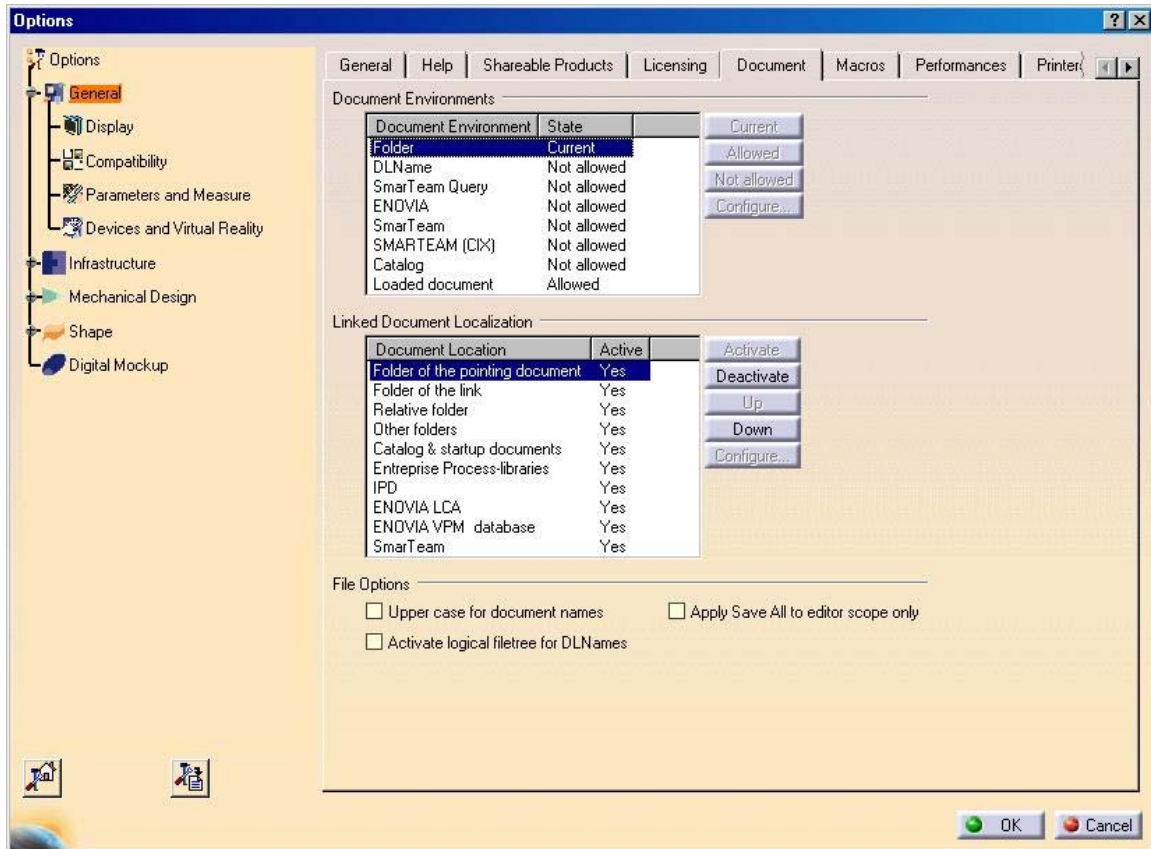
## CATIA

### Document Folder

To ensure a correct working of the ECC integration with the Agile standard dump it is necessary to

carry out the following adaptations in CATIA.

1. Click Tools > Options and select General.
2. Open the Document tab.  
Documents linked with the folder are listed here.
3. In the Document Environments field select Folder and select Current from the State column.
4. In the Linked Document Localization field select Folder of the pointing document and select Yes from the Active column.



## Integration Environment

Check the correct settings of environment variables described below. These variables must be adapted to the present installation environment.

The content of integration scripts (e.g. the syntax) depends on the operating system.

MS Windows: ecc6\com\ECC\_Rx.cfg

UNIX: ecc6/com/unix/ECC\_Rx.cfg

---

**Note** MS Windows notation is used for the following settings.

---

## Adapting the File CATIA\_ECC.txt

The file ...bin\intel\_a\plm60rXsp\CATIA\_ECC.txt has to be adapted. Its structure agrees with the CATEnv-file V5Rx.B0x.txt.

Integration-specific enhancements in the CATIA settings are:

```
CATDLLPath = ...;%EccRoot%\bin\intel_a;
%EccRoot%\bin\intel_a\%EdbV%\%Csp%;%EccRoot%\bin\intel_a\%EdbV%
CATDictionaryPath = ...;%EccRoot%\dic
CATGraphicPath = %EccRoot%\Menu\icons;...
CATMsgCatalogPath = %EccRoot%\Menu\msgcatalog;...
```

## Connection Configuration Files

The connection configuration files are located in the directory ...ecc6\ini\.

ECC\_Rx.cfg – Variables to Determine ECC Conditions

Variable	Description	Possible Values
EccCustom	Master configuration file	Ecc6Initialize.ini
EccUserRoot	Root of the user directory	D:\EccUser
CatiaEnv	CATIA environment	CATIA_ECC
EdbV	EDB version	Plm61
Csp	CATIA service pack	r13spx
CatiaBin	Directory of the CATIA installation binaries	E:\DessaultSystemes\B16\intel_a\codes\bin

Script plm61.cfg – Variables to Determine EDB Conditions

Variable	Description	Possible Values
EdbDrive	Drive for installing Agile e6	F:\
EdbPath	Path for installing Agile e6	F:\Agile_e6\ext\bin\intel-ms-nt5.0
EdbWeiling	Path for installing Agile e6	F:\Agile_e6\axalant\bin\intel-ms-nt5.0
EdbServer	EDB server	%COMPUTERNAME%
EdbApplication	EDB application	demoedb
EdbUser	EDB user	EDBCUSTO

Script eciplm50.cfg – Variables to Determine ECI Conditions

Variable	Description	Possible Values
EccCadimClient	Computer on which the Agile e6.1 client is running	Local host
EccEciTopic	Logical link name	3334
EccEciTransport	Transport type	1

## Master Configuration File Ecc6Initialize.ini

After the integration is set up correctly, the behavior can be configured.

Configuration settings as well as customization is carried out in the mapping files located in the directory ...ecc6\ini\.

**Note** Generally, mapping files must not contain empty rows!!!

The most important settings for the functioning of the Agile-CATIA integration are given in the master configuration file, Ecc6Initialize.ini.

Section	Key	Possible Values	Description
[ Initialize ]			
basic values	EccCustomerId	None	Customer identification.
	EccLanguage	English, German	Language setting.
	EccMappingFile	Ecc6.ini	Name of the basic configuration file.
	EccCustomerFile	Ecc6Customer.ini	Name of the customer specific configuration file.
	EccMessages	Ecc6Messages.ini	Name of the message file.
	EccFileInfo	Root   Type	Information about the memory location of the file can be found either in the document root (Root) or in the document type (Type).
	EccFileServer	Vault	Switch to use the file server.
	CatiaVersion	CATIA_V5	Identification string of the CAD system.
	CatiaV4Version	CATIA CATIA_V4.2 Catia_4.2.4	Identification string of the CAD system CATIAV4.
	EccCheckinArea	CAX	Defines the logical name of the vault.
	FrameDirectory_windows /_unix	[ format %s\Frames \$env(EccRoot) ]	Directory of the drawing frames (Windows, Unix).
	EccFormatDefault	ECC6-DEFAULT	Label of the name format.
	EccFileNameGeneration	IEF, PAL, SuppressRename	PAL = Definition in Ecc6 integration IEF = Definition in PLM integration SuppressRename = File renaming is suppressed
	EccReplaceSpace	0, 1	1 = Replaces spaces in file names with the character of value of the key "EccReplaceCharacter" 0 = Does not replace spaces
	EccReplaceInch	0, 1	1 = Replaces spaces in file names with the character of the value of the key "EccReplaceCharacter" 0 = Does not replace spaces



Section	Key	Possible Values	Description
	EccCharactersToReplace	° § \$ % & / , @ .....	Any character can be added to this list. The format of some of these characters requires that the character be preceded with '\\\' (double backslash) as shown below: \\° \\§ \\\$ % & / , @ : \\( \\) \\! \\? ` ~ ' > < ... Replaces the mentioned special characters in file names with the character of the value of the key "EccReplaceCharacter".
	EccReplaceCharacter	-	This character is entered as replace character.
	EccProcessAdditionalFiles	.wpc .act .cgr	These file types are also loaded from vault if existing.
	EccIgnoreCheckinError	.wpc .act .cgr .xls .txt	For these file types generated errors are ignored during check-in.
	EccIgnoreCheckoutError	.xls .txt	For these file types generated errors are ignored during check-out.
	EccPreviewSize	175	Size of preview in pixel.
	EccPreviewFormat	JPEG	File format of preview.
	EccPreviewExtension	jpg	File extension of preview.
	EccQuickLoadType	SMT	Sets the used QuickLoad type.
	EccDeltaTime	20 (9000000000 stands for Deactivation)	Tolerance range in milliseconds from the point the integration assumes that an object has changed. This is important for file systems without time synchronization.
	EccDeltaModificationTime	30	For internal use only.
	CatiaV5ReleaseEnvVar	CATIAV5_ENV	Enables the handling of several CATIA environments.
	EccActiveChangeManagement	0, 1	For internal use only.
Type and Mask			
Load Lists			
	Ecc.DocumentLoadList	EDB-DOC-SLI	Loads the Agile e6.1 form to enable searching the documents belonging to the CAD object.
	Ecc.ItemLoadList	EDB-ART-SLI	Loads the Agile e6.1 form to enable searching the items belonging to the CAD object.
Documents			


Section	Key	Possible Values	Description
	Ecc.CATDrawingType	DRAWING	Document type for drawings
	Ecc.CATPartType	3D_MODEL	Document type for parts
	Ecc.CATProductType	3D_MODEL	Document type for products
	Ecc.CATProcessType	3D_MODEL	Document type for processes
	Ecc.CATAnalysisType	3D_MODEL	Document type for analyses
	Ecc.modelType	3D_MODEL	Document type for CATIAV4 3D models
	Ecc.drawingType	DRAWING	Document type for CATIAV4 drawings
	Ecc.xlsType	3D_MODEL	Document type for design tables
	Ecc.txtType	3D_MODEL	Document type for design tables
	Ecc.CATShapeType	3D_MODEL	Document type for shapes
Lists			
	Ecc.CATDrawingList	EDB-DOC-DRW-TLI	List mask for drawings
	EccCATPartList	EDB-DOC-3DM-TLI	List mask for parts
	EccCATProductList	EDB-DOC-3DM-TLI	List mask for products
	Ecc.CATProcessList	EDB-DOC-3DM-TLI	List mask for processes
	Ecc.CATAnalysisList	EDB-DOC-3DM-TLI	List mask for analyses
	Ecc.modelList	EDB-DOC-3DM-TLI	List mask for CATIAV4 3D models
	Ecc.drawingList	EDB-DOC-3DM-TLI	List mask for CATIAV4 drawings
	Ecc.xlsList	EDB-DOC-3DM-TLI	List mask for design tables
	Ecc.txtList	EDB-DOC-3DM-TLI	List mask for design tables
	Ecc.CATShapeList	EDB-DOC-3DM-TLI	List mask for shapes
Forms			
	EccCATDrawingForm	EDB-DOC-DRW-TFR	Form for drawings
	EccCATPartForm	EDB-DOC-3DM-TFR	Form for parts
	EccCATProductForm	EDB-DOC-3DM-TFR	Form for products
	Ecc.CATProcessForm	EDB-DOC-3DM-TFR	Form for processes
	Ecc.CATAnalysisForm	EDB-DOC-3DM-TFR	Form for analyses
	Ecc.modelForm	EDB-DOC-3DM-TFR	Form for CATIAV4 3D models

<b>Section</b>	<b>Key</b>	<b>Possible Values</b>	<b>Description</b>
	Ecc.drawingForm	EDB-DOC-3DM-TFR	Form for CATIAV4 drawings
	Ecc.xlsForm	EDB-DOC-3DM-TFR	Form for design tables in MS Excel format
	Ecc.txtForm	EDB-DOC-3DM-TFR	Form for design tables in text format
	Ecc.CATShapeForm	EDB-DOC-3DM-TFR	Form for shapes
	EccltemUpdate	EDB-ART-CFR	Name of the mask for updating the items
	EccltemCreate	EDB-ART-CFR	Name of the mask for creating the items
	EccltemDisplay	EDB-ART-CFR	Name of the mask for displaying the items
Fields			
	Ecc.CATDrawingFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of drawings.
	Ecc.CATPartFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of parts.
	Ecc.CATProductFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of products.
	Ecc.CATProcessFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of processes.
	Ecc.CATAnalysisFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of analyses.
	Ecc.modelFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of models.
	Ecc.MODELFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of 3D object file.
	Ecc.drawingFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of drawings.
	Ecc.xlsFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of Excel sheet files.
	Ecc.txtFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of text files.
	Ecc.CATShapeFieldForFileName	T_DOC_DAT.CAX_FILE_NAME	Name of the field for the file name of shapes.
	Ecc.CATDrawingFieldForCaxType	T_DOC_DAT.CAX_TYPE	Name of the field for the object type of drawings.
	Ecc.CATPartFieldForCaxType	T_DOC_DAT.CAX_TYPE	Name of the field for the object type of parts.
	Ecc.CATProductFieldForCaxType	T_DOC_DAT.CAX_TYPE	Name of the field for the object type of products.
	Ecc.CATProcessFieldForCaxType	T_DOC_DAT.CAX_TYPE	Name of the field for the object type of processes.

Section	Key	Possible Values	Description
	Ecc.CATAnalysisFieldForCaxType	T_DOC_DAT.CAX_TY PE	Name of the field for the object type of analyses.
	Ecc.modelFieldForCaxType	T_DOC_DAT.CAX_TY PE	Name of the field for the object type of models.
	Ecc.MODELFieldForCaxType	T_DOC_DAT.CAX_TY PE	See description above.
	Ecc.drawingFieldForCaxType	T_DOC_DAT.CAX_TY PE	Name of the field for the object type of drawings.
	Ecc.xlsFieldForCaxType	T_DOC_DAT.CAX_TY PE	Name of the field for the object type of Excel sheet files.
	Ecc.txtFieldForCaxType	T_DOC_DAT.CAX_TY PE	Name of the field for the object type of text files.
	Ecc.CATShapeFieldForCaxType	T_DOC_DAT.CAX_TY PE	Name of the field for the object type of shapes.
	EccCATDrawingFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of drawings.
	EccCATPartFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of parts.
	EccCATProductFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of products.
	Ecc.CATProcessFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of processes.
	Ecc.CATAnalysisFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of analyses.
	EccMODELFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of models.
	Ecc.modelFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of models.
	Ecc.drawingFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of drawings.
	Ecc.xlsFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of xls-files.
	Ecc.txtFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of txt-files.
	Ecc.CATShapeFieldForFilePath	T_DOC_DAT.CAX_FIL _PATH	Name of the field for the directory of products.
	Ecc.CATDrawingFieldForFrame	T_DOC_DAT.CAX_FR AME_ID	For internal use only.
	Ecc.CATFieldForCATEnv	T_DOC_DAT.CAX_CA TENV	Name of the field for the environment name.

Section	Key	Possible Values	Description
BOM			
	AuxPartIdentifier	AUX	Definition of auxiliary parts, no item allocation.
	AuxPartDocumentField	T_DOC_DAT.AUX_PART_IDENT	Name of the field for defining an auxiliary part.
Options			
	EccCreatePreview	0, 1	1 = Preview (blob) is created 0 = No preview is created
	EccEnableRename	0, 1	1 = Renames the file name when changing the document id 0 = No rename of the file name
	EccUpdateFrameAfterLoading	0, 1	1 = Updates frame after EdbLoad 0 = Frame is not updated
	EccUpdateFrameBeforeCheckin	0, 1	1 = Frame update before check-in 0 = Frame is not updated
	EccForceReplaceFrame	0, 1	For internal use only
	EccDevelopMode	0, 1	1 = More detailed error output 0 = No error output
	EccEnableBaselines	0, 1	1 = Baseline facility enabled 0 = Baseline facility disabled
	EccEnableStandardParts	0, 1	1 = Normpart facility enabled 0 = Normpart facility disabled
	UsePermanentConnect	0, 1	For internal use only.
	EccEnableSheetManagement	0, 1	1 = Multisheet management enabled 0 = Multisheet management disabled
	EccEnableDesignTableManagement	0, 1	1 = Design table management enabled 0 = Design table management disabled
	EccEnableCache	0, 1	1 = Option for EdbLoad
	EccExtendedProperties	0, 1, 2, 3	0: Product Properties; 1: + Userdefined Properties; 2: + Mechanical Properties; 3: All Properties
	EccQuickLoad	0, 1	1 = A stored procedure is used for file types mentioned in EccQuickLoadType 0 = No stored procedure is used for file types mentioned in EccQuickLoadType
	EccEnableConcurrentCheck	0, 1	1 = A check management ensures parallel working 0 = No check management ensures parallel working

Section	Key	Possible Values	Description
	EccStepCompliantStructure	0, 1	1 = Transforms matrix for each object instance because multiple referenced components are correctly handled in the document structure 0 = No transformation
	EccUseFmsDll	0	For internal use only.
	EccUseAuthentication	0, 1	1 = Used for login when using a "Secured ECI connect" 0 = Not used for login
	EccDisableErrorOnGlobalView	0, 1	1 = EdbLoad disable error in global view 0 = EdbLoad generates error in global view
	EccAllowBatchCreate	0, 1	1 = Creation of CATIA-documents in batch mode is enabled 0 = Creation of CATIA-documents in batch mode is disabled
	EccAutoReconfigure	0, 1	1 =Reconfigures automatically before EdbSave 0 = No automatic reconfiguring
	EccAutoLoadParts	0, 1	1 = Loads "unloaded" parts is enabled 0 = Loads "unloaded" parts is disabled
	EccIgnoreStepForDrawings	0, 1	1 = Administrates DrawingViews; refers to EccStepCompliantStructure
	EccIgnoreStepForParts	0, 1	1 = Administrates PartViews, refers to EccStepCompliantStructure
	EccCncReadOnly	0, 1, 2	0 = No reservation management when parallel working is used 1 = Uses reservation management when parallel working is used 2 = For internal use only
	EccSuppressReplaceBackgroundView	0, 1	1 = Suppresses replacement of drawing-backgrounds at EdbUpdateFrame 0 = Replacement of drawing background is not suppressed
	EccWriteHistoryAfterCheckin	0, 1	0 = Writes no history information into metadata set after check-in 1 = Writes history information into metadata set after check-in
	EccCheckCatiaEnvironment	0, 1	1 = CATIA environment is checked 0 = CATIA environment is not checked
	EccUseIconsInList	0, 1	0 = No icons will be used in SaveManager 1 = Icons will be used in SaveManager
	EccI18NSupport	0, 1	1 = I18N facility is enabled 0 = I18N facility is disabled

Section	Key	Possible Values	Description
	EccCompareVersion	0, 1	1 = Comparing the assembly structure of the current and existing version is enabled 0 = Disabled
	EccUseBatchCheckin	0, 1	1 = Activates the batch checkin functionality 0  = Disabled This option is available for Windows only.
	EccSupressWindow handling	0, 1	1 = Deactivates the window handling. This is useful for a two screen workstation 0 = Activates the window handling.
	EccCreatePreviewECI	0, 1	1 = Uses ECI functionality to save the previews. 0 = Uses LogiView procedure to save the previews.
	EccUseEciMulSel	0, 1	1 = Activates the eci_mul_sel functionality. Available for e6.x only. Enabling this option increases the performance of the "Save" function 0 = Deactivates the eci_mul_sel functionality.
	EccUseOptimizedAccessCtrl	0, 1	1 = Activates an optimized check of the access rights of the documents. Enabling this option increases the performance of the "Save" function. 0 = Deactivates the optimized check of the access rights of the documents.

## Options for EdbUser Settings

Important customer specific settings (e.g. customer specific userexits) should also be applied to the master configuration file Ecc6Initialize.ini in section [CustomerFunction].

Proposals for the following userexit defining variables are provided as commented lines in the standard state of the integration because the concrete behavior of the respective userexit has to be adapted:

Variable	Used values
CatiaCreateDrawing-CatiaOpenFile-01	StandardCATIAOpenFile
CatiaLoadDrawing-CatiaOpenFile-01	-
CatiaCreateObject-GetPath-01	-
CatiaCreateObject-GetPath-02	-
CatiaCreateDrawing-ModifyDrawingMask-01	-
CatiaCreateDrawing-ModifyDrawingMask-02	-
CatiaUpdateObject-ModifyTable-01	-
CatiaUpdateObject-ModifyTable-02	-

CatiaUpdateFrame-ModifyTable-01	-
---------------------------------	---

In the standard state of the integration only the variable CatiaGetDefaults-ModelForDrawing-01 is activated using the userexit "StandardModelForDrawing". This userexit can also be adapted.

## Internal Structure of Mapping Files

A mapping file describes any number of “quasi-static” parameters in a text file, i.e. these parameters can be determined during the project introduction phase.

The structure of a mapping file is as follows:

```
[ SectionName ]
Attribute = Value
...
# : Comment symbol
```

The syntax of the attributes is:

```
<key_value>:<attribute_value>
```

The <key\_value> identifies the row, i.e. any sequence of attribute/value rows can be selected within a section in the mapping file. A particular row is ignored if invalid values are selected for <key\_value>. The first entry of a <key\_value> in the segment of a section is always valid unless otherwise specified. The following <key\_values> can be used:

Accessing the individual Objects

Key Value	Description
DEF	Default values for fields (multiple entries allowed)
DLM	Flag for deleting the trash after the <onloff> operation
ENT	Name of the entity
MAS	Name of the mask
RES	Reservation identifier <onloff>
RET	Return value (multiple entries allowed)
RMN	An <onloff> switch to indicate if the widget should be closed after use
SEL	Selection criteria (multiple entries allowed)
SYS	An <onloff> switch to access system fields
TYP	Type name of an entity
WDG	Type of the widget <listform>

Accessing the Object Relations

Key Value	Description
ENT1	Name of the parent entity
ENT2	Name of the child entity
MAS1	Name of the parent mask
MAS2	Name of the child mask



RDEF	Default values for fields in the relation mask (multiple entries allowed)
RDEL	Flag for deleting the trash after the <onloff> operation
RES1	Reservation identifier for the parent widget <onloff>
RMAS	Name of the relation mask
RRET	Return value of the relation mask (multiple entries allowed)
RSYS	An <onloff> switch to access system fields in the relation mask
RWDG	Type of the relation widget <listform>
SEL1	Selection criteria in the parent widget (multiple entries allowed)
SEL2	Selection criteria in the child widget (multiple entries allowed)
SYS1	An <onloff> switch to access system fields in the parent mask
TYPE	Type of the relation
VIEW	View name of the relation
WDG1	Type of the parent widget <listform>

## Basic Configuration File Ecc6.ini

The name of the actual configuration file Ecc6.ini has been defined in the master configuration file Ecc6Initialize.ini. It contains standard parameters for the respective integration functions.

## Customer Configuration File Ecc6Customer.ini

The name of the actual configuration file Ecc6Customer.ini has been defined in the master configuration file Ecc6Initialize.ini. It contains customer specific values for creating the Agile e6.1 objects and definitions for the drawing frames. It is recommended to determine default settings in this file.

The segments like [Default.CATProduct], [Default.CATPart], [Default.CATDrawing], [Default.CATAnalysis], or [Default.CATProcess] determine default values and formats (e.g. DEF:T\_DOC\_DAT.DOCUMENT\_ID = F:%-20.20s:Name) for the particular CATIA objects. These values are used as default values when creating the Agile e6.1 documents.

The internal structure of the value starts with a capital letter followed by a colon.

- E: = Environment variable
- F: = Format
- V: = Presetting
- R: = Returned value

Transferring defaults:

```
DEF:<table>.<field> = V:<value>
```

Transferring the environment variables of the current session:

```
DEF:<table>.<field> = E:<env-variable>
```

Accessing the attributes of CATIA objects:

DEF:<table>.<field> = <Name, Type, Dir>

Attribute Name	Description
Name	Name of the CATIA object
Type	Type of the CATIA object
Dir	Path of the CATIA object

Formatting the attributes or defaults:

DEF:<table>.<field> = F:%-20.20s:Name (number of characters)

DEF:<table>.<field> = R:9.36:Name (characters from - to)

Segments like [Duplicate.CATProduct], [Duplicate.CATPart], [Duplicate.CATDrawing], [Default.CATAnalysis], or [Default.CATProcess] determine values that are transferred as default when saving the original document in Agile e6.1 using the “EdbSaveAs” function:

RET:DEF:<table>.<field> = <table>.<field>

The content of the <table>.<field> elements to the left define the field in the target document and those to the right define the field in the source document.

# Parameter Transfer

## Transfer from CATIA to Agile e6.1

The integration uses the so-called standard templates for creating drawings. Text information defined with a name is included in the “Background view” of these templates. The integration uses this name to locate placeholders for storing information given in the Agile e6.1 document (metadata set) of the drawing.

Templates must be stored in the directory that has been determined in the master configuration file “Ecc6Initialize.ini” under the “FrameDirectory” keyword.

The standard directory for templates is ...\\ecc6\Frames.

The file name of a template must match the selection option (except the extension) given in the “T\_DOC\_DAT.CAX\_FRAME\_ID” and the following syntax:

<FrameType>\_<SizeIndicator>.CATDrawing

## Frame Mapping

Allocation of the Agile e6.1 information and the placeholders are implemented in the Ecc6Customer.ini mapping file using the following definitions:

1. Define the fields to be used for character field mechanism

```
Segment:          [ EccUpdateFrame ]
# Customer specific
RET:<logical name>      =      <table>.<field>
```

2. Allocate the physical object names of CATIA texts to the logical names

```
Segment [ TitleBlock_<FrameType> ]
<physical name>      =      <logical name>
```

3. Read the modified entries from the document history

```
Segment [ EccGetSimpleDocumentHistory ]
RRET:<logical name> =      T_DOC_HIS.<field>
```

An additional filter definition is required for allocating the physical names of texts to the logical names since the history is represented by a list of information:

```
Segment [ History_<FrameType> ]
<physical name> = <logical name 1 >
<value>:<logical name 2>,
<first | last>
```

This definition must be read as follows:

If the value of a field in the history of a document, which uses <logical name1>, is equal to the content of <value> then assign the text with the name <physical name> to the content of the

field <logical name 2>.

The <first | last> expression is used to determine whether the first or the last matching entry is to be used since identical entries may appear repeatedly in the history.

#### 4. Formatting text

Segment [ FormatsOfTitleBlockTexts\_<FrameType> ]  
 <physical name> = <format | procedure>

In this case <format> is the formatting specification that has to be used. This specification follows the guidelines of languages like “TCL” or “C”.

Every formatting instruction begins with the % symbol and has the following general form:

% [<flags>] [<width>] . [<precision>] [l] <type>

“-” is mostly used for <flags>. This means that the conversion result should always be left justified in the output area.

The decimal sequence number <width> determines the minimum field width. Spaces to the left are padded (or to the right in case of the “-” flag for left justification) if the converted value has less characters than the field width.

The precision field <precision> determines the number of places after the decimal point for the conversion character used, e.g. “f”.

---

**Note** In this case the letter “l” is not used for “long integer”.

---

The meaning of the conversion character <type> is as follows:

f	The float or double argument (arg) is converted to decimal display of the form “[-]ddd.ddd”. The number of characters after the decimal point is defined by the precision value. 6 digits after the decimal point are assumed if no precision value has been defined. No decimal point is displayed if the specified precision is zero.
s	The argument (arg) is interpreted as a pointer to a character string. The output contains the number of characters specified by the precision value or the number of characters until the first \0 character. If no precision value has been defined, the output contains all characters until the first \0 character. The result is undefined if the argument value is NULL.

Some examples of the parameter specifications are given below:

Value	Description
y	Fixed text entry – in this case, default value = “YES”
%s	Undefined character string (String)
%-10.2f	Left justified ten-digit numeric value with two digits after the decimal point.
%-4.0f	Left justified four-digit numeric value without any digits after the decimal point

## Drawing Sheets

The ECC integration allows managing drawing sheets.

The “EccSheetmanagment” option must be set to “1” for this purpose.

One entry is added to the document table for every particular sheet. The first sheet also has an

entry. However, values are not interpreted and those given in the document are used instead.

These fields must be included in the T\_CAX\_SHE table and the CAX-SHE-RLI mask. Details regarding the data model are given in the "...\ecc6\install\DataModel.xls" file.

This table should at least contain one entry for the format definition. This field has the same functionality as the "Ecc.CATDrawingFieldForFrame" field. It determines the template frame. Other fields can overwrite entries in the drawing frame.

These fields are defined in the "Ecc6Sheet.ini" file in the segment [EccGetSheet] (Key RRET:).

Segments like [ SheetBlock\_<typ> ] and [ FormatsOfSheetBlockTexts\_<typ> ] are used for the format definition and text allocation similar to their use in the normal drawing header definition.

## Baselines

Baselines (for more information about this see the Agile e6.1 online help documentation) are associated with the document in the form of files. Only the modified files are checked in. If a file is not modified, only a link to the file is created in the database but the file itself is not duplicated. This data model is a prerequisite for integrating baselines into the customer-specific organization within Agile e6.1. A separate function package that is independent of the CAD system and is based on LogiView has been developed for this purpose. Enhancements can be imported using an ASCII loader file. It is stored in the directory "../Loader" and has the name "baseline2.dat".

Individual tables have the following definitions:

T\_BSL\_DAT

Field Name	Type	Index	Remark
BASE_NAME	S80	BSL_IDX	
EDB_ID	I10		Number generator
DOC_IDENT	S40	BSL_IDX	C_ID document
CRE_SYSTEM	S80		Optional

Unique baseline names for a document are ensured with the help of BSL\_IDX (global unique).

T\_DOC\_BSL

Field Name	Type	Remark
BAS_REF	I10	Baseline ID (not Null)
BAS_DATE	S40	Default with @TODAY
BAS_ID	S255	Component identification
BAS_TYPE	S255	
FIL_REF	I10	File C_ID
FIL_NAME	S255	File name
CRE_SYSTEM	S80	
CAX_001	S255	Time stamp
CAX_002...5	S255	

The name of the baseline (stored in the EDB field T\_BSL\_DAT.BASE\_NAME) is displayed in the mask EDB-DOC-BSL-?LI using the T\_DOC\_BSL.BAS\_REF=T\_BSL\_DAT.EDB\_ID(+) join.

If the Baseline option is used the table T\_DOC\_FIL has to contain the additional field CAX\_BASELINE (type = S 40)

## **Comparing Assembly Versions – EccCompareVersion**

Now, the integration enables comparing the assembly structure of the current and existing version. This is done by changing to Global View and searching for a released version. The integration then combines the old and the current version to a conjoined CATProduct.

In section [EccNolef\_OLD\_VERSION] of the file Ecc6Util.ini you can define the filename structure of the old version.

Values = DocumentId DocumentVersion

Syntax = %-15.15s -Old Version- %3.3s

## Chapter 5

# Debugging

The most profound view into the integration's functioning enables tracing the ECI-calls which are sent from the local Agile client to the connected Agile server including the responds. A file is created on the Agile server machine which can be viewed by user using the client.

It is possible to start and to stop the ECI-trace at any time during the session.

The trace function is activated in Agile e6:

1. Click Tools > Trace > Select Module
2. Select the checkbox for ECI-Module E 9.
3. Click OK.
4. To start the trace log any time during the session, click Tools > Trace > Trace New.

The name of the new trace file is displayed in the message bar (e.g. Test output on D:\AgilePLM\tmp\tst173.trc).

---

**Note** The file location is on the Agile server machine!

---

5. Execute the process you want to test/trace.
6. To end tracing, click Tools > Trace > Trace Off.
7. To display the contents of the trace file in a list, click Tools > Trace > Show trace.

You can copy and paste the trace contents into an Excel file and save it locally.

---

**Note** The information in the trace file can be extended by e.g. adding SQL statements.

---

