Agile Product Lifecycle Management

Engineering Collaboration Client User Guide

v1.1 SP5

Part No. E12007-02
October 2008

Copyright and Trademarks

CONTENTS

Preface

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

Note	To read the PDF files, you must use the free Adobe Acrobat Readerâ„¢ version 7.0 or later. This program can be downloaded from the Adobe Web site http://www.adobe.com.

The Oracle Technology Network (OTN) Web site http://www.oracle.com/technology/documentation/agile.html can be accessed through Help > Manuals in both the Agile Web Client and the Agile Java Client. If you need additional assistance or information, please contact support http://www.oracle.com/agile/support.html (http://www.oracle.com/agile/support.html) for assistance.

Note	Before calling Agile Support about a problem with an Agile PLM 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 Agile PLM can be found in the Readme file on the Oracle Technology Network (OTN) Web site http://www.oracle.com/technology/documentation/agile.html

Agile Training Aids

Go to the Oracle University Web page 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.

Chapter 1

Introduction to Engineering Collaboration

This chapter includes the following:

Overview

Agile Engineering Collaboration (EC) is an application that provides data and process integration between CAD applications and Agile PLM. The application consists of a core Engineering Collaboration Client (known as the "EC Client"), and individual MCAD Connectors to specific CAD systems (such as Pro/ENGINEER, SolidWorks, etc.). It allows CAD designers and engineers to capture and control the data representing a primary source of the product record. The EC Client provides a window into the Agile environment that is geared towards CAD designers and engineers. It supports searching and viewing of Agile data, and provides the user interface for all Connector operations such as saving and loading CAD data.

Each captured CAD file is stored in a PLM object. This object can be a Document, or as a new option available with Agile PLM 9.2.2.4 release, a Design object. With Agile 9.2.2.4, a configuration option allows setting the EC Client to work with either object class.

You can view CAD datasets created by the EC Connectors using Agile Enterprise Visualization (EV) allowing anyone with access to Agile PLM to be able to view, markup, and collaborate in real time on the 3D CAD designs across the web, without using the CAD tool.

Agile Engineering Collaboration requires Agile Product Collaboration (PC) as a base module, which provides the underlying item management and change management required to manage CAD designs.

Key Features

The main features of the EC Connectors are:

- Save - Saves native CAD data from the current session into Agile
- Load - Loads native CAD data from Agile into the current CAD session
- Manage Change - Allows users to control checkout reservation and revisioning
- Update Properties - Updates property (attribute) values between the CAD files and Agile
- Create Viewables - Creates neutral format files, such as PDF, to be used for viewing, plotting, or manufacturing

- BOM Publication - Automatically creates and updates the Part BOM based on the CAD design structure, with Agile change control.

Some of the characteristics of Agile Engineering Collaboration are:

- Multi platform, Java Client.
- Centralized configuration via XML.
- Dynamically loaded sessions support customization of the CAD connector's business logic.
- Simplified Common Search and View functionality for CAD designers and engineers.
- Manages both work in progress (WIP) and released CAD design data.
- Supports concurrent engineering, enabling multiple designers to work with a common assembly.
- Data access controlled by checkout reservation and/or timestamp.
- Automated BOM creation process.
- Bi-directional metadata exchange.
- Support for special CAD capabilities such as Pro/ENGINEER family tables, CATIA CGR fields and SolidWorks configurations.

Agile PLM Documentation

This guide is intended to be sufficient to use Engineering Collaboration, provided Agile PLM is installed at your company. If you need to install Agile PLM, please contact your Agile representative or use appropriate installation documentation.

Given below are lists of Documentation manuals, which will help you with Agile PLM installation and upgrade procedures.

- *Agile PLM Readme*
- *Agile Database Installation Guide*
- *Agile Database Upgrade Guide.*
- *Installing Agile PLM for OAS.*
- *Installing Agile PLM for WebLogic.*
- *Agile PLM Administrator Guide* - complete reference to the Administrator component of Java Client

Documentation for Agile PLM end-users:

Getting Started with Agile PLM - this is a compendium of information about cross-platform features in Agile PLM, as well as utility applications such as Import, Export, and FileLoad. It documents the following:

- User interfaces of Java Client and Web Client
- Agile PLM concepts and terms
- Agile business objects, particularly file folders for attachments

- Searches to find data in Agile.

EC also integrates with the following Agile PLM solutions:

Agile Product Collaboration User Guide - this manual is central to implementing the change control process in
Agile, with information about items (parts and documents) and changes, such as ECOs.

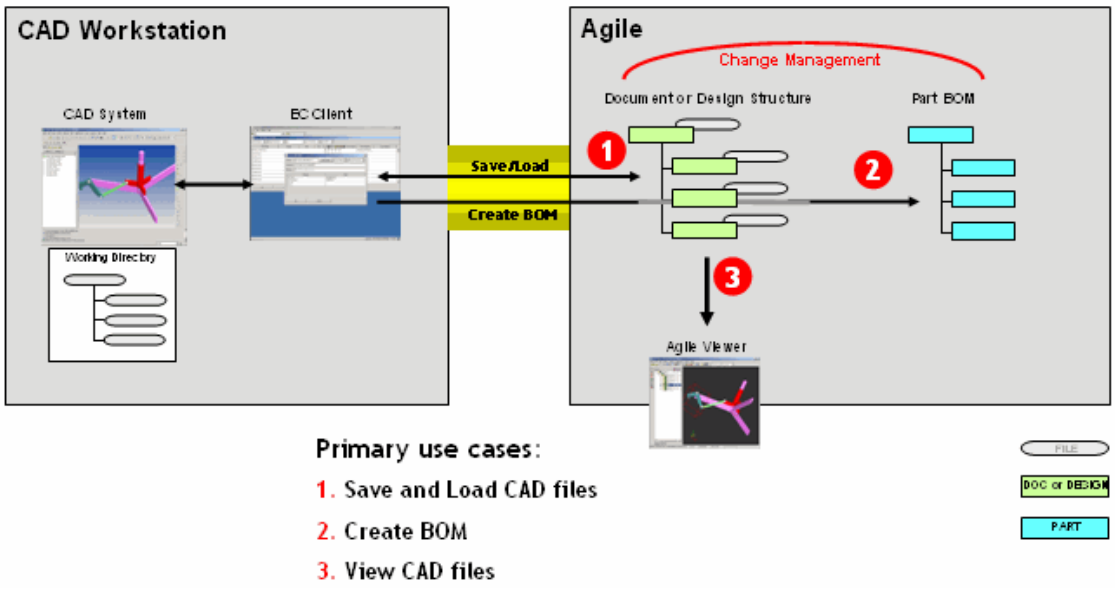
Agile Product Portfolio Management User Guide - for linking designs to projects.

Agile Product Cost Management User Guide - for cost management and supply certainty.

Agile Product Governance and Compliance User Guide - for environment rating and ensure compliance with government regulations for environment ratings.

Engineering Collaboration Process

The diagram below illustrates the main use cases supported by Agile Engineering Collaboration.



Save and Load CAD files

CAD designs are created within the CAD system environment, with files in a working directory (which may be local or network attached). The designer saves into Agile, which creates a Document or Design structure that mimics the structure of the CAD assembly. You attach the native CAD files to this structure, and use them as the basis for loading and re-saving the CAD designs. Since Agile is a centralized repository, all CAD designers in the enterprise have access to these files, subject to the control of Agile roles and privileges. Individual designers can set checkout reservation in Agile when they load files into their CAD session, enabling concurrent engineering within CAD assemblies. You can attach additional viewable files (PDF, HPGL, etc.) to the Document or Design structure.

Create BOM

When a design or design change is completed, the designer may use the Create Item/BOM command to create or update the Agile Part BOM, representing the true product structure. You use this function when there is a high correlation between the document/design structure and product structure, to avoid tedious manual entry of the Part BOM. This works in the context of an Agile change object, resulting in automated BOM redlining. You can also manually update the BOM, to add bulk items such as paint or glue. Further automated updates from CAD will not remove the manually added items.

View CAD Files

One reason for managing the native CAD files within Agile are that the Agile viewer can be used to view and markup the files. This works across the web, and without having the native CAD system. Advanced functionality such as digital mockup, 3D comparison, interference checking, and real-time collaboration make this an important tool to support the overall design process.

Chapter 2

Installation

This chapter includes the following:

The EC Client is installed automatically as part of the MCAD Connector installation. See Agile documentation "*MCAD Connectors for Agile Engineering Collaboration User Guide*" for installation instructions.

System and Version Support

For a complete and up-to-date list of officially supported Agile PLM versions for each EC release, supported CAD systems and versions for each EC CAD Connector, see Obtaining Software from Oracle E-delivery.

Prerequisites

Prior to the installation of the Engineering Collaboration interface on a local system, you must verify the following items:

- Database is operational and running
- Agile PLM (see Introduction for supported versions) is successfully installed on an accessible server (the prerequisites for Java Runtime Environment are the same as for Agile PLM server).

Important If you are not working with a member of Agile's Solutions Delivery Organization, you are strongly encouraged to refer Oracle | Agile Product Lifecycle Management Documentation for installation procedures.

- Agile File Management Server is usable and accessible.
- A test environment is prepared
- You install a CAD system, and the test user from the home directory can launch it.
- Agile PLM recognizes the login name and password of the Agile PLM test user.
- The test user can launch an Agile PLM client session.

Once you set these prerequisites, it is time to install Engineering Collaboration Client EC 1.1 SP5 for Agile PLM.

Java Version Support

The EC Client runs on Java 1.4.2 (for Agile 9.2.2.3 and earlier) and Java 1.5 (for Agile 9.2.2.4 and later). The appropriate JRE environments are bundled with the MCAD Connectors and hence available for the EC Client. For details on JRE configuration, see "*MCAD Connectors for Agile Engineering Collaboration User Guide*".

Obtaining Software from Oracle E-Delivery

Oracle products are distributed as Media Packs on Obtaining Software from Oracle E-delivery (http://edelivery.oracle.com). A Media Pack is an electronic version of the software. Refer to the Media Pack description or the list of products that you purchased on your Oracle Ordering Document. Then, view the Quick Install Guide License List to help you decide which Product Pack you need to select in order to search for the appropriate Media Pack(s) to download. Prior to downloading, verify that the product you are looking for is in the License and Options section of the E-Pack README. Oracle recommends that you print the README for reference.

There will be an itemized part list within each of the packs and you will need to download all items in order to have the complete download for the desired Oracle Agile release.

All Oracle E-Delivery files have been archived using Info-ZIP's highly portable Zip utility. After downloading one or more of the archives, you will need the UnZip utility or the WinZip utility to extract the files. You must unzip the archive on the platform for which it was intended. Verify that the file size of your downloaded file matches the file size displayed on E-Delivery. Unzip each Zip file to its own temporary directory.

Installing EC Client

The EC Client is installed automatically as part of the MCAD Connector installation. See the documentation "*MCAD Connectors for Agile Engineering Collaboration User Guide*" for installation instructions.

EC Client Configuration Options

Configuration File - CAXClient_{type}.xml

Regardless of which CAD Connector you are using, the main user interface is the EC Client. The EC Client is configured by editing an XML file called CAXClient_{type}.xml, where {type} is either Documents or Designs. Each file, depending on the connector you choose to work with, is located in one of the following locations.

CAD Connector	Location
Pro/E	AgileEC\acp\jar\Agile9
SolidWorks	AgileEC\acw\jar\Agile9
Unigraphics	AgileEC\acu\jar\Agile9

CATIAv5	AgileEC\acc\jar\Agile9
CATIAv4	/acc-rt/jar
SolidEdge	AgileEC\ace\jar\Agile9

In each case, this file is configured in exactly the same way. In fact, if you use more than one CAD Connector at your site, you can use the same configuration of CAXClient_{type}.xml for each.

This configuration file is used to:-

- Control the behavior of EC Client
- Define the data model used for storing CAD data in Agile.

Since the EC Client can be run with one of two data models (Documents or Designs) there are two independent XML files. The file to be used is determined by the "CAXCLIENTXML" parameter in the CAXClient.batscript.

clientConfig Parameters

Since the EC Client can be run with one of two data models (Documents or designs), there are two independent XML files. The file to be used is determined by the 'CAXCLIENTXML" parameter in the CAXClient.bat script.

The tag <clientConfig> indicates the parameters in this section. Configure these parameters as desired to match system and process requirements at your site.

clientConfig section

Name	Value	Remarks
userName	cax	default user (filled in login form)
serverURL	http://localhost:8888/Agile	used server connect string
dateFormat	yyyy-MM-dd HH:mm:ss z	default date format
enableFields	ON | OFF	If ON, will automatically turn on the necessary attributes configured for the sub-classes defined in CAXClient.xml (does not name the attributes however).
editMode	ON | OFF	Allows editing within forms
ping	ON | OFF	Helps to keep the client session alive under certain conditions
ObjectCache	ON | OFF	ADVANCED OPTION - Contact Agile Solution Delivery for assistance
SessionXMLCache	ON | OFF	ADVANCED OPTION - Contact Agile Solution Delivery for assistance
CopySourceAttachmentsForward	ON | OFF	If ON, all non-SOURCE attachments will automatically be removed when creating a new pending revision using EC Client. Useful for removing viewable and baseline attachments.

Name	Value	Remarks
SingleFieldUpdate	ON | OFF	If ON, updates objects one attribute at a time, rather than all at once. Required only to bypass an update problem on AIX JRE platform.
drsListenPort	5112	Dresden (drs) protocol listen port (do not change)
creationMode	INT | BAT	Default mode (possible is INT or BAT) for creation of new objects in Save Command (interactive or batch-mode)
creationModePart	INT | BAT	Default mode (possible is INT or BAT) for creation of new parts in create Item BOM command (interactive or batch-mode)
revisionSequence (also revisionSequenceECO, revisionSequenceDVO etc.)	Examples: ,A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z, ,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,	Default Sequence of Revision letters. Keep a comma at the beginning and the end of the sequence. You can create multiple entries of this line, by appending the name of an Agile Change sub-class on the end. That will assign the given revision sequence to that sub-class only.
revisionSequenceEditor	ON | OFF	If ON - allows selecting a revision from the pre-defined revisionSequence list in Manage Change. If OFF - revision must be manually entered.
EcoDefaultWorkflow	Examples: "Default Change Orders" "{class} Default Workflow"	The name of the workflow to be automatically assigned when creating a Change object, ECO in Manage Change. If the name contains the string "{class}", the name of the Change sub-class is inserted. This allows assigning a unique workflow per Change sub-class.
EcoIgnoreRuleForLatest	ON | OFF	If ON - allows a non-latest revision to be selected for creation of a pending revision, in the Manage Change command (although a warning icon will be displayed). This supports design branching. If OFF - attempting to create a pending revision from a non-latest revision will not work (a stop sign will be displayed).

Name	Value	Remarks
NumberingMode	Allowable values: NONE Pro/E UG CATIA SOLIDWORKS SOLIDEDGE	If set to a value other than NONE, when creating a new CAD model using the Agile > New command, the appropriate CAD extension for the given CAD system will be appended to the autonumber from Agile, to create the Document number. The value of NumberingDelim is also inserted (see next entry). For example, if this is set to SOLIDWORKS, and the next available autonumber in the New dialog is P12345, the Document number will be P12345.SLDPRT (instead of P12345).
NumberingDelim	Examples: "." (period) "-" (dash)	The value is inserted between the autonumber value and the value of the NumberingMode parameter (see above).
NumberingCreateParts	ON | OFF	If ON - A Part object is automatically created during the New command, and linked to the created Document. The Part number will match the selected autonumber. In the example above, the Part will be P12345 and the Document will be P12345.SLDPRT.
InitialECOClass	Allowable values: NONE A name of a Change sub-class	If set to something other than NONE, when Documents are first created in the EC Client (using New or Save), they will be automatically placed on a Change object, using this sub-class.
InitialRev	Examples: "1" "A" "-" (dash)	If InitialECOClass is not NONE, this will be the value of the initial pending revision created for Documents during New or Save.
InitialPartECOClass	Allowable values: 32105 A name of a Change sub-class	If set to something other than NONE, when Parts are first created in the EC Client (using New, Save, or Create Item/BOM), they will be automatically placed on a Change object, using this sub-class.
InitialPartRev	Examples: "1" "A" "-" (dash)	If InitialPartECOClass is not NONE, this will be the value of the initial pending revision created for Parts during New, Save, or Create Item/BOM.
AutoCreateECO	ON | OFF	If ON, automatically creates Change objects and assigns revision values based on values coming from CAD. Used for dataload operations.

Name	Value	Remarks
AddPartToECO	ON | OFF	If ON, and if Parts are automatically generated using the NumberingCreateParts option, then the Parts will automatically be assigned as affected items on the same Change as the Documents, in the Manage Change command.
assignDrawingToPart	ON | OFF	If ON, will link the 2D drawing Document as well as the 3D model Document into the Part BOM, during the Create Item/BOM command. Is only valid if disableDocumentPartLink is OFF.
disableDocumentPartLink	ON | OFF	If ON, the Create Item/BOM command will not link the Documents into the Part BOM, but will instead set attributes on each object, referencing each other (for 9.2.1) or Relationship links (for 9.2.2).
checkLocalStamp	0 | 1	ADVANCED OPTION - Contact Agile Solution Delivery for assistance
overWrite	ASK | ON | OFF	Sets the default value of the Assignment selector found at the top of the Save dialog ASK = Confirm ON = Assign and Overwrite OFF = Don't Assign or Overwrite
autoSaveBatch	0 | 1	If set to 1- saves in batch mode with no client interaction. Used for dataload operations.
mapAttributesToFileFolder	ON | OFF	If ON, will set attributes on the File Folder object attached to the Document, that are equal to the attributes set on the Document. This requires configuration of the File Folder section of the CAXClient.xml file. This is used to better control access at the File Folder level.
getFilesSelective	0 | 1	If set to1- an additional check box column is enabled in the Load dialog that allows users to select which files to load. CAUTION: This capability is only supported by some CAD systems.
baselineColumn	0 | 1	If set to 0 - the Create Baseline check box column in the Save dialog is removed, preventing users from creating Baselines. If set to 1- Displays a save Baseline column in Save dialog.
DefaultUnit	Metric | Inches	Sets the default units for the New dialog

Name	Value	Remarks
LoadOptionsLatest	Allowable values (one or more): LATEST_PENDING RELEASED ASSAVED	Sets valid options for structure resolution during Load command, for a pending revision that is selected. The user will receive a pop-up dialog prompting them to select an option.
LoadOptionsNonLatest	Allowable values (one or more): LATEST_PENDING RELEASED ASSAVED	Sets valid options for structure resolution during Load command, for a released revision that is selected. The user will receive a popup dialog prompting them to select an option.
LoadSimplified	0 | 1 | NONE	Enables the "Simplified Reps" checkbox in the Load dialog, which is only valid for Pro/ENGINEER. If set to 0 - the checkbox is enabled and not checked. If set to 1 - the checkbox is enabled and checked. If set to NONE - the checkbox is disabled.
ViewablesDrawing	Example: TIF,PDF,CGM,PDF,JPG,HPGL,PS,DXF,DWG,EDRW,CALS	The list of allowable viewable filetypes for Drawings that will be on display in the Save Preferences dialog. Note that further configuration may be necessary to have the desired filetype be generated.
ViewablesModel	Example: JT,CGR,WRL,STL,X_T,X_B,STEP,IGES,EPRT,EASM,ED,EDP,EDZ,3DXML,SAT,VDA	The list of allowable viewable filetypes for Models (parts and assemblies), that will be displayed in the Save Preferences dialog. Note that further configuration may be necessary to have the desired filetype be generated.
AutoSubmitXYZ (Where XYZ is the name of a Change sub-class. Example: DVO)	Examples: Released Submitted	Gives the target workflow status for given Change sub-class, for use with "Submit Changes" option in Save dialog. The workflow must be configured with a valid path using "default next status" to target workflow status.
Viewer	Allowable values: ON OFF LOCAL	Controls the viewing function from the EC Client. If set to OFF - view button is disabled. If set to LOCAL - view button is enabled and supports launching file attachments using client-side MIME type applications. If set to ON - view button is enabled and uses the Agile Viewer.

Name	Value	Remarks
PublishModelFiles	Allowable Values PRT, JT, CGM, EPRT, EASM, SLDASM, SLDPRT	Sets which filetypes from 3D Design objects (parts and assemblies) will be attached to Part objects during the Create Item/BOM process.
PublishDrawingFiles	Allowable Values EDRW, CGM, TIF, PDF, PRT	Sets which filetypes from CAD drawings (parents of 3D models) will be attached to Part objects during the Create Item/BOM process.
Label Types	Examples DV DC	
LabelApproversXYZ (where XYZ is the name of the Label Type. Example: DV)	Examples: DV Approvers	Name of the approver group for a specific label type. When you choose this label type, Users within this group are assigned to the Routing Slip
LabelObserversXYZ (where XYZ is the name of the Label Type. Example: DV)	Examples: DV Observers	Name of the observer group for a specific label type. When you choose this label type, Users within this group are assigned to the Routing Slip.
LabelAutonumber XYZ (where XYZ is the name of the Label Type. Example:DV)	Examples: DV Label	Name of the Autonumber to use for the specific label type. Note: The prefix of the autonumber must begin with label value. For example, the autonumber prefix for DV must begin with "DV".
LabelUseRevisionLogicXYZ (where XYZ is the name of the Label Type. Example:DR	0 | 1	For this label type, will the major revision increment upon the next checkout. 0=No, 1=Yes.
DesignSequenceRevision	Examples; A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z	Revision sequence for the major revision component of the Design Revision field.
DesignVersionSequence	Either the value "NUMERIC" or an actual sequence. (similar to above)	Revision sequence for the minor revision component of the Design Revision field.
DesignVersionIndicator	Examples: "v" A v1 "" A1	Separator text between major and minor revision components of the Design Revision field.
AllApprovalsRequired	0 | 1	Controls ability to check out Design based on existing Routing Slip approval. If set to 1, all approvers have to approve. If set to 0, only one approval (with no rejections) is required.

Name	Value	Remarks
CheckOutWhileInApproval	0 | 1	If set to 1, Users are allowed to check out the next Design version even if the current one is still in approval. If set to 0 users cannot check out until approval occurs. Note: If checkout occurs, prior to approval, major revisioning will not occur.
DFM	0 | 1	If set to 1 - enables DFM capability If set to 0 - disables DFM capability.

fileOperation Parameters

The tag <fileOperation> indicates the parameters in the section. They control the behavior of the Save and Load operations. Configure them to match the process requirements at your site.

fileOperation section

Name Parameter	Value	Remarks
get		Parameters relating to the "Load" command
autoCheckout		"0" no automatic checkout (reserve) "1" automatic reserve
URLcopy	0	Obsolete
put		Parameters relating to the "Save" command
saveOption		"1" enable save if object isn't checked out by another user "2" enable save if object isn't checked out by another user and the object has not been changed in PLM (not out-of-date) "3" obsolete (out-of-date) "4" enable save only if object is checked out by current user
timeStampField	6151	Obsolete
viewable	VIEWABLE	The value that is used to identify source (native) CAD files, in the Attachment Type attribute of the Attachments tab on the Document.
source	SOURCE	The value that is used to identify viewable files, in the Attachment Type attribute of the Attachments tab on the Document.
baseline	BASELINE	The value that is used to identify baselines, in the Attachment Type attribute of the Attachments tab on the Document.
viewables	,CGM,TIF,TIFF,JPG,JPEG,GIF,PDF,JT,X_T,CGR,EPRT,EASM,EDRW	A comma-separated list of file extensions that will be flagged as viewable file types. Make sure there is a comma at the beginning and end of the list.

Name Parameter	Value	Remarks
keepCheckout	0 | 1	Default preselection of checkboxes for objects in save dialog. "1" means the file is still checked out after the Save command is completed.
checkInDefault	0 | 1	Default preselection of checkboxes for objects in save dialog, "1" means the filefolder is checked in after file transport to make changes visible to other users.

Setting EC Client Data Model

EC Client also allows you to switch between the Document and Design object based on the properties set in your configurations file. These settings pre-exist in the CAXClient_Designs.xml and CAXClient_Documents.xml, respectively.

To use Design objects:

<createObject defaultClass="FILEFOLDER"/>

<setProperties defaultClass='FILEFOLDER'/>

To use Document objects:

<createObject defaultClass="DOCUMENT"/>

<setProperties defaultClass="DOCUMENT"/>

Agile Data Model Parameters

The CAXClient_{type}.xml file defines the types of objects that are used by Agile to store your CAD data, and also which attributes from those objects are accessed. The object definitions are within the <objectProperties> section, and consist of <subclass> definitions, which define tabs and attributes.

There are three main <subclass> sections: File Folder (Design), Document, and Item.

- The File Folder subclass section is used when using Design objects to manage CAD files.
- The Document subclass section is used when using Document objects to manage CAD files.
- The Item subclass defines the object class for the Part BOM (the "Product Structure").

Subclass definitions appear as follows:

```
<subclass name="DOCUMENT" type="2" id="24147">
```

where type is the Agile object type, and id is the Agile sub-class ID. By setting the id equal to a specific Agile sub-class (such as CAD Model), you can control which Document sub-class appears in the EC Client by default.

Note	The drawback of this setting is that the sub-class ID cannot be determined directly through the Agile administrator client. However, when you run EC Client it will output a list of all object sub-class IDs in the CaxClient.log file.

Tab definitions appear as follows:

<table name="TITLE_BLOCK" id="0">

Within each tab, attribute definitions appear as follows:

<attribute name="DESCRIPTION" id="1002" set="1" get="1" mandatory="1"/>

which has the following parameters:

id = the Agile base ID of the attribute

set = 0 or 1 where 1 means to include this attribute in UI for creating or setting object properties

get = 0 or 1 where 1 means to return this attribute to the CAD Connector

mandatory = 0 of 1 where 1 means that the user cannot continue until a value is entered.

Note	The setting of "mandatory" is independent of the Agile server definition of "required" fields. For best results, mark those fields that are required by the Agile server, also as mandatory for the EC Client. Both mandatory and required fields are indicated by bold text in the EC Client user interface.

The Agile data model for EC is defined by determining which attributes to set within the CAXClient.xml file, and then configuring these attributes in Agile. These attributes are set in the Agile Java Client.

Data Model Configuration for Design Objects

If you are using Design objects as the basis of storing CAD files, most data model configurations are already in place. The main task is to re-configure the "Model Name" attribute from a text attribute to a multitext attribute.

- Rename the existing “Model Name” attribute (ID 2000008376) to a name of your choice.
- Rename the “Structure MultiText01” attribute (ID 2000008365) to “Model Name”

In the FILEFOLDER subclass section of the CAXClient.xml file, set the CAX_FILENAME entry to the new ID as follows:

```
<attribute name="CAX_FILENAME" id="2000008365" set="0" get="0"/>
```

In addition, you need to configure the Part class to use the Create Item/BOM functionality.

Data Model Configuration for Document objects

The data model configuration effort when using Document objects is more extensive. The following table lists the attributes that are contained within the CAXClient_Documents.xml file. Those attributes that have a value shown in the "Attribute Name" column should be configured in the Agile system, using the Base ID listed.

You can configure any Document sub-classs to store CAD data. The default is the standard "Document" sub-class (ID 9141).

To use a different sub-class such as CAD Model:

- Create the sub-class.
- Modify the ID of the Document sub-class line to match the actual ID, in the CAXClient_Documents.xml file.

You need not change the word "Document" in the XML file.

Note Sub-class IDs are written in the CAXClient log file.

Note The administrator must ensure that the tabs named below are set to "Visible." This is done in the specific class (Java Client > Admin > Classes > <specific class> > User-Interface Tabs > <specific tab, double-click> > set Visible property to Yes).

Attributes to be configured:

Sub class	Symbolic name (in CAXClient.xml)	Attribute name (in Agile)	Default Attribute	Base ID	Comment
Document / Page Three	CAX_CRE_SYSTEM	CAD System	PAGE_THREE.TEXT11	1585	
	CAX_FIL_NAME	CAD Filename	PAGE_THREE.TEXT12	1586	
	CAX_FIL_OLD_NAME	CAD Old Filename	PAGE_THREE.TEXT13	1587	Optional
	CAX_TIMESTAMP	CAD Timestamp	PAGE_THREE.TEXT14	1588	
	CAX_TYPE	CAD Type	PAGE_THREE.TEXT15	1589	
	CAX_SUBTYPE	CAD Subtype	PAGE_THREE.TEXT16	1590	
	CAX_FAM	CAD Family	PAGE_THREE.TEXT17	1591	
	CAX_VAR	CAD Variant	PAGE_THREE.TEXT18	1592	
	CAX_DWG_NAME	CAD Drawing Name	PAGE_THREE.TEXT19	1593	Optional
	CAX_FRAME_ID	CAD Frame ID	PAGE_THREE.TEXT20	1594	Optional
	CAX_NAM_FMT	CAD Name Format	PAGE_THREE.TEXT21	1595	Optional
	CAX_PROJECT	CAD Project Name	PAGE_THREE.TEXT22	1596	Optional
	CAX_PART	CAD Part Number	PAGE_THREE.TEXT23	1597	

Sub class	Symbolic name (in CAXClient.xml)	Attribute name (in Agile)	Default Attribute	Base ID	Comment
	CAX_FIL_PATH	CAD File Path	PAGE_TWO.MULTITEXT31	1570	Optional
	CAX_MODEL_TYPE	CAD Model Type	PAGE_THREE.TEXT01	1575	
	CAX_MODEL_REF	CAD Model Reference	PAGE_THREE.TEXT02	1576	
	CAX_LINK_TYPE	CAD Link Type	PAGE_THREE.TEXT03	1577	
	CAX_LINK_REF	CAD Link Reference	PAGE_THREE.TEXT04	1578	
Document / BOM	CAX_FILENAME	CAD Filename	BOM.MULTITEXT30	1341	
	CAX_TRANSFORM	CAD Revision	BOM.MULTITEXT31	1342	
	CAX_IDENT	CAD Ident	BOM.TEXT01	2175	
	CAX_COMPONENT	CAD Component	BOM.TEXT02	2176	
	CAX_REFERENCE	CAD Reference	BOM.TEXT03	2177	
	CAX_CONFIG	CAD Config	BOM.TEXT04	2178	
Document / Where Used Also configure these on the "Pending Change Where Used" tab	CAX_FILENAME	CAD Filename	BOM.MULTITEXT30	1413	
	CAX_TRANSFORM	CAD Revision	BOM.MULTITEXT31	1414	
	CAX_IDENT	CAD Ident	BOM.TEXT01	2216	
	CAX_COMPONENT	CAD Component	BOM.TEXT02	2217	
	CAX_REFERENCE	CAD Reference	BOM.TEXT03	2218	
	CAX_CONFIG	CAD Config	BOM.TEXT04	2219	
Document / Relationships	N/A	Part Link	RELATIONSHIP.TEXT01	5846	

Sub class	Symbolic name (in CAXClient.xml)	Attribute name (in Agile)	Default Attribute	Base ID	Comment
Document / Attachments	(none)	Attachment Type	ATTACHMENT_MAP .ATTACHMENTTYPE	4681	Add these list values: SOURCE VIEWABLE BASELINE

Attributes to be configured - Part Class for both Document and Design mode.

Sub class	Symbolic name (in CAXClient.xml)	Attribute name (in Agile)	Default Attribute	Base ID	Comment
Parts / Page Two	CAX_CRE_SYSTEM		PAGE_TWO.TEXT11	1301	
	CAX_FIL_NAME		PAGE_TWO.TEXT12	1302	
	CAX_FIL_OLD_NAME		PAGE_TWO.TEXT13	1303	Not required
	CAX_TIMESTAMP		PAGE_TWO.TEXT14	1304	Not required
	CAX_TYPE		PAGE_TWO.TEXT15	1305	
	CAX_SUBTYPE		PAGE_TWO.TEXT16	1306	
	CAX_FAM		PAGE_TWO.TEXT17	1307	Not required
	CAX_VAR		PAGE_TWO.TEXT18	1308	
	CAX_DWG_NAME		PAGE_TWO.TEXT19	1309	Not required
	CAX_FRAME_ID		PAGE_TWO.TEXT20	1310	Not required
	CAX_NAM_FMT		PAGE_TWO.TEXT21	1311	Not required
	CAX_PROJECT		PAGE_TWO.TEXT22	1312	Not required
	CAX_PUBLISHED	Published From	PAGE_TWO.TEXT23	1313	
	CAX_FIL_PATH		N/A	1331	Not required
	CAX_FIL_OLD_PATH		N/A	1332	Not required
	CAX_DOC	CAD Model	PAGE_TWO.TEXT01	2007	
	CAX_DOC_CONFIG	CAD Model Configuration	PAGE_TWO.TEXT02	2008	
Parts / BOM	CAX_FILENAME	CAD Filename	BOM.MULTITEXT30	1341	

Sub class	Symbolic name (in CAXClient.xml)	Attribute name (in Agile)	Default Attribute	Base ID	Comment
	CAX_TRANSFORM	CAD Revision	BOM.MULTITEXT31	1342	
	CAX_IDENT	CAD Ident	BOM.TEXT01	2175	
	CAX_COMPONENT	CAD Component	BOM.TEXT02	2176	
	CAX_REFERENCE	CAD Reference	BOM.TEXT03	2177	
	CAX_CONFIG	CAD Config	BOM.TEXT04	2178	
Parts / Where Used Also configure these on "Pending Change Where Used" tab	CAX_FILENAME	CAD Filename	BOM.MULTITEXT30	1413	
	CAX_TRANSFORM	CAD Revision	BOM.MULTITEXT31	1414	
	CAX_IDENT	CAD Ident	BOM.TEXT01	2216	
	CAX_COMPONENT	CAD Component	BOM.TEXT02	2217	
	CAX_REFERENCE	CAD Reference	BOM.TEXT03	2218	
	CAX_CONFIG	CAD Config	BOM.TEXT04	2219	

EC Client Log File

Each time an EC Client session is starts, the user's working directory (for SolidWorks the file is called CAXJavaStart.log instead) creates a log file called CaxClient.log. This helps us to see what is happening during the client session.

Agile Roles and Privileges

For detailed information on Agile Roles and Privileges, see the *Agile PLM Administrator Guide*.

However, given below are some aspects critical to proper operation of EC Client.

Create a role or roles for those CAD users who will create objects in Agile through Engineering Collaboration Client. They should have privileges to modify objects from any Document or Design subclass that you have created - for example, a subclass called CAD Model - to be used to store CAD data. Other users should not have Modify privileges to these subclasses.

Privileges required to access and change Document objects include:

- Modify object in a preliminary state
- Modify redlined BOM
- Check in object in a preliminary state
- Check out object in a preliminary state

You need to specifically exclude Check-in and Checkout when the object has left the preliminary state, otherwise it will be possible to modify data that is in Released status through EC Client, as well as Java Client or Web Client. When using the Design object for storing CAD data, there is a standard role called "Design Engineer" that has the proper base set of privileges for managing Designs.

EC Client Customizing

The MCAD systems call CAD Connectors. The MCAD system delivers and receives data to and from Agile using I Ag l Parameter objects. There are three different ways to modify this dataflow:

- Add your own ActionHandler to CAD Connector
- Extend your own Connector from CAD Connector
- Extend your own Connector from CAD Connector and add your own ActionHandler to CAD Connector (combination of both steps above).

1. To add your own ActionHandler:
    1. Create your own JAVA class, which extends CAXAction and implements ICAXAction. Please see example of com.Agile.cax.custom.CustomAction in CustomConnector.jar. ICAXAction implements methods, called within the workflow of, save and load processes. Here you can modify the behavior using pre-defined entries.
    2. Create your own JAVA-jar-File, which contains your class. Compile and link against the CAXConnector.jar, CAXClient.jar, CaxAglDataTypes.jar and AgileAPI.jar.
    3. Ensure your jar-file and all needed sub-jars are contained in your java class path when you start CAXClient. You can check cax_client.bat for that.
    4. Register your ActionListener for CAXConnector in CAXClient.xml. If your class is com.Agile.cax.custom.CustomAction, the line looks like this:

```
<drsExtension logicalName="com.Agile.cax.CAXConnector"
className="com.Agile.cax.CAXConnector"
actionHandler="com.Agile.cax.custom.CustomAction" >
```

Note	You can only add this ActionListener to the mentioned line. Ignore all other drsExtension tags.

After restart of CAXClient you see your registered handler in stdout.

2. To extend CAXConnector:
    1. Create your own JAVA class, which extends CAXConnector. Please see the example of com.Agile.cax.custom.CustomConnector in CustomConnector.jar. This CAXConnector implements all methods, called directly by MCAD. Here you can modify the behavior by changing the dataflow or making a complete replacement of single methods (overloading).
    2. Please follow 1.1 and 1.2 for creating and using your jar file in CAXClient.
    3. Register your connector in CAXClient.xml by creating your own drsExtension tag (see CustomConnector for example). Assumed you implemented your own Connector in class com.Agile.cax.custom.CustomConnector than the lines look like this:

    ```
    <drsExtension logicalName="com.Agile.cax.custom.CustomConnector"
    className="com.Agile.cax.custom.CustomConnector" ></drsExtension>
    ```

    4. Register your connector and the changed methods in CAXClient.xml. Each method called by MCAD has a name, which is mapped to a JAVA class in XML: <classMapping methodName="symbolicName" className="containingClass">

    If you implemented your own method "searchObject" in your connector com.Agile.cax.custom.CustomConnector, the XML entry has to be modified like this:

    ```
    <classMapping methodName="searchObject" className="
    com.Agile.cax.custom.CustomConnector">
    ```

    The available methodName's are predefined and called by CAD.

3. To extend CAXConnector and ActionHandler combined:
    1. In order to extend your own Connector and use an ActionHandler just combine the steps in 1 and 2 above.

ActionHandlers only work for a CAXConnector and can only be registered there. A "CustomConnector" may not have its own ActionHandler, but you can register this ActionHandler at the CAXConnector. Always first try to use own ActionHandlers instead of own Connectors, because making your own Connector is much more complex than making your own ActionHandler

Chapter 3

Using Engineering Collaboration Client

This chapter includes the following:

Agile Engineering Collaboration Platform provides integration between CAD systems and the Agile PLM software, and allows you to manage CAD data and processes. Since most of the operation of the EC Client is common between all the supported CAD systems, this Guide covers user operations in a common way.

EC operates from within your CAD system environment. Your administrator will provide you with a startup command or icon that will start your CAD system with EC functions enabled.

Note In order to use EC you must be a registered Agile user.

Starting EC Client

You can perform user operations that require interaction with Agile using the EC Client, a separate window that runs in combination with the CAD system. Much of the use of EC Client is controlled by commands that you run from the Agile menu in your CAD system. For example, picking the Save command will bring EC Client to the front, and bring up the save dialog.

For details about EC Client commands that are executed from the CAD system Agile menu, such as Save, Load, Manage Change, and Create Item/BOM, please see the document, "*CAD Connectors 2.5*". This chapter describes basic client aspects of EC Client.

EC Client is designed for Agile operations commonly done by Engineers and Designers. You can perform standard functions from EC Client including, searching data in Agile, and viewing forms of objects such as Documents, Designs and Parts.

The toolbar menu and its functions are listed in the below table.

Menu	Image	Description
File	 Login  Recent Visits	Allows you to connect or disconnect from Agile· Shows a window listing Agile objects recently visited.
Tools	 Quick Search  Advanced Search	Provides basic search capability Provides more advanced search capability

	Saved Search	Displays a list of saved searches
Window		Provides standard window-management capability. EC Client windows can be cascaded, tiled, docked and undocked.
Help	About Agile EC…	Provides information about the release level of the EC Client.

For more details on the file and tool functions see, "Searching for objects in Agile on page 25".

Connecting to Agile

When the user first starts the CAD system, the Agile menu is available but a connection to Agile has not yet been established. The first task is to start EC Client, using the Start Client command, unless your administrator has configured your system to start the client automatically. Once the client is open, you will need to log in to Agile. Depending on your preference, you can choose to connect using one of the methods from the table below.

To connect to the EC Client:

You can login from the EC Client window.

1. Pick the Login icon from the toolbar or choose File> Login.

You can also login from within CAD.

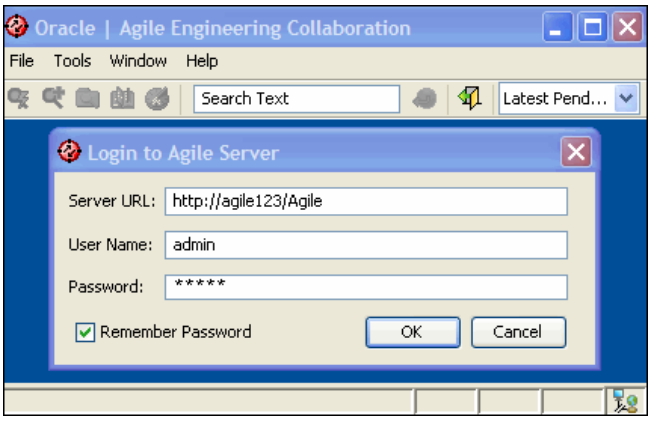
2. Pick the Agile> Connect command.

   You can also login using any Agile menu command.

3. Pick any Agile menu command that will force a login, such as Save, Load etc.

4. Log in to Agile in exactly the same way you log in to a regular Agile client, and the CAD connector's connection to Agile is established.

   After you login using one of these methods, an Agile login screen will appear. Your system administrator sets the Server URL field in the Login dialog; you would normally not need to change this.User Login



5. Enter the desired user name and password to log in. Checking the Remember Password box will

save your login parameters for future use. Click OK.

The Agile EC Client appears.

Docking and undocking in EC Client

Using the docking function of EC Client will dock the window in the tabbed docking area at the bottom of the client window.

To dock a window:

1. Locate the grayed area in the window header and right click on it.
2. Pick the Dock function.

To undock a window:

1. Right-click the grayed area of the tab.
2. Click Undock.

Disconnecting from EC Client

You can log out from EC Client without closing the client window or CAD system. There are two ways to do this:

To log out from EC Client:

1. From EC Client window, pick the Logout icon or the File > Logout... command or
2. From within CAD, pick the Agile > Disconnect command.

Once you are disconnected, you are no longer using an Agile PC license. You may continue to work with your CAD model locally, without connecting to Agile. You can re-connect to Agile later, and it is OK to leave the EC Client window active, without being connected.

To close EC Client:

1. Choose File > Exit command (from EC Client window, not from CAD).
2. Alternatively, you can click the "X" icon in the corner of the window.

It is always best to disconnect from Agile before closing the client window.

Searching for Objects in Agile

Much of the basic EC Client capability focuses on searching, so that you can find your data in Agile. There are six types of functions geared towards searching, which are Simple Search, Quick Search, Advanced Search, Saved Searches, Bookmarks and Recent Visits. All of these are similar to equivalent capabilities found in other Agile clients.

The main difference is that EC Client is limited to a sub-set of Agile object types that can be searched. This is configurable by your administrator, but by default consists of Items (including Documents, Designs, and Parts), File Folders, and Changes. These "EC Objects" are the object types used by the CAD Connectors.

Please refer the "*Getting Started with Agile PLM* " in the Agile Documentation Web site for further search details (for example, to see the allowable wildcard characters).

The table below lists a basic overview of the search function.

Search Function	Description
 Run Simple Search	Searches EC objects that match set criteria (Number, Description).
 Quick Search	Quickly searches for objects in Agile that match the set criteria. Searches for pre-defined EC objects – Items, Designs, Changes and FileFolders.
 Advanced Search	The Advanced Search dialog allows you to create more detailed searches, which can better narrow in on the desired search results. You can add multiple and specific search criteria, combining them with "And" and "Or" operators.
 Saved Searches	Saved Searches is an Agile capability which allows you to define searches and re-execute them.
 Bookmarks	The Bookmark icon lists objects which are used frequently. To Bookmark an object - Select the object you want to Bookmark and click the Bookmark icon in the object dialog that appears.
 Recent Visits	Lists 10 recently visited objects in Agile, with the most recent listed first. Enables you to easily get back to objects that you worked with before. This list is updated by both the standard Agile clients, and EC Client. However, in EC Client only the objects that you Save or Load are added to the list.

Note Agile 9.2.2 and later versions support structured bookmarks.

Agile Forms

Besides Search, the other main function of EC Client is to display forms of Agile objects. You can double-click on the name of the object from any search results list, or from the Recent Visits list, to bring up its Agile Form. This is where you can see detailed information about an object, including its attributes, relationships, and files.

Note The Agile > Show Agile Form command will pop up EC Client with the form of the active object in CAD.

You can customize every object type at your site, as every object has its own unique layout. However, it is useful to understand some of the basic layout of these Agile forms. Standard tabs for an Agile Item (a Document, Design, or Part) are these:

Form Tabs

Tab Name	Purpose
Title Block	Contains primary attributes, such as Number and Description
Page Two	Contains supplemental attributes, typically customized
Page Three	Contains additional attributes for the specific sub-class, such as CAD Model

Tab Name	Purpose
BOM	Shows the item structure below the current object (looking down the tree)
Structure	Contains information about parent object and version specific information about child objects.
Where Used	Shows all items that the current object belongs to (looking up the tree)
Routing Slip	Approvers and Observers are assigned to a Routing Slip which handles the workflow in Design object.
Attachments	Shows attached files

Additional Functions

Function	Purpose
Send Button	Allows you to send notifications
Route	Routes objects for approval
Revision Selector	Selects which revision of the current object to view, or to load into CAD
Load Button	When the Agile > Load command is active, this will load the current object (and objects in its structure) into CAD
Add File	Allows you to add files to the current object
Site	Allows filtering by Site
Approve/Reject	Approve/Reject the routed Label
Bookmark	Adds the current object into the Bookmark list
View	The User is able to see files in the Agile Viewer
Refresh	Refreshes the contents of the form

This page is blank