

Objecteering
Version 6.0 SP2: Readme

1.	INTRODUCTION	3
2.	INSTALLING OBJECTEERING 6	4
2.1.	REQUIRED CONFIGURATION	4
2.2.	REQUIREMENTS FOR USE OF THE DIFFERENT MDA COMPONENTS (MDACs).....	4
2.3.	INSTALLATION PROCEDURE	5
2.4.	UPDATE PROCEDURE	5
2.4.1.	<i>Migration of projects (model and diagrams).....</i>	<i>5</i>
2.4.2.	<i>Migration of modules developed using Objecteering/UML Profile Builder.....</i>	<i>5</i>
2.5.	OBJECTEERING DIRECTORIES AFTER INSTALLATION.....	7
2.6.	REPORTING BUGS TO OBJECTEERING SOFTWARE	7
3.	USING OBJECTEERING	9
3.1.	LICENSE MANAGER SERVICE (FLEXLM).....	9
3.2.	SERVER ACCESS RIGHTS	9
3.3.	AVAILABLE MDA COMPONENTS (MDACs)	10
4.	FURTHER INFORMATION.....	11
4.1.	USAGE PRECAUTIONS.....	11
4.2.	RESTRICTIONS AND KNOWN ISSUES.....	11
4.2.1.	<i>Installation.....</i>	<i>11</i>
4.2.2.	<i>Services on Windows.....</i>	<i>11</i>
4.2.3.	<i>Migration.....</i>	<i>11</i>
4.2.4.	<i>UML Explorer.....</i>	<i>11</i>
4.2.5.	<i>Search.....</i>	<i>11</i>
4.2.6.	<i>Models Diff/Merge.....</i>	<i>12</i>
4.2.7.	<i>MultiUser and other CMS.....</i>	<i>12</i>
4.2.8.	<i>Dimensions.....</i>	<i>12</i>
4.2.9.	<i>SCC (SCCModule).....</i>	<i>12</i>
4.2.10.	<i>Documentation.....</i>	<i>12</i>
4.2.11.	<i>C# Developer (CsDeveloper).....</i>	<i>12</i>
4.2.12.	<i>JavaDeveloper.....</i>	<i>13</i>
4.2.13.	<i>Java Developer and Diff/Merge.....</i>	<i>14</i>
4.2.14.	<i>Requirements (RequirementsManager).....</i>	<i>14</i>
4.2.15.	<i>Linux related issues.....</i>	<i>14</i>

1. Introduction

This document provides you with all the information necessary for your first steps with Objecteering 6. It does not, in any way, replace the documentation provided with Objecteering 6, but does provide important information which will help you get off to a good start. We strongly recommend that you read this document before using Objecteering 6.

2. Installing Objecteering 6

2.1. Required configuration

Objecteering 6 has been tested on the following platforms:

- Windows 2000 SP4 (x86)
- Windows XP SP2 (x86)
- Windows 2003 (server only)
- Linux Red Hat 9 (x86)
- Linux Red Hat Enterprise 4.2
- Solaris 2.8, 9 and 10 (SPARC)(server only)

The minimum configuration necessary when using Objecteering 6 with small-scale databases (~10Mb) is as follows:

- Pentium IV 1 Ghz
- 512 Mb memory
- 600 Mb hard drive (400 Mb in Stand Alone configuration)

The recommended configuration when using Objecteering 6 with large-scale UML models is as follows:

- Pentium IV 1 Ghz
- 2 Gb memory
- 700 Mb hard drive (500 Mb in Stand Alone configuration)

2.2. Requirements for use of the different MDA components (MDACs)

- Dimensions has been tested with the following tools:
 - Serena Dimensions 8.0.4.0
 - Serena Dimensions 9.1.0
- SCC has been tested with the following tools:
 - PVCS Version Manager 6.8.1
 - Microsoft Visual Source Safe 6.0
- ClearCase has been tested with the following tool:
 - Rational ClearCase 2003
- CMSynergy has been tested with the following tools:
 - Telelogic CMSynergy 6.3
 - Telelogic CMSynergy 6.4
- CSDeveloper requires the following application:
 - Microsoft .NET Framework 1.1
 - Microsoft Visual Studio .NET Professional 2003
- JavaDeveloper requires the following application:
 - J2SE Development Kit (It is recommended to use version 5 in order to take advantage of the latest improvements of the module).

2.3. Installation procedure

To install the Objecteering configuration that corresponds to your specific needs, please see either the "Objecteering 6 installation procedure" user guide or the installation guide in PDF format (root directory of the CD ROM) and follow the steps detailed for your desired configuration (stand alone or server/clients).

2.4. Update procedure

The installation procedure does not update earlier versions of Objecteering. When opened, Objecteering 5.3 databases are automatically migrated. Databases created using versions prior to version 5.3 of Objecteering cannot be directly migrated to Objecteering 6, and must first be migrated to Objecteering 5.3 and then to Objecteering 6. For more information contact the Objecteering Software Support team at the following address : support@objecteering.com.

2.4.1. Migration of projects (model and diagrams)

An ".ofp" file containing an Objecteering 5.3 project is automatically migrated when the file is opened in Objecteering 6.0.

Diagrams are automatically migrated but require that a "manual" check be carried out by the user. When a UML 1.4 diagram has to be migrated to UML 2.0, this happens automatically the first time the diagram is opened in Objecteering 6.0. A new UML 2.0 diagram is then created, and the original UML 1.4 diagram can still be consulted but cannot be modified. Given that the UML metamodel evolved significantly between versions 1.4 and 2.0, the migration of diagrams cannot be 100% guaranteed in UML 2.0, and it is up to the user to compare the original UML 1.4 diagram with the new UML 2.0 diagram. Once this comparison has been carried out and the conformity of the new migrated diagram has been accepted, the old diagram can be deleted.

2.4.2. Migration of modules developed using Objecteering/UML Profile Builder

Modules developed in Objecteering 5.3 with the UML Profile Builder should be migrated into "MDA components (the new name for modules in Objecteering 6 – these can be developed using MDA Modeler). The J code of MDA components should then be ported so as to be compatible with the UML 2.0 metamodel. This migration is carried out in two stages:

1st stage - migration of a module into an MDA component

This operation is carried out by Objecteering Software.

Objecteering has been supporting the MDA technology for several years and now provides a new tool, "Objecteering MDA Modeler", which replaces the "Objecteering/UML Profile Builder" tool featured in the version 5 product range. This major tooling evolution dedicated to the development of UML profiles is accompanied by a technological evolution adapted to UML 2.0/MDA, and results in ".mdac" files instead of ".prof" files.

The migration of ".prof" files into ".mdac" files is not automatic and must be carried out by Objecteering Software, who are able to migrate your modules into MDA components using dedicated tools.

2nd stage - adaptation of the MDA component to the UML 2.0 metamodel

This operation can be carried out by Objectteering Software or by you.

The UML 2.0 compatibility of modules migrated into MDA components is not 100% guaranteed, as some manual adaptations to the J implementation code are necessary according to the modules in question. As an example, an interface is a stereotyped class in UML 1.4 and becomes a specific metaclass in UML 2.0. If your module uses interfaces, this UML 2.0 metamodel evolution should be taken into account, in order to make the module migrated into an MDA component compatible with UML 2.0, so that it will be operational in Objectteering 6.0.

Objectteering Software module migration services

Objectteering Software provides two levels of service:

- The migration of your modules into MDA components, together with a detailed report specifying which lines of J code have to be modified.
- The complete migration of your modules into MDA components delivered in a directly UML 2.0 compatible state.

At the end of the first stage, the migration of a module into an MDA component leads to the delivery of the corresponding ".mdac" file, together with a detailed analysis report specifying which metamodel elements should be taken into account to make the module migrated into an MDA component UML 2.0-compatible, as well as which lines of J code should be modified.

Whatever your choice, please send your request to the following address:

module-migration@objectteering.com

2.5. Objecteering directories after installation

The Objecteering 6 installation defines a hierarchy whose root corresponds to the `OBJING_LOCAL_PATH` and `OBJING_SERV_PATH` environment variables.

The `OBJING_LOCAL_PATH` environment variable corresponds to the local resources access path. The `OBJING_SERV_PATH` environment variable corresponds to the server access path.

The `OBJING_LOCAL_PATH` directory contains the following sub-directories:

bin	Contains the tool's executables.
bundle	Contains the third-party tools used by Objecteering.
database	Contains the resources necessary for the creation of a new ofp project.
eclipse	Contains an archive for the plugin's Eclipse.
macros	Default directory for macro files.
help	Contains the Objecteering on-line help (files in html format).
mdaplugs	Contains the different files necessary to the correct functioning of MDA components.
res	Contains the resource files necessary to the correct functioning of Objecteering.
update	Contains the data used to migrate Objecteering databases created using a version of Objecteering prior to version 6.
tpf	Contains the files necessary to documentation generation.
work	Can contain user projects.

The `OBJING_SERV_PATH` directory contains the following sub-directories:

mdastore	Contains packaged MDA components.
site	Contains the Objecteering license file and the site counter.
setup	Contains the client installation procedure necessary to deploy Objecteering 6 clients.

2.6. Reporting bugs to Objecteering Software

When reporting bugs to Objecteering Software, don't forget to provide the following information:

- your platform
- your operating system and the version you are using
- your Objecteering configuration → (Client/Server or Stand Alone)
- the type of post on which the bug has been detected → (Server, Client, Stand Alone)
- the description of your bug

This information should be provided to the Objecteering support through the form accessible at <http://www.objecteering.com/support.php>.

N.B:

Objecteering produces a report file and a dump file in the case it crashes.

These 2 files must be sent to the Objecteering support to enable them to analyse the crash.

These files are generated in a \$(TEMP)/wobjing directory, where \$(TEMP) refers to the temporary directory as defined in the environment variables.

3. Using Objecteering

3.1. License Manager Service (FlexLM)

If the FlexLM service has not already been installed, please use the "lmttools.exe" application, located in the "bundle/flexlm" directory, and refer to the "Objecteering 6 installation procedure" document.

3.2. Server access rights

Objecteering users must have **read-write rights for the site file** ("site/site.dat" file) and **read rights for all Objecteering server directories**.

For further information, please refer to the "Objecteering 6 installation procedure" document.

3.3. Available MDA components (MDACs)

- ClearCaseModule (version 5.0.33) and ClearCaseAdministration
- CMSynergyModule (version 5.0.35) and CMSynergyAdministration
- CORBADesigner (version 3.0.09)
- CsDeveloper (version 2.0.14)
- CxxDeveloper (version 1.1.05)
- DimensionsModule (version 5.0.34) and DimensionAdmin
- Documentation (version 5.0.07)
- DOORS (version 3.0.14)
- FortranDeveloper (version 1.1.03)
- JavaDeveloper (version 4.1.27)
- Macros (version 2.0.00)
- MDAModeler (version 2.0.17) and MDARuntime
- Metrics (version 3.0.01)
- MultiUser (version 5.0.33) and AdministrationMultiUser
- ProcessManager (version 2.0.07)
- RAMComponent (version V1.0.10)
- RequirementsManager (version 3.0.14) and RequirementsAdmin
- SCCModule (version 5.0.33) and SCCAdministration
- SQLDesigner (version 2.0.10)

4. Further information

4.1. Usage precautions

We recommend that at regular intervals you archive projects that are being used and your site.dat file.

4.2. Restrictions and known issues

4.2.1. Installation

- As Objectteering 6 can be installed on a station already containing Objectteering 5.3.0 or older version, the 'ofp' files are associated to the last installed Objectteering. If Objectteering 6 is the last installed Objectteering then the 'ofp' files are associated to Objectteering 6.
N.B: if you open an 'ofp' file from 5.3.0 version with Objectteering 6, **it will be automatically migrated without any warning.**
- Directory names must not contain accented characters or @ character.

4.2.2. Services on Windows

- In a server installation, on Windows, the "Objectteering License Manager" service is not always run at the end of installation. This produces an error when "wobjing" is started. **In this case, the "Objectteering License Manager " service should be launched manually from the services item in the control panel.**

4.2.3. Migration

- Some MDACs (MDA components) parameters values are not preserved during projects migration.

4.2.4. UML Explorer

- The methods named "create" or "delete" are no longer constructors or destructors. For this, the "create" or "destroy" stereotypes should be added. The modify box has been improved and the "isConstructor" and "isDestructor" methods updated. The "AddStereotypeCreateOrDestroy.jmf" macro delivered with Objectteering automatically adds these stereotypes to the methods (if the methods have already been stereotyped, a warning message informs the user that the method has not been modified).
- If you wish to have "\$\$(GenRoot)" in generation products when they are created, define "\$\$(GenRoot)" at module parameter level. If you enter only "\$\$(GenRoot)" at module parameter level, the "\$\$(GenRoot)" variable will be expanded in the generation product.
- Due to a restriction in the graphic library, Objectteering can loop if more than 64 000 characters are pasted in a note at one time.
- It is not possible to copy, cut or directly move a dataflow from the browser.
- A Tagged value can be moved inside another tagged value

4.2.5. Search

- Stereotypes cannot be searched for by name.

4.2.6. Models Diff/Merge

- The number of elements in the Diff/Merge hierarchy cannot exceed 65000. For large-scale projects, **it is strongly recommended that you run Diff/Merge operations on the sub packages rather than on the root package of the project.**

4.2.7. MultiUser and other CMS

- In the case of a multiple selection, only the first selected element is dealt with by the CMS command.
- In order to limit inconsistency during the update of the repository, the MultiUser MDAC locks out simultaneous access to the repository by several users.
- Despite the fact that the * ? / \ : " < > and | characters are allowed when naming a project within Objecteering, it is not possible to use them with the MultiUser MDAC.

4.2.8. Dimensions

- Due to a limitation of the number of characters in the Serena Dimensions identifiers, login names cannot exceed 12 characters and project names are limited to 20 characters.

4.2.9. SCC (SCCModule)

- In the context of the Objecteering/SCC PVCSVM integration, the following message appears when the repository is created :

```
Objecteering/SCC 5.0.16 - Copyright 2000-2006 Objecteering Software
Added: conf Created Project., 1
Added: v:\PVCS\vm\common\ide_database\work\ProjectName\conf\version.conf, 1
Saving selected modules list in the repository ...
Error: / ProjectName/conf/modulesList.conf: The entity (or entities) for
"/ProjectName/conf/modulesList.conf" could not be loaded., 1
The entity (or entities) for "/ ProjectName/conf/modulesList.conf" could not be
loaded., 1
Added: v:\PVCS\vm\common\ide_database\work\
ProjectName\conf\modulesList.conf, 1
done
```

This PVCSVM error does not affect the creation of the repository and must be ignored.

- In the context of the Objecteering/SCC PVCSVM integration, during a check-out operation, if the project has the use of several "Promotion Groups", the relevant "Promotion Group" has to be selected for each imported element.

4.2.10. Documentation

- The complete generation paths should be defined to avoid the generation of files in the hierarchy of the Objecteering server (the directories '.' and '..' of a PC correspond to the directory from where the binaries are launched).
- If step 4 of documentation generation is not carried out properly, you must check and correct the settings of the "Editors" MDAC parameters.
- The integrated explorer in the HTML documentation generated by Objecteering only works recent browsers (IE 5.5 and upper, Mozilla, Firefox). It does not function with Opera.
- For certain values selected in Partial generation tab, the error "The page cannot be displayed" can appear.

4.2.11. C# Developer (CsDeveloper)

- CsAttribute notes are not taken into account for Packages, Enumerations and Events during generation.
- CsSummary, CsRemarks or Description notes are not taken into account for Packages, Enumerations and Events during generation.
- The reverse functions using the ISO 8859-1 character set. Please note that the UNICODE character sets are not yet supported.
- Source files containing special characters can cause exceptions during reverse operations.
- An exception is provoked during reverse operations when C# attributes concerning operation return parameters are reversed.
- C# attributes are not taken into account for Packages, Parameters, Enumerations and Events during reverse operations.

4.2.12. JavaDeveloper

- In read-only mode, accessors on Boolean-type attributes are now generated by default in the form "*is<AttributeName>*". To generate these accessors in the form of "*get<AttributeName>*", the "Generation of accessors in Java Bean format" tickbox in the "Generation options" parameter set of the Java Developer module should be unchecked. For further information, please consult the JavaDeveloper user guide.
- During a Java reverse, an exception can occur if one of the Java archive files (zip, jar files) specified cannot be found. The error displayed is as follows:

```

java.util.zip.ZipException: The specified access path cannot be found.
  at java.util.zip.ZipFile.open(Native Method)
  at java.util.zip.ZipFile.<init>(ZipFile.java:112)
  at java.util.jar.JarFile.<init>(JarFile.java:127)
  at java.util.jar.JarFile.<init>(JarFile.java:92)
  at com.softteam.javatoxml.xmlModel.TypeFinder.<init>(TypeFinder.java:75)
  at com.softteam.javatoxml.xmlModel.TypeFinder.getInstance(TypeFinder.java:111)
  at com.softteam.javatoxml.xmlModel.Generator.parseElement(Generator.java:126)
  at com.softteam.javatoxml.xmlModel.Generator.visitClassDef(Generator.java:109)
  at com.softteam.javatoxml.structuralModel.ClassDef.accept(ClassDef.java:68)
  at com.softteam.javatoxml.xmlModel.Generator.dumpXML(Generator.java:101)
  at com.softteam.javatoxml.xmlModel.Generator.visitPackageDef(Generator.java:88)
  at com.softteam.javatoxml.structuralModel.PackageDef.accept(PackageDef.java:43)
  at com.softteam.javatoxml.xmlModel.Generator.dumpXML(Generator.java:101)
  at com.softteam.javatoxml.xmlModel.Generator.visitPackageDef(Generator.java:88)
  at com.softteam.javatoxml.structuralModel.PackageDef.accept(PackageDef.java:43)
  at com.softteam.javatoxml.xmlModel.Generator.dumpXML(Generator.java:101)
  at com.softteam.javatoxml.xmlModel.Generator.generateXML(Generator.java:41)
  at
  com.softteam.javatoxml.ui.ReverseLauncher.reverseSourcesfiles(ReverseLauncher.java
:249)
  at
  com.softteam.javatoxml.ui.ReverseLauncher.launchReverseFromDirectory(ReverseLaun
cher.
                                                                    java:87)
  at
  com.softteam.javatoxml.ui.ReverseLauncher.launchReverse(ReverseLauncher.java:57)
  at
  com.softteam.javatoxml.ui.ReverseWizardModel.run(ReverseWizardModel.java:437)
  at java.lang.Thread.run(Thread.java:534)

```

If this exception appears, please check:

- That the "General/JDK Path" parameter references the root directory of the JDK.
- The "General/Accessible classes [CLASSPATH]" and "Reverse/Classpath for the Reverse" parameters.
- The Java archive files and existing directories.

4.2.13. Java Developer and Diff/Merge

- In round-trip mode, modifications to source files outside Objecteering are transferred into the model. Notes and tagged values are recreated. It is, therefore, normal to have differences on these elements (deletions and creations of notes and tagged values).

4.2.14. Requirements (RequirementsManager)

- To retain traceability links between the UML model and requirements during an XML import, the UML model should be imported first and then the requirements and dictionaries.
- To run a search, we recommend that you use the advanced search engine instead of the search engine.
- Requirement and dictionary editors
 - The sort by date function works correctly in the English format.
 - If the sort function in an editor appears to be inactive, you should simply close and re-open the editor.
 - Enumeration order is taken into account when sorting properties typed by enumerations. You should order your properties according to this strategy.
 - In the requirements or dictionary spreadsheet editors, after validating a multi-line field using the TAB or ENTER keys, you should press the TAB key again to be able to continue browsing using the keyboard. This is also true after canceling using the ESCAPE key.
 - To validate the edition of a multi-line field, the CTRL-TAB key combination must be used (or click outside the field).
- Tips
 - The shrink/expand function (activated through the "+", "-", "/" and "*" keys and the buttons on the bottom right of the editor) are used to display either the entire contents of term and requirement fields, or simply the first line.
 - To insert carriage returns and tabulations, use the SHIFT-ENTER keys for carriage returns and the SHIFT-TAB keys for tabulations.
 - To create (or delete) lines, you should simply use the INSERT (or DELETE) key. Use of the TAB key on the last line and last column automatically adds a new line.
- Since the RequirementsManager MDAC is integrated with Microsoft Word, the "Edit and Reverse documentation" command will only be operational if Microsoft Word is installed.

4.2.15. Linux related issues

- If you are using Linux with the KDE window manager, Objecteering may have incorrect colors display.
This is due to the fact that KDE override color settings in Objecteering.
To display expected colors follow these steps :
 - Open the KDE Control Center.
 - Select the Appearance & Themes/Colors option panel.
 - Uncheck the "Apply colors to non-KDE applications" toggle.
- It is not possible to use the "Copy image" (UML 1.4 diagrams) and "Copy the diagram as a graphic" (UML 2.0 Diagrams) commands under Linux.
To export a diagram as an image you can use the "Save as" (UML 1.4 diagrams) and "Save the diagram in a file" (UML 2.0 Diagrams) commands.
Then simply insert produced file in your documents.
- While using the "Save diagram in a file" command of the UML 2 diagrams or the "Save as..." command of the UML 1.4 diagrams do not forget to add the extension at the end of the file name otherwise the file will not be generated. Under Linux the "Save diagram in a file" command of the UML 2 diagrams does not support emf and wmf formats.

- You are experiencing one of the following problems with the font in Objectteering 6:
 - In the console, messages like the one below appear:
-- llvDisplay::createSystemFont: Bad font:
Edit the “~/.Xdefaults” file and delete the lines beginning with *Objectteering4** and then restart Objectteering 6.
 - In the console, messages like the one below appear:
Creating Font: Invalid Font
Please check that the Graphique X server font packages are correctly installed, notably the Xfree86-75dpi-fonts package.