

ORIGO SERVICES LIMITED

Origo XForms Generator Extension Plug-Ins

Controlling Agency : Origo Services

Date	Version	Distribution	Document Reference
14 th October 2005	1.0	Working Group	RPT/10142005

The copyright in this document is the property of Origo Services Limited.
It may not be copied without specific prior written consent from Origo Services Limited

© Origo Services Limited, 2005

DISCLAIMER

Origo Services Limited believes it has employed personnel using reasonable skill and care in the creation of this document. However, this document is provided to the reader 'as is' without any warranty (express or implied) as to accuracy or completeness and Origo Services Limited cannot be held liable for any errors or omissions in this document, nor for any losses, damages or expenses arising consequent to the use of this document by the reader.

CHANGE HISTORY

<i>Date</i>	<i>Version</i>	<i>Changes incorporated</i>
14 th October 2005	1.0	Initial Issue

CONTENTS

1. ORIGO XFORMS GENERATOR EXTENSION PLUGIN.....	4
1.1 SUMMARY	4
1.2 ENVIRONMENT SETUP	4
1.2.1 <i>Internet Explorer Novell XForms Plug-in</i>	4
1.2.2 <i>Java Runtime</i>	4
1.2.3 <i>Eclipse</i>	4
1.2.4 <i>IBM XForms Generator Plug-in</i>	5
1.2.5 <i>Origo XForms Generator Extension Plug-in</i>	5
1.3 USING THE PLUGINS TO GENERATE XFORMS.....	6
1.3.1 <i>Setting up the Eclipse Project</i>	6
1.3.2 <i>Set the IBM XForms Generation Plug-in Preferences</i>	7
1.3.3 <i>Edit the Stylesheets to reflect the location of the control file</i>	7
1.3.4 <i>Generating the XForm from an XML Instance</i>	7
1.4 APPLICATION LOGGING	8
1.4.1 <i>Logging Levels</i>	8
1.4.2 <i>Log File</i>	9
1.4.3 <i>Log Layout Pattern</i>	9

1. ORIGO XFORMS GENERATOR EXTENSION PLUGIN

1.1 SUMMARY

Origo have been using Eclipse 3.1 and IBM's XForms Generator Plug-In for Eclipse to generate a first draft copy of an XForm representing an Origo XML Schema.

For more information on IBM's XForms Generator see

<http://www-128.ibm.com/developerworks/xml/library/x-mdxfg1/>

The next logical step was to modify the first draft copy of the XForm produced by the IBM generator to:

- ❑ apply business rules (constraints) from the Origo Message Implementation Guidelines to the XForm.
- ❑ apply some grouping (structure) to the elements of the XForm to simulate the way the data is processed in real-life.

To achieve this Origo have developed a post-processor Eclipse plug-in which transforms the output from the IBM XForms Generator by applying a series of XSLT stylesheets one after the other.

The remainder of this document details the configuration requirements and how to use the plug-in.

1.2 ENVIRONMENT SETUP

1.2.1 Internet Explorer Novell XForms Plug-in

To display XForms in a browser, you need XForms support to be available.

Internet Explorer requires a plug-in to be installed before it supports XForms.

This piece of work has mainly used Novell's Internet Explorer XForms plug-in to date. Visit <http://lab.cph.novell.com/nxie/> to install the Novell plug-in, which allows Internet Explorer to process XForms.

It is our intention to make XForms available that have been tested against a wider range of XForms implementations when time permits.

1.2.2 Java Runtime

The IBM XForms Generator plug-in requires Sun's JRE 1.4.x (preferably 1.4.2). This is in line with all IBM's Eclipse based products.

The Origo Eclipse Plug-in uses Xalan-J 2.4.1 (which is the underlying XSLT processor in Sun's JRE 1.4.2 used by Origo's Eclipse 3.1 environment) to apply the transformations.

1.2.3 Eclipse

To use the IBM and the Origo Eclipse plug-ins you must have Eclipse 3.1 installed.

Eclipse 3.1 and the pre-requisites must be installed in the order listed below.

All Eclipse archives (zips) can be located at:

<http://www.eclipse.org/downloads/index.php>

Archive	Description
eclipse-SDK-3.1-win32.zip	Eclipse 3.1
emf-sdo-xsd-SDK-2.1.0.zip	Eclipse Modelling Framework (EMF)
GEF-SDK-3.1.zip	Graphical Editing Framework (GEF)
JEM-SDK-1.1.zip	Java EMF Model (JEM)
wtp-0.7RC3.zip	Web Tools Platform (WTP)

or on the Origo Standards website where they have also been placed:

<http://www.origostandards.com/XForms/InstanceValidator/pre-reqs>

1.2.4 IBM XForms Generator Plug-in

The IBM XForms Generator Plug-in is available at:

<http://www.alphaworks.ibm.com/>

and can be installed as follows:-

1. Start the Eclipse IDE (eclipse.exe).
2. Click “Help”/”Software Updates”/”Find and Install”.
3. Click “Search for new features to install”.
4. From the Install dialog click the “New archived site” button and navigate to IBM XForms Generator Plug-in archive you downloaded.
5. On the “Edit Local Site” dialog click “OK”.
6. Back on the “Install” dialog ensure that only the IBM XForms Generator Plug-in archive is checked and select “Finish”.
7. Accept the dialog stating that the jars are unsigned.

Eclipse will restart to load the new plug-in and make it available for immediate use. To ensure that installation has been successful check the Window/Preferences menu. You should now see an entry called “XML Forms Generator”.

1.2.5 Origo XForms Generator Extension Plug-in

The Origo XForms Generator Extension plug-in implements Apache Log4j Application Logging to allow granular logging in case of any problems.

The Logging Plug-in must be installed first, followed by the Origo XForms Generator Extension plug-in.

Download the Logging plug-in (*com.tools.logging_1.0.0.zip*) from:

http://www.origostandards.com/XForms/InstanceValidator/OrigoPlugin/com.tools.logging_1.0.0.zip

Next download the Origo XForms Generator Extension plug-in

(*OrigoXfgExtension0.1.20050907.100000.zip*) from:

<http://www.origostandards.com/XForms/InstanceValidator/OrigoPlugin/OrigoXfgExtension0.1.20050907.100000.zip>

Install as follows:-

- (a) to install the Log4j Logging for Eclipse plug-in

(*com.tools.logging_1.0.0.zip*) inside Eclipse 3.1 choose:

- Help.
- Software Updates.
- Find and Install.
- Search for new features to install.
- click 'New Archived site' and navigate to the archive *com.tools.logging_1.0.0.zip*
- click Finish and accept the license and unsigned jars dialogs.

Eclipse will restart to load the new plug-in and make it available for immediate use.

(b) to install the Origo XForms Generator Extension plug-in (*OrigoXfgExtension0.1.20050907.100000.zip*) inside Eclipse 3.1 choose

- Help.
- Software Updates.
- Find and Install.
- Search for new features to install.
- click 'New Archived site' and navigate to the archive *OrigoXfgExtension0.1.20050907.100000.zip*
- click Finish and accept the license and unsigned jars dialogs.

Eclipse will restart to load the new plug-in and make it available for immediate use.

1.3 USING THE PLUGINS TO GENERATE XFORMS

1.3.1 Setting up the Eclipse Project

First of all you will need an XML instance with associated schemas and stylesheets to work with.

Download the sample Eclipse Project from

<http://www.origostandards.com/XForms/InstanceValidator/OrigoBondForms/OrigoBondForms.zip>

This Eclipse Project contains:

- stylesheets (constraints.xml and structure.xml).
- stylesheetlist.xml which is a list of stylesheets to be applied – currently only two (constraints.xml and structure.xml).
- instances – one example NB Investment Bond XML message.
- schemas – associated schemas (including a Schematron schema used to specify business rules).
- forms – folder to contain the XForms created – currently only contains the css stylesheet used by the generated forms and one example form.

Create an empty Eclipse workspace and import the project from the *OrigoBondForms.zip* archive by following these instructions.

1. Open a new Eclipse workspace (eclipse.exe).
2. Select File/Import.

3. Choose “Existing projects into workspace” option.
4. Navigate to *OrigoBondForms.zip* which you previously downloaded.
5. Open this archive (zip) and click Finish.
6. This will setup a project in your workspace called *OrigoBondForms* containing example XML instances and schemas to be used to create XForms.

1.3.2 Set the IBM XForms Generation Plug-in Preferences

Set the XForms Generation plug-in preferences as follows:

1. In Eclipse select Window/Preferences.
2. Open the XML Form Generator preference.
3. Select Output and check the “create element demarcations”.
4. Open the Output preference and select the Stylesheets option and change the cascading stylesheet to *novell.css*.
5. Open the ‘Post processing’ option and navigate to the location on your hard disk where *stylesheetlist.xml* exists (within the current workspace) to specify the list of style-sheets to be applied during post processing.

1.3.3 Edit the Stylesheets to reflect the location of the control file

Modify the *structure.xsl* and *constraints.xsl* to include the correct location for the control file which manages the way the transformations are handled.

The current setting is:

```
<xsl:variable name="control"
select="document('D:\eclipse\workspaces\OrigoBondForms\OrigoBondForms\stylesheets
\control.xml')/control"/>
```

You should modify the highlighted section to reflect the way your environment is set up.

1.3.4 Generating the XForm from an XML Instance

Generate an XForm using the IBM and the Origo Extension plug-in by following these instructions (which use an example Origo New Business XML message).

1. In the OrigoBondForms Eclipse project expand the instance folder and select the QNBDBDistributionBondNewBusiness.xml.
2. In the Navigator pane right Click on the selected instance and choose “Generate XHTML/XForm”.
3. Provide a name for the output form being created e.g. newForm.xhtml.
4. Select Finish.
5. After approx 20 seconds the new XForm will be created and presented in an open editor.
6. In the Navigator pane right click on the new XForm and select ‘Render with Internet Explorer’.
7. You should see the well structured XForm with the stylesheets applied (constraints and structure).

If you experience any problems at this stage you should check the application log to determine what the problem is – see Application Logging.

1.4 APPLICATION LOGGING

The Origo XForms Generator XSLT Extension plug-in uses Apaches Log4j for application logging. If you experience any problems with the Origo XFG Extension plug-in then the application log file is the first place to look for diagnostic information.

By default the plug-in is currently deployed with logging switched on at the “information” level (INFO).

Log4j allows the setting of the logging level to be changed without impacting any application code.

The logger.properties file is located at the root of the plug-in contains the logging configuration information and can be changed as required.

For example the logging.properties file can be found in

C:\eclipse\plugins\com.origo.xfg.extensions.xslt_0.1.20050907.100000.jar

It looks like this:

logging.properties
<pre># For the general syntax of property based configuration files see the # documentation of org.apache.log4j.PropertyConfigurator. log4j.rootCategory=INFO, A1 # A1 is set to be a PluginFileAppender log4j.appender.A1=com.tools.logging.PluginFileAppender log4j.appender.A1.File=xsltplugin.log log4j.appender.A1.layout=org.apache.log4j.PatternLayout log4j.appender.A1.layout.ConversionPattern=%d{ ddMMMyyyy HH:mm:ss} %p %c - %m%n</pre>

1.4.1 Logging Levels

Logging is configured above to produce INFO level traces.

Logging levels can be changed to produce more or less information.

The logging level hierarchy is DEBUG,INFO,WARNING,ERROR.

INFO produces less than DEBUG

WARNING produces less than INFO

ERROR produces less than WARNING

The Java application will contain logging statements that are only executed if the logging level is appropriate. For full debug logging set the value to DEBUG.

1.4.2 Log File

The log file is named in the `logger.properties` file. It is currently set as `xsltplugin.log` and will be produced relative to the Eclipse workspace. You may specify an absolute path in the `logger.properties` if required.

For example with the `logger.properties` file detailed above the log file would be located at:

```
C:\eclipse\<workspace-location>\<workspace-name>
\metadata\plugins\com.origo.xfg.extensions.xslt
```

1.4.3 Log Layout Pattern

The layout pattern produces log messages in the format of this example.

```
02Sep2005 15:29:04 INFO com.origo.xfg.extensions.xslt.OrigoXSLTPostprocessor - Starting postprocess method
02Sep2005 15:29:04 INFO com.origo.xfg.extensions.xslt.OrigoXSLTPostprocessor - Preference Stylesheet
List=C:\eclipse\workspaces\OrigoBondForms\StylesheetComponents\stylesheetlist\stylesheetlist.xml
02Sep2005 15:29:04 INFO com.origo.xfg.extensions.xslt.OrigoXSLTPostprocessor - Transformation has started
02Sep2005 15:29:06 INFO com.origo.xfg.extensions.xslt.OrigoXSLTPostprocessor - *****Transformation has ended
02Sep2005 15:29:06 INFO com.origo.xfg.extensions.xslt.OrigoXSLTPostprocessor - Transformation has started
02Sep2005 15:29:07 INFO com.origo.xfg.extensions.xslt.OrigoXSLTPostprocessor - *****Transformation has ended
```