Acceleo 3.0 - Helios Release Review

Release Review and Graduation Review

Helios Simultaneous Release

Jonathan Musset (jonathan.musset@obeo.fr)

Release Review : June 11, 2010

Communication Channel : eclipse.modeling.m2t newsgroup

Process Documentation : http://www.eclipse.org/projects/dev_process/development_process.php

*Project Plan : http://www.eclipse.org/projects/project-plan.php?
planurl=http://www.eclipse.org/modeling/m2t/acceleo/plan-helios.xml*

This Release Review document is also available at
http://wiki.eclipse.org/Acceleo/ReleaseReview#Helios_Release_Review_-_Acceleo_3.0

Helios Release Highlights

Graduation

Acceleo graduates with its Helios release. This usually means the plugin grows from version 0.x to version 1.0. However, Acceleo already has a version history and an existing adopter community. In order not to unsettle this community and to stay coherent with our history, Acceleo will jump under eclipse from version 0.8 to version 3.0 (as the last released version on Acceleo.org was 2.x).

New and noteworthy

Tooling improvements

- New views have been introduced to help you on several aspects of the generation cycles
  - The Result view to get feedback on the traceability information
  - The Overrides view to easily create templates overriding others

- Acceleo now provides a built-in profiler, an advanced debugger and a stack trace mechanism for runtime exceptions

- Acceleo 3.0 is compatible with Eclipse 3.4, Eclipse 3.5 and Eclipse 3.6 (Helios)

Editor Improvements

- Performance improvements

- Basic refactoring actions

- Portions of text can be extracted as new templates

- The editor now highlights all occurrences of the selected element

- Quickfixes are available to fix compilation issues by creating new module elements

- New menus and quick fixes to wrap Java extension points (Java services)

- Bottom-up approach when developing generators
  - Initialize the content of the generator with an example
  - Specific actions to transform an example in dynamic OCL expressions

Improvements on the language

- Full MTL/OCL standard library implementation

- Build mode, either Strict or Lax MTL compliance

- Migration facilities for those who own Acceleo 2.x templates

Quality of APIs

The component lead certifies that the requirements for Eclipse Quality APIs have been met for this release. All non-API code is in "internal" packages.

End of Life Issues

None

Committer Diversity

4 Committers from Obeo, and 2 individual committers.

IP Issues

The about files and use licenses are in place as per the Guidelines to Legal Documentation.

CQ 2518 - The Content includes items that have been sourced from the OMG MTL 1.0 specification : http://www.omg.org/spec/MOFM2T/1.0/

All other contributions (code, documentation, images, etc) have been committed by individuals who are either Members of the Foundation, or have signed the appropriate Committer Agreement. In either case, these are individuals who have signed, and are abiding by, the Eclipse IP Policy. The other contributions of the IP log are not significant or are written 100% by employees of the same employer (Obeo) as the Submitting Committer (EclipseLegalProcessPoster.pdf).

All contribution Questionnaires have been completed.

The "provider" field of each plugin is set to "Eclipse Modeling Project".

The "copyright" field of each plugin is set to the copyright owner.

Any third-party logos or trademarks included in the distribution (icons, logos, etc) have been licensed under the EPL.

The Acceleo IP log is located at http://www.eclipse.org/projects/ip_log.php?projectid=modeling.m2t.acceleo

Non-Code Aspects

Unit tests

706 unit tests launched on every build. 1219 unit tests launched every day. Code coverage is about :

> 71 % of the code generation engine
> 84 % of the compiler
> 29 % of the UI

Code quality

Checkstyle activated on each distinct plug-in.

Eclemma used on a regular basis to improve test coverage.

Yourkit used on a regular basis to improve performances.

Javadoc represents more than 46% of the Java source code.

An Ohloh page is available at http://www.ohloh.net/p/acceleo-eclipse/analyses/latest

APIs

Metamodel definitions and interfaces are considered APIs.

Commitment to provide stand-alone code generation.

Non-API classes are separated from the exposed API through an "internal" namespace.

API Tooling is activated on all plugins to mark exceptions within the exposed API.

Documentation

The documentation that comes with Acceleo 3.0 is available in the Help menu : Acceleo Model To Text Transformation Language.

Each document of the documentation set deals with a different aspect of mastering Acceleo : There are a Plug-in Overview, a Quick Start, an Android Code Generation Tutorial, a User Guide, several short tutotials to create code generation projects, an Acceleo/OCL operations reference to describe the Acceleo/OCL library, and a Text Production Rules document.

The main documentation is the User Guide because it fully describes each and every feature of Acceleo. From the installation to the execution with a trip through the editing features, it gives an in-depth description of everything there is to know about Acceleo.

Basic Examples are available in the example category : UML to Java, and Ecore to Python.

There is also a basic information page on the Eclipse Wiki http://wiki.eclipse.org/Acceleo

Bugzilla

Status

Severity	NEW	RESOLVED	VERIFIED	CLOSED	Total
blocker	2	.	1	1	4
critical	.	3	.	2	5
major	8	3	.	9	20
normal	22	45	1	38	106
minor	6	4	.	1	11
enhancement	6	3	.	2	11
Total	44	58	2	53	157

Note : these figure is subject to change as the whole team is in the process of fixing bugs until the final release (this snapshot has been taken on May 26, 2010)

Tool usability

Localization : integrated into Babel

Standards

This implementation of Acceleo is based on the OMG MOF Model-To-Text OMG Standard : http://www.omg.org/spec/MOFM2T/1.0/

End of Life

There are no specific end of life concerns for this release.

Communities

Talks have been given on the following events:
- Acceleo at Eclipse Con 2010 : Acceleo Code Generation - Let's start with an Android example and Acceleo at Eclipse Modeling Runaway 2010
- Acceleo at Eclipse Summit Europe 2009 : From Acceleo.org to Eclipse Modeling
- Acceleo at Eclipse Con 2009 : a Real Standard Alternative for Code Generation

Activity on the M2T newsgroup (eclipse.modeling.m2t) :
308 messages on Acceleo from June 2009 to May 2010

Schedule

Since the feature freeze our commitment was to follow the EMF and the MDT OCL builds as closely as possible.

Acceleo 3.0 Release Plan

M1   *08/18/2009*
M2   *09/29/2009*
M3   *11/10/2009*
M4   *12/15/2009*
M5   *02/02/2010*
M6   *03/16/2010*
M7   *05/04/2010*
RC1 *05/18/2010*
RC2 *05/25/2010*
RC3 *06/01/2010*
RC4 *06/08/2010*
Final *06/16/2010*
3.0   *06/23/2010*

Project Plan

The Helios project plan is available at http://www.eclipse.org/projects/project-plan.php?planurl=http://www.eclipse.org/modeling/m2t/acceleo/plan-helios.xml

What's next?

- E4 Compatibility
- Macro support
- Full compatibility with ATL
- Improve the bottom-up approach when developing generators
- Advanced quick fixes
- Improve the JUnit tests coverage on the UI
- "On file close" support
- Compatibility with the next OMG specification version

Legal Notices

Java and all Java-based trademarks are trademarks of Oracle, Inc. in the United States, other countries, or both.

UML, OMG, EMOF, OCL and XMI are trademarks of the Object Management Group.

Other company, product, or service names may be trademarks or service marks of others.