

eTrice 0.1.0 Release Review



Planned Review Date: 30.11.2011

Communication Channel: eclipse.etrice

Thomas Jung, Henrik Rentz-Reichert, Thomas Schütz

Introduction

eTrice provides an implementation of the ROOM modeling language for event driven real-time systems together with editors, code generators for Java code (in the future also C++ and C) and exemplary target middleware.

The model is defined in textual form (Xtext) with graphical editors (Graphiti) for the structural and behavioral (i.e. state machine) parts.

Features: Textual Model Editor

- Models are stored in a purely textual and easily readable form.
- The textual model holds all the semantic information.
- The DSL is created using Xtext and a fully featured editor with syntax highlighting, content assist, template proposals and outline view.
- Import of model parts allows an arbitrary granularity as well as working with modeling libraries.
- Merge conflicts resolvable by user.

Features: Graphical Structure Editor

- Graphiti based editor for the Actor Structure.
- Simple automatic layout for initial creation of graphical representation of semantic model elements.
- Supports creation and deletion of elements such as Ports, Actor References and Bindings.
- Automatic update after changes in the textual (semantic) model.
- Simple and fast navigation between diagrams.

Features: Graphical Behavior Editor

- Graphiti based editor for the Actor Behavior in form of hierarchical finite state machines.
- Simple automatic layout for initial creation of graphical representation of semantic model elements.
- Supports creation and deletion of elements such as States, Choice/Initial/Entry/Exit/Transition Points and Transitions.
- Automatic update after changes in the textual (semantic) model.
- Simple and fast navigation in the state machine hierarchy.

Non-Code Aspects

- Online documentation (Eclipse Help) is provided and available also in PDF form.
- Several Tutorials are available from basic to more advanced and comprehensive.
- Tutorial code is shipped with the eTrice bundles and is easily installable to the user's workspace.
- Article about eTrice has been published in German Eclipse Magazin 2/11

APIs

- eTrice is a modeling tool, not a framework. Therefore it is not very API centric.
- A typical way to extend eTrice will be to create new generators for new languages together with a corresponding middleware. Therefore we extracted major functionality of the generator including model traversing algorithm into the target language independent generator model and helper classes.

Architectural Issues

- eTrice has been built on top of the popular EMF and Xtext and the new Graphiti framework.
- Emphasis has been laid on a simple and sound architecture. Continuous refactorings have been done to keep the architecture clear and concise.
- eTrice is still in its very beginning. The ways it can be adopted and extended will evolve together with a growing community of users.

Tool Usability

- From the very beginning simplicity of use has been a major goal of the eTrice project.
- ROOM as a modeling language allows the user to solve typical problems of the development of highly concurrent real-time systems on a higher level of abstraction.
- All editors, textual and graphical are simple to use and designed to support a fast development.

End-of-Life

- There has been no previous release.

Bugzilla

- All bugs scheduled for the 0.1.0 version of eTrice

Standards

- The modeling language ROOM is not standardized but was published in [1].
- [1] defines a graphical notation which was adopted.
- [1] also contains a textual notation. But this is incomplete and we decided to develop our own notation based on this.

[1] Bran Selic, Garth Gullekson, Paul T. Ward:
Real-Time Object Oriented Modeling, New York:
John Wiley, 1994 (ISBN 0-471-59917-4)

UI Usability

- The Eclipse UI Guidelines have been adhered to.
- No language packs have been provided since the user community is used to tools in English language only.
- eTrice delivers the very good UI usability of the underlying Graphiti and Xtext frameworks

Schedule

- eTrice started from scratch in May 2010 and did its initial contribution roughly a year ago.
- Though no strict time frame was set it can be said that the project is delivering its first release within an acceptable time.

Communities

- Since its start in October 2010 the project has won one additional committer and two contributors.
- Talks have been given on a number of conferences and opportunities like Eclipse Summit Europe 2010 and EclipseCon 2011, Eclipse Demo Camps, Eclipse Embedded Days.
- eTrice is already used as part of the tool chain in industry projects.
- At the time eTrice users and developers mainly use direct communication. Therefore the newsgroup and the development mailing list are only rarely used.

IP Log

- The Eclipse IP policies and procedures have been followed
- The eTrice IP Log can be found at http://www.eclipse.org/projects/ip_log.php?projectid=modeling.mdt.etrice
- A frozen version can be found at <http://eclipse.org/etrice/pages/etrice-ip-log-0.1.0.htm>

IP Issues

- The EMO explicitly asks during the Release Review if any Member would like to assert that this release infringes their IP rights.
- If so, the EMO and the project will follow the Eclipse IP Policy in discussions with that Member.

Credits and Kudos

- Kudos to all who contributed to the project, be it in form of feedback, suggestions, questions or most valuable code contributions