

The Fujaba Newsletter*

Editorial

Hi all. This is the new 'Fujaba Newsletter', which we (the Fujaba project committee) originated to inform all developers and users about the newest trends and developments in this project. In fact, there have been many meetings and discussions in the run-up to this newsletter. The most important results are presented in the article about the re-organization of the project. A very important topic for all you developers out there should be the article about the new Fujaba architecture. Moreover, we decided to put some more effort in the documentation of the project which will help you to get deeper into the rabbit hole Fujaba. So, enjoy this newsletter – and stay tuned to the project! (duke)

(Re)organizing the Fujaba Project

The Fujaba project has evolved towards a multi tool project and therefore a major redesign effort is required. On the one hand, multiple rather independent tools that reuse the Fujaba core functionality such as Reddmon are under development or planned, while on the other hand extensions such as postscript support, changes such as the Fujaba swing adapter (fsa) and quality improvements for the basic Fujaba infrastructure (LIFE project) that effect the core functionality are on their way. Therefore, the staff of the software engineering groups has established a special task force to identify suitable adjustments to address this conflict in a reasonable manner. The major outcome is besides the need for regular project management activities an architectural redesign effort towards a modular architecture with a core Fujaba framework and plugin support.

*Fujaba (Form UML to Java and Back Again.) is a project from the Software Engineering Group, University of Paderborn, Germany. See www.fujaba.de for further details.

We will present these first results in the forthcoming project meeting on December the 19th. See you! (hg)

Fujaba Developer Documentation

One result of the analysis of the Fujaba project during the meetings of the task force has been, that it is currently rather hard for new students and staff members to find out what is relevant for a specific task when developing within the Fujaba project. Therefore, we started an initiative to collect and develop relevant documents in form of a common documentation repository for Fujaba developers. Keeping the documentation up to date is a rather difficult task and therefore we encourage all developers to adjust identified errors or shortcomings on the fly in the plain HTML base document WWW tree. For each document a responsible staff member has been assigned whom you might also contact to discuss your concerns. (hg)

The new Fujaba architecture

Fujaba has grown in the last four years to over 700.000 lines of code. It's an monolithic system with numerous functions. The effort of extending and maintaining Fujaba is too high. So the Fujaba task force decided to develop a new plug-in architecture. The main idea is to split Fujaba into little pieces that can be maintained more easier. These pieces are handled in separated repository modules with restricted access. Functionality will be extracted into plug-ins. A Fujaba core is created to load these plug-ins at runtime.

A second mechanism - called Stubs - is provided to extend the Fujaba meta-model. The "classic" Fujaba meta-model with its "UML"-meta classes will be transformed into "ASG"-meta classes (Abstract Syntax Graph). The new meta-model

will be part of the core and can't be changed by public any more. An abstract class ASGStub provides an association to ASGElement, the root class of the new core meta-model. A new classes that should be added to the Fujaba meta-model by plug-ins uses subclasses of ASGStub to associate with classes of the core meta-model.

The objective is to develop a Fujaba platform that is more easier maintainable and that can be composed by core and different plug-ins for your own purpose. For further details see the Fujaba documentation. (lowende)

The Fujaba Outpost in Braunschweig

For about a year now, the Fujaba-Team has an outpost in Braunschweig. I created this outpost due to my new position as a temporary professor for software engineering at the technical university of Braunschweig. In Braunschweig, I give lectures on software engineering and of course I am supervising several bachelor, master and PhD students. Most of this work is done in the context of Fujaba. As a student programmer, Thomas Maier created a basic usecase diagram editor (that lacks deletion commands, only). Ralf Schuhmacher created within his bachelor thesis a sequence diagram editor (and especially a reverse engineering functionality as provided by TogetherJ). Unfortunately the sequence diagram editor faces still some problems with Fujaba's new FSA GUI structures, but Carsten Reckord promised to solve these issues soon. To facilitate the step from analysis to design, Thomas Maier's master thesis has the task to derive statecharts from sequence diagrams. I have already seen some promising demos. Based on this work, another master thesis by Hendrick Kay will

provide reverse functionality, i.e., Hendrick Kay will derive sample scenarios from statecharts. Such generated sample scenarios provide a reasonable test coverage for a set of given statecharts. Alternatively or in addition, Hendrick Kay's work will propagate statechart changes back into the original sequence diagram scenarios. In the area of design level debugging, Leif Geiger's bachelor theses integrates Mr. Dobs and the Java debugger frontend JSwat. Thus, Mr. Dobs now allows to step through invoked methods. Currently, this stepping is done at code level. However, the thesis of Bjoern Waide has already installed basic features to allow stepping at the story diagram level. This shall be finished in spring 2002. Last but not least, the Braunschweig outpost tries to spread Fujaba, too. Within her PhD thesis, Ira Diethelm uses Fujaba like the Life³ project to teach object oriented concepts at a highschool in Braunschweig. Especially, Ira's thesis will provide a sophisticated course book to teach story diagrams to freshmen. (zuendorf)

May I introduce...

...the top management of the project? It's Professor Wilhelm Schäfer who built our department in Paderborn in 1994.



As head of the software engineering group, he is significantly involved in the development of new concepts. Many of them are then implemented in Fujaba. Like to get more information? Visit his

homepage – it is available in English and German at <http://www.upb.de/cs/wilhelm>. (duke)