

Java EE Patterns Proposal

Introduction

The Java EE Patterns Project is a proposed open source project under the [Eclipse Web Tools Platform Project](#).

This proposal is in the Project Proposal Phase (as defined in the [Eclipse Development Process document](#)) and is written to declare its intent and scope. This proposal is written to solicit additional participation and input from the Eclipse community. You are invited to comment on and/or join the project. Please send all feedback to the <http://www.eclipse.org/newsportal/thread.php?group=eclipse.webtools.patterns> newsgroup.

Background

The long experience of Java EE application developments established few best practices known as Java EE Design Patterns that determine how common implementation problems should be solved with the means of Java EE. The Eclipse Web Tools Platform provides a rich set of instruments that can be used to develop Java EE application by following the established design patterns. In the usual case this means executing at least several operations (actions, wizards) and some hand work. Automating this routine job would give lots of benefits to developers like:

- Shorter development time
- Using of semantically error proof generated code
- Improved learning curve

Scope

The objectives of the Java EE Patterns project are to:

- Introduce common Java EE Design Patterns to Java EE application developers.
- Introduce other common Java EE development tasks as patterns.
- Act like an incubator for new operations needed for implementing patterns. These operations should later graduate to become part of the corresponding Eclipse project.
- Enable all developed pattern for headless execution.

Description

All common tasks and design patterns that are in the Scope of the project will be implemented as patterns of the Pave framework. In other words, this project is a client of the Pave framework and exemplifies the usage of the last one.

This project will act as an incubator for new Java EE patterns and related operations. Each of the incubating patterns will graduate independently to the corresponding subproject of Eclipse WTP. It is very likely that any operations developed in relation to the pattern will move to the corresponding Eclipse WTP subprojects that could be different from the one where the pattern itself is moved.

The initial contribution consists of the following two patterns.

Session CRUD Façade

The Session CRUD Façade is an exposed EJB session bean with methods for Create, Read, Update and Delete functions of the corresponding JPA entities.

This pattern creates a new EJB session bean and generates the needed CRUD methods and queries depending on the given entities as input. The pattern reuses the operation and UI of the Session Bean wizard that is already part of the Eclipse WTP project and defines additional operations.

JSF Application

This pattern generates a skeleton of a complete Java EE application – up to the JSF front end – based on the given JPA entities as input. This enables Java EE developers to quickly test their JPA domain model and database structure, or even jump start with a working sketch of the entire Java EE stack.

The pattern is a composition of the Session CRUD Façade pattern and additional operations defined for:

- Generating JSF manage beans.
- Generating JSP pages.
- Including navigation rules in the faces-config.xml descriptor.

Organization

Initial committers

- Dimitar Giormov (SAP): project lead
- Milen Manov (SAP): committer

Mentors

Looking for mentors!

Interested parties

SAP AG

Developer community

We expect to extend the initial set of committers by actively supporting a developer community that will develop new Java EE patterns.

User community

Java EE patterns are of great benefit for beginners. Therefore, we expect that large part of the Java EE developers' community will use the provided patterns.

Tentative Plan

- *Apr 2009*: Submit project proposal.
- *May 2009*: Construct web site with documentation and tutorials.

- *Jun 2009*: Creation review.
- *Jun 2009*: Prepare project infrastructure.
- *Jul 2009*: Initial contribution and IP review.
- *Jul 2009 – Feb 2010*: Develop new patterns and improve existing ones. Respond to community feedback.
- *Feb 2010*: Graduate mature patterns to the corresponding Eclipse WTP subprojects.