Discipline by Discipline: Requirements

As many who follow my blog entries and have read my articles know, I use the Unified Process as framework to manage projects and programs. While the phases of the UP (Inception, Elaboration, Construction and Transition) are powerful ways to manage the risk and narrow the ‘cone of uncertainty‘ of a project, I find the disciplines within the Unified Process as useful containers for ensuring roles are established and that artifacts are being developed that will support the project.

However, beyond the phases and disciplines I find most of the artifacts and activities as too abstract for effective application in most real world projects. Instead, I mix in a series of techniques that I have applied successfully and found round out the details of each of the disciplines with RUP. This first article focuses on the top of the “V” model, Requirements and Analysis.

Read More

A Unified Approach to Agility (Article)

With the increasing interest in Agile techniques such as Scrum and XP, I often come across clients and project managers assuming that these approaches alone are sufficient to ensure the success of their projects. In actuality, the Agile Principles are really a value system that help contribute to

effective behaviors on a project. None of the agile techniques recommend dispensing with the well defined practices that govern effective project implementations such as risk, scope and change management (amongst others). In fact most of the Agile techniques found in current literature are intended to work within existing frameworks and metamodels, without which your projects won't succeed.

When I develop project plans and teach project management approaches, I frequently turn to the Metamodel offered by the Unified Process. What I like specifically about the Unified Process is that it breaks a project into four phases (**Inception, Elaboration, Construction and Transition**) that have clear entry and exit criteria that are easy to manage against. In addition the phases are well defined and relatively intuitive to most people (**Inception** involves scoping and structuring the project, **Elaboration** focuses on de-risking the project and developing an Architecture, **Construction** emphasizes the rapid development phsae of the project and **Transition** focuses on readying the application for deployment).

The UP also contains a number of useful *'disciplines'* which reflect major workstreams in a project lifecycle. **Business Modeling, Requirements, Analysis and Design, Implementation, Test and Deployment** ebb and flow across the project lifecycle while **Project Management, Configuration/Change Management and the Environment** disciplines are focused on supporting the lifecycle in its entirety (these latter are found in the IBM version of the Unified Process called the Rational Unified Process.

The popularity of the Unified Process is reflected in its evolution into a number of forms including the Agile Unified Process, Enterprise Unified Process and even the Oracle Unified Process. IBM recently released an open source version called OpenUP which is based on its popular Eclipse Process Framework.

Over time I've come across a number of agile/lean techniques that support the disciplines I mentioned above and enhance these disciplines to make them more effective. Over the next few postings I'll offer a walk through on a discipline by discipline basis on each of these techniques.

Discipline by Discipline

- Requirements and Analysis
- Design
- Development
- Testing
- Project Management

A Unified Approach to Agility

With the increasing interest in Agile techniques such as Scrum and XP, I often come across clients and project managers assuming that these approaches alone are sufficient to ensure the success of their projects. In actuality, the Agile Principles are really a value system that help contribute to effective behaviors on a project. None of the agile techniques recommend dispensing with the well defined practices that govern effective project implementations such as risk, scope and change management (amongst others). In fact most of the Agile techniques found in current literature are intended to work within existing frameworks and metamodels, without which your projects won't succeed.

Read the article