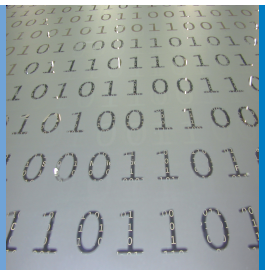

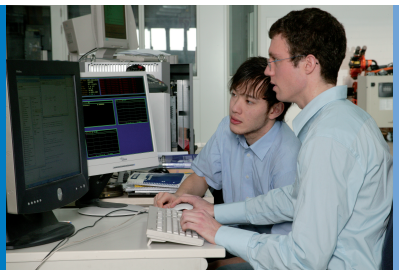

Agile Software Development

Get More Flexible Executable Software Quicker with Modern Software Engineering

Agile methods have proved to be an important means for enhancing software quality in recent years. Agile processes allow you always to respond flexibly, appropriately, as the situation demands. Adoption of agile methods also decreases the risk of failure of software development projects.

The Problem

Mostly, software development is a chaotic process characterized by the term „code fix“. Because of this unstructured development method, new features are more difficult to integrate as company size increases. In addition, the error rate mounts considerably and removing errors gets more intricate. Chaotic engineering methods and bureaucracy prolong the test stages for such systems and prevent reliable scheduling of tests and bug-fixes.

The Method

Agile process models replace the prevailing process standards by just a few important stages, disciplines, and activities. Instead of meticulously detailing stages and disciplines, only a few essential actions are specified per stage and discipline. Applying proved and tested methods and procedures (Best Practice) is of particular importance here.

The following principles are characteristic:

- Being open to changes instead of rigidly clinging to schedules
- People and communication are more important than processes and tools
- Results-oriented, not process-oriented
- Active integration of all team members
- Best Practice from experience instead of prescribed standards
- Suitability instead of extremism

The highest premise regarding processes is: Do as little as possible, but as much as necessary.

The Solution

Irrespective of your current development status, software maturity can be improved step by step using easily integrated methods. Depending on development status, the following methods are available:

- Versioning
- Unit tests, module tests
- Code reviews
- Automated testing
- Rapid Application Development (RAD)
- Extreme Programming
- Continuous Integration

ITQ