MIMACOM

www.mimacom.com

mimacom path toolchain

mimacom path toolchain creates the right technological environment for efficient development, along with extensive know-how and experience accumulated over many years. Standard products that are ideally matched are used to comprehensively cover all areas of modern software development. The modular design is pivotal, as it allows the various components to be integrated with the client's existing systems.

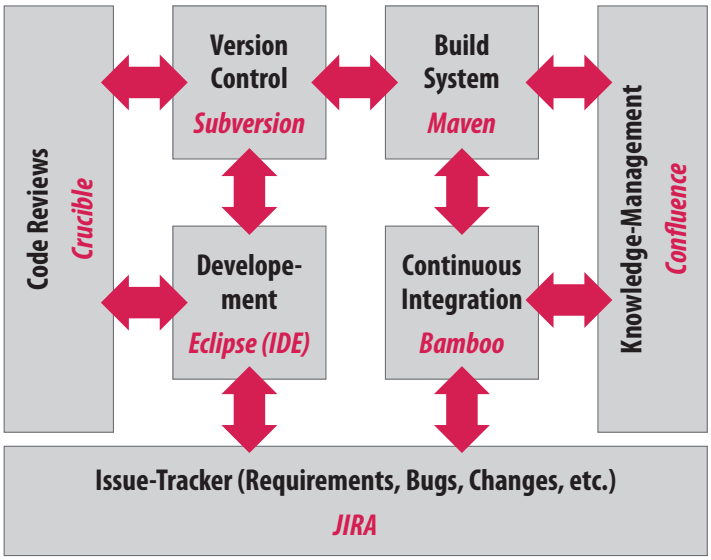
The mimacom path toolchain contains a 'best-of-breed' selection of tools from many different areas of software development. These tools are well-matched and facilitate a seamless development that is traceable at any stage. Optimum support of agile methods and technologies, which are part of the mimacom application stack, play a central role. Furthermore, continuous integration provides the required quality assurance.

Based on Java, the mimacom path toolchain includes

- Subversion for version control
- Apache Maven for modern application building and dependency management
- Eclipse as integrated development environment including the required plug-ins
- Jira for issue tracking and agile management
- Confluence for a structured knowledge management
- Bamboo for ongoing integration, testing and reporting
- Crucible and FishEye for code review and code inspection

Transparency and traceability when using these tools are of the utmost importance to us, not only with regard to the customer but also the team and the management. This is why we aim to initiate an active discussion with our clients and provide them with access to the relevant parts of the product at an early stage. Of course, you decide which data is made available when and to whom.

The interaction of the individual tools with each other and the connection with the customer's existing systems are facilitated by means of various plug-ins and extensions in a simple and straightforward manner.



Customer benefits

- Overview of requirements, features and issues, as well as their status
- Traceability and backtracking of the requirements through to functioning code
- Optimum distribution of know-how and long-term backup by means of ongoing, automated documentation
- Standardized and customized reports, as well as shared dashboards on your projects
- Transparency during the development and quality assurance through continuous integration
- Modern, integrative and well-matched development environment
- Integration of new modules with existing systems