

**Project Note**

# Harmonised Build-Toolset Promotes Productivity

Zühlke proposed and implemented a Java application build environment made of low-cost integrated components, leading to full traceability of requirements and dependencies, high reliability and faster return on investment.

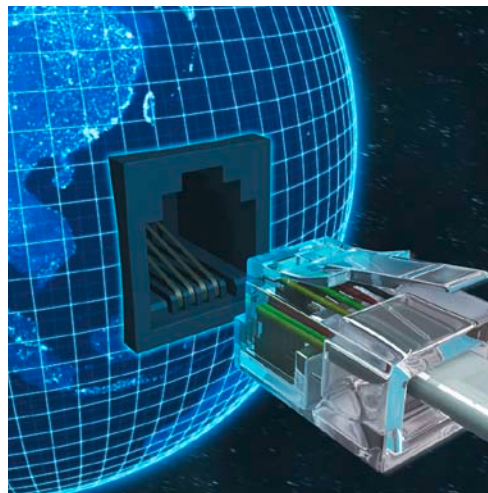
**Tasks**

A leading international investment bank is developing an ambitious series of Web 2.0 applications that place unprecedented levels of interactive control over fixed-income bond portfolios in the hands of institutional clients. The development team must be enabled to use best-of-breed tools and components while, to comply with legislation, guaranteeing repeatable builds with full traceability of all requirements and all ingredients of every software release. Change must be easy to manage, so the team needs low-impedance support for demand and issue management, continuous integration and testing, quality assurance and release.

**Implementation**

Zühlke proposed, implemented and maintained an integrated environment to support project managers, analysts, architects, developers, testers and operational staff throughout the lifecycle. Principal components of the solution comprised

- Atlassian Confluence for on-line documentation and knowledge sharing
- Atlassian JIRA for lightweight requirements management and issue tracking
- Tigris Subversion for source code version and configuration management
- Maven 2 for dependency management and build control
- JUnit 4 for unit testing
- JetBrains IntelliJ IDEA for Java developers and a specialised Eclipse IDE for GUI developers
- JetBrains TeamCity 2 for continuous integration, test and deployment
- Project DB – an Oracle database continuously updated with information from the SCM repository (Subversion) to support ad hoc queries
- Cenqua FishEye for repository browsing, searching and analysis



- Cenqua Crucible for online code reviews
  - The bank's in-house approval, packaging and release mechanisms
- The combination of these components enabled the bank to
- Reliably manage complex interdependencies
  - Reduce development cycle time
  - Avoid releasing unneeded features
  - Create accurate release notes without superhuman effort
  - Avoid nasty surprises for users and customers
  - Keep architectural options open
  - Safeguard traceability of changes
  - Lower maintenance costs and accelerate turnaround for fixes

**Technical Data**

Workstations:  
Windows XP

Servers:  
Red Hat Enterprise Linux

Development languages:  
Java, scripting languages (particularly for GUI), SQL

**Customer benefits**

- Zühlke was able to exploit existing knowledge of some of the tools, together with its numerous contacts within the bank, to hit the ground running
- Zühlke helped the customer to understand which investments would provide the quickest payback
- Zühlke transferred knowledge to the development team, allowing it to take responsibility for further maintenance.



Zühlke Engineering AG  
Wiesenstrasse 10a  
8952 Schlieren (Zurich)  
Switzerland

Phone +41 44 733 6611  
Fax +41 44 733 6612  
info@zuehlke.com  
www.zuehlke.com