



“IKAN ALM manages your software development wherever it is done – in-house, distributed and global, outsourced, in a satellite office or even at home.”

Managing or controlling software development without the proper tools to do so, is like flying blind, hoping for the best.

**IKANALM**  
ALM through evolution

[www.ikanalm.com](http://www.ikanalm.com)

## Know the status of the development process

Software development is more and more done at multiple distributed locations: distributed and global, outsourced, in a satellite office and by telecommuting developers at home. With the increasing complexity and growing volume of software development, many problems related to multi-site development emerge. Even when a lot of care is taken to ensure the proper communication process, integration problems and accidents do still happen. Therefore, all participants in the development process need a clear view of the project status, and management must have full control of all development phases and software artifacts “on the move”.

IKAN ALM deals with these issues, as it offers a secure yet flexible process-centric software change management solution for both local and distributed development teams. It complements existing version management tools by automating the complete software life-cycle management process, offering a single point of control and delivering support for your build, deploy, release and software life-cycle management and associated approval processes. Thanks to IKAN ALM your entire project team will work in a more efficient and cost-effective way, being kept continuously informed about the status of the project.

**IKAN**  
DEVELOPMENT



## Use your favorite versioning system and avoid conversion costs

There exist many good version management systems on the market today, offering complete versioning control. However, when building, deploying and tracking the life-cycle of your software applications, you can lose sight of your code as version control tools typically do not cover those areas. Many users would like to evolve to this next level, without having to give up their favorite versioning tool. IKAN ALM solves this problem by providing plug-and-play, out-of-the-box integration and enhancement for existing version management systems. IKAN ALM continues where version management systems stop, and helps project teams to further automate the complete software life-cycle management process. Thanks to this integration, you are free to use the IDE of your choice, as all of the IDE products have their own optimized versioning control integration. IKAN ALM even supports mixed environments in which a project or subproject uses sources from different version management systems.

## Eliminate build complexity and gain an accelerated release cycle

IKAN ALM enables you to automate, accelerate and simplify software builds ensuring that your software applications always get built correctly. Early feedback on broken builds or failed tests is very important since this information will dramatically reduce the time to identify and fix such errors, and consequently, will also cut down overall development time. A controlled build process is provided, independent of the versioning tool used. It offers you the possibility to reuse existing build/test or deploy scripts from leading scripting tools like Ant, NAnt and Maven 2, reducing build complexity and solving complex deployment issues.



## Monitor development and control the activities

Building a software solution involves many complex processes, roles and deliverables, which need to be managed so as to fit together. Streamlining these processes imposes a major effort. A truly comprehensive software change management (SCM) solution not only manages the simple versioning of your source code files. It also facilitates support for a continuous integration process and accurate and predictable build management, provides the ability to deploy the end result, ensures approval processes and manages the complex run-time dependencies of the application. Auditing facilities provide feedback on what has changed, when it was changed, who made and who approved the change. Finally, approval and notification processes need to be implemented for every step in the life-cycle, thereby improving the communication within the entire project team.

As IKAN ALM offers all these functionalities, it is both the glue that holds together the various components and phases of the application life-cycle, and the oil that lubricates the smooth and efficient interaction of those components. It delivers automated workflow to drive a continuous flow of activity throughout the development life-cycle and efficiently coordinates and streamlines development changes.

## Control yet flexible life-cycle process without constraints

Developing applications is not just about writing code. Once code has been developed, it must be tested, approved and delivered to a live environment. Therefore, the correct software change workflow process must be in place to fulfill the needs of the entire project team. IKAN ALM provides a secure and flexible process-centric framework for individual, local and distributed development teams. A tailored life-cycle from development to production can be implemented, including testing and quality assurance, offering a comprehensive framework across all major platforms including Windows, UNIX, Linux and z/OS main-frame systems.

IKAN ALM also supports a stream-based project model, meaning that a project manager has the ability to add a life-cycle to each version of a product. IKAN ALM has state-of-the-art architecture to control remote build and deploy servers, and offers a one-click interface to bring your application to life on as many servers as you want, wherever and whenever you want. In the event something still does go wrong, you can easily roll-back to a previous release, protecting yourself from nightmare scenarios like production shutdown.

## Make sure that all compliance rules are in place and be able to prove it

IKAN ALM facilitates compliance with legislation like the Sarbanes-Oxley Act and numerous similar state-specific acts and improves an organization's ability to conform to popular process standards (such as the CMM, ITIL, Six Sigma™). It can play an important part in achieving this compliance through automation of the business process, including all software development processes, by providing a reliable, repeatable, recordable and auditable process throughout the development life-cycle.

## Controlled freedom for every participant in the development process is the key to success

IKAN ALM enables organizations to adjust process control at user, team and project level, offering the development and the IT organizations unlimited flexibility and scalability. On the one hand IKAN ALM relieves the developer of the concerns of building and deploying an application, while on the other hand it gives the developer the comfort of continuing to work with his favorite IDE and its related versioning system. IKAN ALM delivers powerful automated workflow to drive a continuous flow of activity through the development life-cycle and efficiently coordinates and streamlines development changes, giving the organization the freedom to focus on core business.

## Life-cycle Management for any IDE

Through the version management system integration, you are able to use any of the popular IDEs, including IBM RAD, Microsoft Visual Studio or Oracle JDeveloper, providing a plug-and-play, out-of-the-box solution. The integration enables the promotion of changes through the life-cycle, from development to production and at any stage in between. Moreover, IKAN ALM will take care of the configuration, process and release management functions. With IKAN ALM you can guarantee the quality of the code and the deployment process, and provide a traceable and auditable logging for each step in the life-cycle. The final goal is a comprehensive end-to-end framework, where not only the needs of the developers are met, but also the interests of all other stakeholders.





## Key Benefits

- ⇒ Fully ready for Oracle Warehouse Builder, Oracle JDeveloper, Eclipse, IBM RAD, Microsoft Visual Studio and other popular IDEs
- ⇒ VCR support for Serena® PVCS® Version Manager™, Subversion, CVS, IBM Rational® ClearCase® (UCM and Base), IBM Rational® ClearCase® LT, Microsoft® Visual SourceSafe®
- ⇒ Keep working with your existing tools for development, versioning, testing, tracking, ...
- ⇒ Controlled build process independent of the versioning tool
- ⇒ Easily reuse existing build/test/deploy scripts from leading scripting tools like Ant, NAnt, Maven2, Make
- ⇒ Early developer feedback: when builds fail, the responsible developers are notified and corrective action can be taken quickly
- ⇒ Reduced build complexity and accelerated release cycles
- ⇒ First-class usable web interface with clear validations and tests to speed up the administration process. No more XML tweaking to setup a Continuous Integration process.
- ⇒ Controlled release process for building/rebuilding and delivering applications independently on the logical (Development, Test, QA, Production) and/or physical environment (Windows, Linux, Unix, z/OS)
- ⇒ Flexible Build configuration
  - Able to manage a pool of build servers
  - Able to do parallel (re)build on multiple platforms. (Different Windows OS, Windows + Linux environments, different Unix machines, build on Mainframe, ...)
  - Support for distributed builds
- ⇒ Speed up long standing bottlenecks in the development process thanks to features like incremental build and partial deploy
- ⇒ Support for both distributed and non-distributed platforms
- ⇒ Solves complex deployment issues like addressing servers in the DMZ or keeping live clustered servers
- ⇒ Notification and Approval facilities for individuals and groups
- ⇒ Support for multiple life-cycles: main development, maintenance release, emergency fix, parallel development, ....
- ⇒ Ensuring the “source to load” link in every stage of the life-cycle
- ⇒ Traceability: Easy to answer questions like who, what, where and when. Directly know which version of build artifacts are deployed on the test/qa/production server. See in an instant the status of a project: which version is currently in production, what is the QA team testing, which bug fixes have been finished by development, what is the status of the offshore development project, which version of which dll/jar/copybook is being used to build your project
- ⇒ Set up an Agile non-intrusive SCM process which suits all your developers, testers, auditors, build- release- and project managers
- ⇒ One-click interface to bring your application to life on AMAYW\* servers in parallel \* (as many as you wish)
- ⇒ Easily roll-back to a previous release
- ⇒ Integrated with the security system in place
- ⇒ Promotes reuse thanks to project dependencies
- ⇒ IT maturity model compliancy (CMM, ITIL, Prince, Prince 2, Six Sigma) and legislation like Sarbanes-Oxley



## Technical Facts

- ⇒ Required Java Environment
  - J2SE v 1.4.x or v 5.0 (1.5.X) SDK
- ⇒ Supported Application Servers
  - Tomcat 5.x
  - Oracle Application Server
  - BEA Weblogic
  - JBoss Application Server
  - Other Application Servers should be able to run the IKAN ALM web application
- ⇒ Supported Operating Systems
  - Windows 2000 : Professional, Server, Advanced Server
  - Windows XP Professional
  - Windows 2003 Server
  - Linux flavors : Red Hat Fedura Core 3, Xandros Desktop OS 3.02 Open Circulation Edition, SUSE Linux Professional 9.3, Other Linux distributions should be able to work with IKAN ALM including Linux 8.0 and 9.0, SuSe Linux 7.3 and 8.0
  - Unix flavors : HP Unix, Solaris, etc.
- ⇒ Client Web Browser (Java scripting must be enabled)
  - Internet Explorer 5.5 (SP 2), 6.x, 7.x
  - Mozilla FireFox 1.x, 2.x
  - Netscape Communicator 7.x
  - Opera web browser 7.x
- ⇒ Supported Version Control Systems
  - CVS 1.x
  - CVS 2.x
  - IBM Rational ClearCase (Base and UCM)
  - IBM Rational ClearCase LT
  - Subversion 1.x
  - Microsoft VisualSource Safe 6.x, 2005,
  - Serena PVCS Version Manager
- ⇒ Supported build tool integrations
  - Ant 1.4.x - 1.7.x
  - WSAD 5.x AntRunner
  - NAnt 0.85
  - Maven 2.0.x
- ⇒ Supported deploy tool integrations
  - Ant 1.4.x - 1.7.x
  - NAnt 0.85
  - Maven 2.0.x
- ⇒ Supported mail integration
  - SMTP
  - Lotus Notes 6.x



**IKAN**  
DEVELOPMENT

IKAN Development NV  
Schaliënhoevedreef 20a  
B-2800 Mechelen, Belgium  
Tel : + 32 15 44 50 40  
Fax: + 32 15 44 50 41  
info@ikan.be  
www.ikan.be