Title	
Author(s)	Pimmuang, Adirake
Citation	
Issue Date	2004-09
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1886
Rights	
Description	Supervisor: , ,

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

Research on Software Configuration Management in Middleware Systems

Pimruang Adirake

School of Information Science,
Japan Advanced Institute of Science and Technology

August 13, 2004

Keywords: component-based development, component deployment, SCM, Middleware

1. Background and Purpose

The middleware is now widely used in business e.g., J2EE, .NET, CORBA. In middleware, the developers can reuse existing components developed by in-house development or provided by other organizations. Each organization can develop components independently in different areas. Some components in an organization can be provided via the Internet to deployment processes of other organizations. Developers need to handle versions of interrelated components in both of development and deployment.

In this research, we develop J2DEP, a system to generate and manage component dependencies in development phase, and apply them to automate deployment. The key feature of J2DEP is to manage the dependency information between in-house components and third vendor components. Also, J2DEP can manage the dependency information in the connection with configuration management system. It uses the dependency information in release and deployment to provide consistency set of components.

2. Problem and Motivation

In this research, we specify scope of the research into component-based development deployment in J2EE middleware. The problem is about how to manage the dependency among the interrelated components in local SCM to support component-based development-deployment.

Since traditional SCMs are much scope in managing of text files, developers need the system to manage the change of both source and binary version of component. In J2EE component

development phase, developers should define component dependency properly in development phase and manage in local SCM, so that every developer can use the dependencies of components in Application Assembler and Deployer. After defining dependency information in development phase, developers can get a dependency component source from different repositories or release server automatically. Also, in deployment phase, component deployer can deploy the consistency set of components into middleware.

1. We summarize the problem raised in component development-deployment process following: Dependency information defined in development phase may not be used in release and deployment phases automatically.

2. Dependency in development tool phase and deployment phase may be different because of the loss of dependency information caused by combining components into a new component.

3. SCMs could not import different kinds of component developed by different organizations into local development environment automatically. The component may be packaged into the application both component source and binary format. Dependency information may not come with the components.

3. The approach

J2DEP research project addresses support in component development-deployment process. This system integrates the functionalities of configuration management, component development and component deployment.

The key insight of this research is to manage the evaluation of third vendor component inside local configuration management. Rather than to bring and control every version of third vendor components in local configuration management, J2DEP supports developers to generate the dependency information as a *dependency metadata* keep and manage only versions shown by metadata in local configuration management instead.

Developers can use J2DEP to import third vendor component into their development space and generate dependency metadata describing about component name, version, type and how to obtain component.

Also, J2DEP connects development space with configuration management API (CM API) for managing version of component sources and their dependencies and connects with release site to publish components with dependency metadata. To support consistency of component version in deployment phase, J2DEP uses the dependency metadata defined in development phase.

4. J2DEP Implementation

We develop J2DEP, a tool supporting variant kinds of J2EE components. J2DEP prototype implementation is built by integrating CVS for managing source and dependency metadata,

HTTP/FTP server for publishing the binary component and we use JBOSS middleware as a platform to deploy J2EE component. J2DEP consists of 2 main support tools following:

4.1 Component Development Support Tool

This J2DEP development support tool supports developers by following functionalities: *Development Space Initialization, Dependency Component Import Tool, Component Packaging, Developer Deployment Support and Component Release Tool.* Each functionality supports to manage the dependency of component in development space.

J2DEP helps developer to create a new development space by creating a new component or pulling an existing component from source repository to development space. This tool also supports developer to import dependency component both from in-house development and third vendor component to development space. Moreover, J2DEP supports to build the java source files and package compiled sources into binary component automatically.

After developer finish testing the component in development space, developer can release the binary component for developer download component to new development space or for user to use the component in his work. To release the component, J2DEP connect with FTP server and put the binary version of component to server. Developer or user can download with Development Support Tool or User deployment agent

4.2 User Deployment Agent

User Deployment Agent supports to deploy and undeploy the consistency set of binary components to final user site.

5. Conclusion

This research shows J2DEP, the integration of component-based development-deployment and software configuration management in middleware. This system is designed to support managing the interrelation of in-house components and third vendor components. J2DEP links between the component development and component deployment. The dependency metadata defined in development phase and component source can be controlled in local SCM. J2DEP uses dependency metadata to support component building, releasing phase of developers.   Also, J2DEP helps developer or deployer to deploy consistency set of components in both development site and user site.

J2DEP is now implemented as a prototype. This tool is still needed several functionalities of development tool. In the future, it would be benefit to integrate J2DEP with the other development tools, so that developer can gain benefit from all-infra structure of the other development tools, for example, implement J2DEP as an Eclipse plug-in etc.