

Spike Watch - Build System

Surendra Singhi
&
Hari Kodungallur

Spike Source
Redwood City, CA

2005-June-10

Part I

Introduction

Problem

Make a system such that it will:

- *Download source code of open source components from the Internet.*
- *Build the components on various platforms.*
- *Test them - run unit tests, code coverage tests and other such test.*
- *Publish the results.*

Potential benefits?

- The above tasks are very time consuming, and resource intensive, so it relieves the development team from the of hassles of periodic building and testing of applications.
- Will help in easy migration and stability of applications across range of platforms.
- We are third party, who are certifying them across platforms.

Toys for fun time!!!

- Ant
- Maven
- Bash shell scripting
- Java
- CVS
- SVN
- Make
- Autoconf

What should the user do?

- `svnUrl=http://svn.apache.org/repos/asf/struts/apps/trunk` (or `cvsRoot` and `cvsModule`)
- `username=`
- `password=`
- `build.always=`
- `ignore.file.patterns=*.html`
- `build.initialize=`
- `build.tool= maven — ant — make`
- `configure.options` (if using `make`)
- `test.tool=junit`
- `coverage.tool=clover — bullseye`
- `build.target=`
- `test.target=`
- `clean.target=`
- `test.coverage.target=`
- `vendor.name=Spring Framework`
- `license.type=ASL`
- `project.url=http://www.springframework.org/`

Part II

Design & Implementation

Objective

Check if the component has changed since last build. Download the component and do the building and testing only if anything has changed since the last build.

CruiseControl

- It requires that the program run in the background continuously.
- Uses cvs log commands which requires the component to be checked on to the local machine.

Our Implementation

- Makes use of CruiseControl classes.
- For SVN we have a Wrapper class which just adds some extra functionality.
- Implemented our own Java class for CVS, which uses “ cvs history” command. The output is then parsed, to decide whether the repository has changed, what files have changed, etc.

Build Process

- Check out the component.
- Using existing template the build script creates maven files for the component.
- The maven file then builds and tests the component, using whatever tool the component requires.
- Use clover or bullseye to do code coverage analysis.

Current Status

- Up and running.
- Geronimo, lucene, nutch, MySQL, JetSpeed were already being built.
- Recently we added Hibernate, Spring, Tomcat5, Struts

Part III

Conclusion

Road Ahead

- Adding few more components which uses make, to see if the build framework for make requires refinement.
- Maybe support for other source control mechanisms.
- Change the web-view.
- Add PHP-coverage tool.
- If you want to add any tool to this build process, please feel free to create a property file, and email it to me.

Questions?

Suggestions!!!

Thank you.