Version control systems

Version control system (VCS) is a system that records changes to a file or set of files over time so that you can recall specific versions later.

It allows you to revert files back to a previous state, revert the entire project back to a previous state, compare changes over time, see who last modified something that might be causing a problem, who introduced an issue and when, and more.

Local version control systems

RCS, which is still distributed with many computers today. Even the popular Mac OS X operating system includes the rcs command when you install the Developer Tools. RCS works by keeping patch sets (that is, the differences between files) in a special format on disk; it can then re-create what any file looked like at any point in time by adding up all the patches.

Centralized Version Control Systems (CVCS) - these systems, such as CVS, Subversion, and Perforce, have a single server that contains all the files versions, and a number of clients that check out files from that central place.

Distributed Version Control Systems

In a **DVCS** (such as **Git**, **Mercurial**, **Bazaar** or **Darcs**), clients don't just check out the latest snapshot of the files: they fully mirror the repository. Thus if any server dies, and these systems were collaborating via it, any of the client repositories can be copied back up to the server to restore it. Every clone is really a full backup of all the data.

Build systems

Build automation is the process of automating the creation of a software build and the associated processes including: compiling computer source code into binary code, packaging binary code, and running automated tests.

The **two ways** build tools differ are **task orient** vs. **product-oriented.** Task oriented tools describe the dependency of networks in terms of a specific set task and product-oriented tools describe things in terms of the products they generate.

The advantages of build automation

- A necessary precondition for continuous integration and continuous testing
- Improve product quality
- Accelerate the compile and link processing
- Eliminate redundant tasks
- Minimize "bad builds"
- Eliminate dependencies on key personnel
- Have history of builds and releases in order to investigate issues
- Save time and money - because of the reasons listed above.

Build tools

- Make
- CMake
- Ant – uses XML file describes the software project being built, its dependencies on other external modules and components, the build order, directories, and required plug-ins
- Maven - used primarily for Java projects. Maven dynamically downloads Java libraries and Maven plug-ins from one or more repositories such as the Maven 2 Central Repository, and stores them in a local cache
- Gradle – uses Groovy-based domain-specific language (DSL) instead of the XML

Maven

Maven uses Convention over Configuration which means developers are not required to create build process themselves.

Developers do not have to mention each and every configuration detail. Maven provides sensible default behavior for projects. When a Maven project is created, Maven creates default project structure. Developer is only required to place files accordingly and he/she need not to define any configuration in pom.xml

POM file

Some of the configuration that can be specified in the POM are following:

- project dependencies
- plugins
- goals
- build profiles
- project version
- developers
- mailing list

POM file

Some of the configuration that can be specified in the POM are following:

- project dependencies
- plugins
- goals
- build profiles
- project version
- developers
- mailing list

Gradle

Gradle uses a directed acyclic graph ("DAG") to determine the order in which tasks can be run.

Gradle was designed for multi-project builds which can grow to be quite large, and supports incremental builds by intelligently determining which parts of the build tree are up-to-date, so that any task dependent upon those parts will not need to be re-executed.

Continuous integration tooling

Continuous integration (CI) is an integral part of an agile software development setup. Sprint after sprint, teams strive to "not break the build" while delivering incremental features. But when developers focus completely on adding features, code errors can sometimes creep in and render the software unusable. To stop such errors from being integrated into the software configuration management (SCM).

Jenkins

Jenkins was created after a falling out between Hudson project contributors and the company that acquired Sun Microsystems. It is developed in Java and can be installed using simple

```
java -jar jenkins.war
```

command after downloading the WAR file. You can also deploy it in servlet containers.

Jenkins' focus is on two major jobs: building/testing software projects continuously and monitoring externally run jobs.

For the former, Jenkins supports an array of SCM tools - Git, Mercurial, Subversion, Clearcase, and many more.

Some popular CI systems: Teamcity, Jenkins, Travis CI

Configuration management systems

Configuration management (CM) is a systems engineering process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational information throughout its life.

Solved problems:

- Configuration identification
- Configuration control
- Configuration status accounting
- Configuration audits

Puppet

Puppet is designed to manage the configuration of Unix-like and Microsoft Windows systems declaratively.

The user describes system resources and their state, either using Puppet's declarative language or a Ruby DSL (domain-specific language).

This information is stored in files called "Puppet manifests". Puppet discovers the system information via a utility called Facter, and compiles the Puppet manifests into a system-specific catalog containing resources and resource dependency, which are applied against the target systems. Any actions taken by Puppet are then reported.

Chef

Chef is both the name of a company and the name of a configuration management tool written in Ruby and Erlang.

It uses a pure-Ruby, domain-specific language (DSL) for writing system configuration "recipes". Chef is used to streamline the task of configuring and maintaining a company's servers, and can integrate with cloud-based platforms such as Internap, Amazon EC2, Google Cloud Platform, OpenStack, SoftLayer, Microsoft Azure and Rackspace to automatically provision and configure new machines.

Chef contains solutions for both small and large scale systems, with features and pricing for the respective ranges.

Ansible

Ansible is a free-software platform for configuring and managing computers, combines multi-node software deployment, ad hoc task execution, and configuration management.

It manages nodes (Linux nodes must have Python 2.4 or later installed on them, Windows nodes require PowerShell 3.0 or later) over SSH or over PowerShell.

Modules work over JSON and standard output and can be written in any programming language. The system uses YAML to express reusable descriptions of systems.

Deployment tools for Desktop applications. JavaWS

Java Web Start (also known as JavaWS) is a framework developed by Sun Microsystems (now Oracle) that allows users to start application software for the Java Platform directly from the Internet using a web browser.

Unlike Java applets, Web Start applications do not run inside the browser. By default, they run in the same sandbox as applets, with several minor extensions like allowing to load and save the file that is explicitly selected by the user through the file selection dialog. Only signed applications can be configured to have additional permissions.

Deployment tools for Desktop applications. JavaWS

Web Start can also launch unmodified applets that are packaged inside .jar files, by writing the appropriate JNLP file. This file can also pass the applet parameters. Such applets also run in a separate frame. Applet launcher may not support some specific cases like loading class as resource.

Like applets, Java Web Start is cross-platform.

The developer prepares a special XML file with JNLP extension. This file describes the application requirements, code location, parameters and additional permissions (if any). The browser downloads this file as any other and (following its MIME type, application/x-java-jnlp-file) opens it with Web Start tool. Web Start tool downloads all necessary resources and launches the application.

Deployment tools for Desktop applications. ClickOnce

ClickOnce is a Microsoft technology that enables the user to install and run a Windows-based smart client application by clicking a link in a web page. ClickOnce is a component of Microsoft .NET Framework 2.0 and later, and supports deploying applications made with Windows Forms or Windows Presentation Foundation. It is similar to Java Web Start for the Java Platform or Zero Install for Linux.

The ClickOnce deployment technology has been integrated into the Visual Studio 2005 and later. It is also natively supported by MSBuild, Microsoft's build management technology.