

Continuous Integration



What is Continuous Integration?

- Continuous Integration is a software development practice where members of a team integrate their work frequently.
- Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible.
- When CI works well, it helps the code stay robust enough that customers and other stakeholders can play with the code whenever they like.
- Like refactoring, continuous integration works well if you have an exhaustive suite of automated unit tests that ensure that you are not committing buggy code.

CI overview

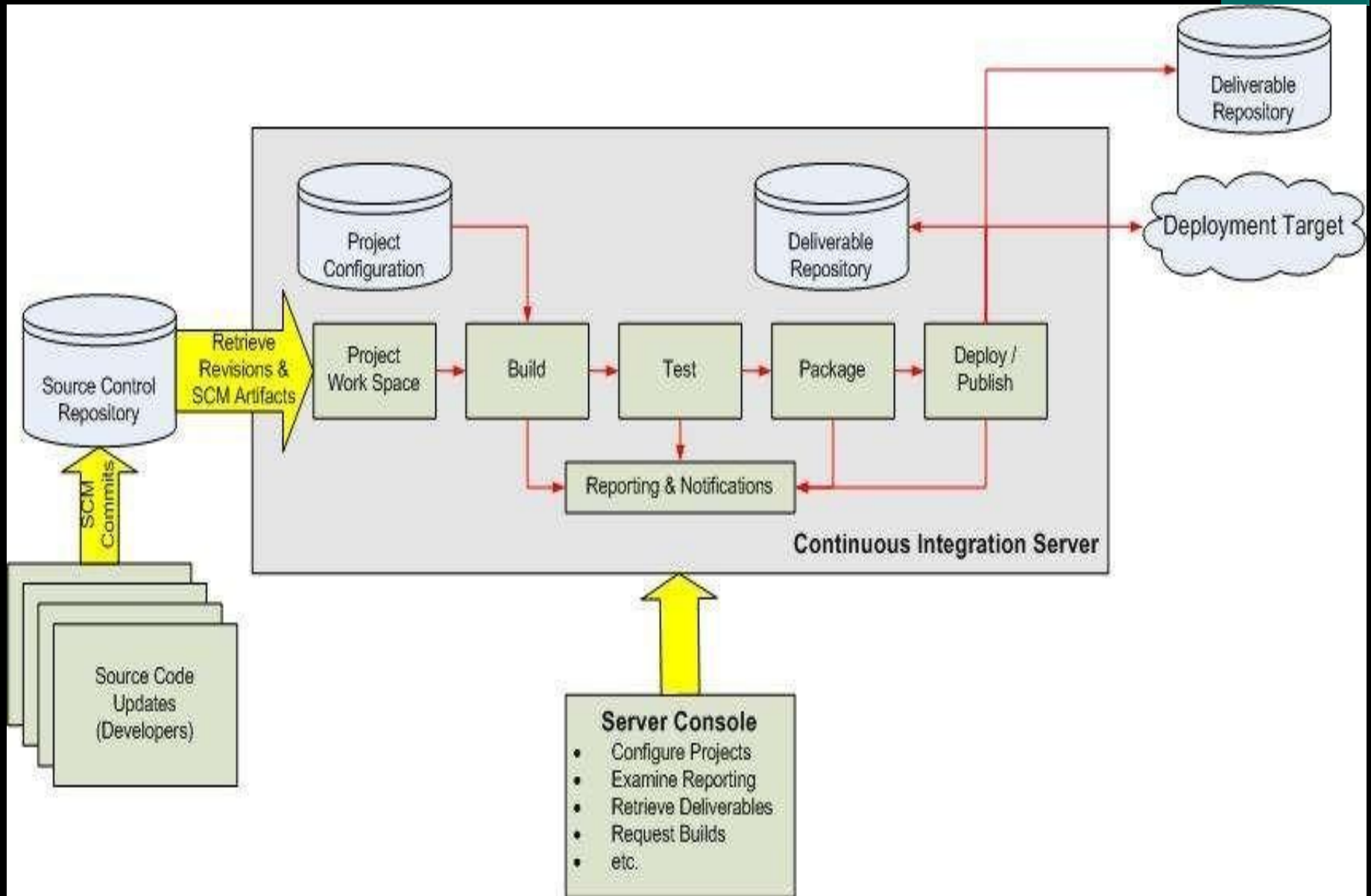




Image: <http://www.the-marketing-it-process-strategist.com/marketing-database-blog.html>

**“...members of a team
integrate work ...frequently...”**



Image: <http://www.the-marketing-it-process-strategist.com/marketing-database-blog.html>

**“...usually each person
integrates at least daily...”**



Image: <http://www.the-marketing-it-process-strategist.com/marketing-database-blog.html>

“...leading to multiple integrations per day.”



Image: <http://www.the-marketing-it-process-strategist.com/marketing-database-blog.html>

**“Each integration is verified by
an automated build...”**



Image: <http://www.the-marketing-it-process-strategist.com/marketing-database-blog.html>

**“...to detect integration errors
as quickly as possible.”**

PRACTICES OF CONTINUOUS INTEGRATION

- Maintain a Single Source Repository
- Automate the Build
- Make Your Build Self-Testing
- Everyone Commits To the Mainline Every Day
- Every Commit Should Build the Mainline on an Integration Machine
- Keep the Build Fast
- Test in a Clone of the Production Environment
- Make it Easy for Anyone to Get the Latest Executable
- Everyone can see what's happening
- Automate Deployment

Continuous Deployment

Build Pipelines

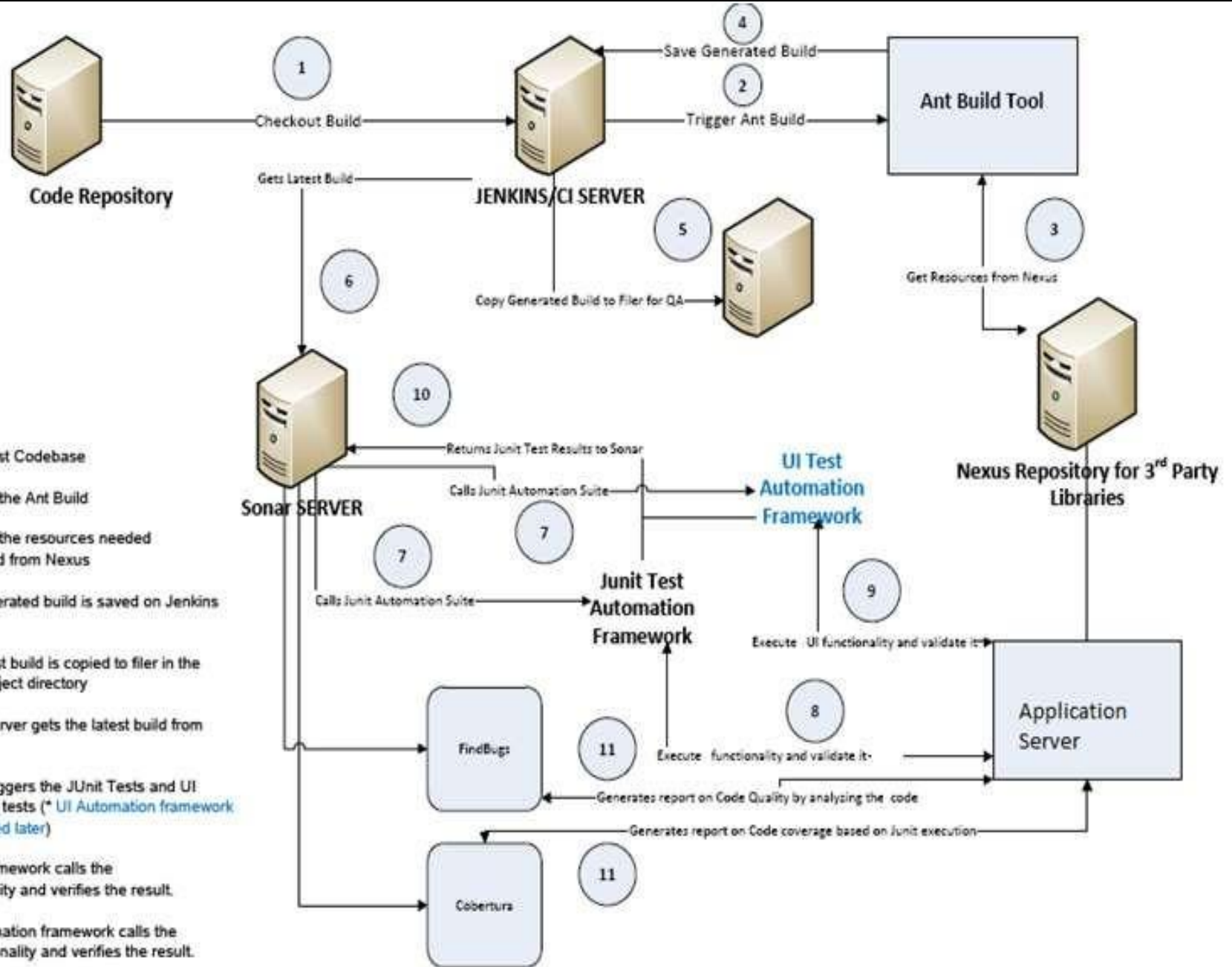
Deployment Automation

Configuration
Management

Continuous
Integration

Testing

Agile



1) Get Latest Codebase

2) Triggers the Ant Build

3) Ant gets the resources needed for the build from Nexus

4) The generated build is saved on Jenkins server

5) The latest build is copied to file in the specific project directory

6) Sonar server gets the latest build from Jenkins

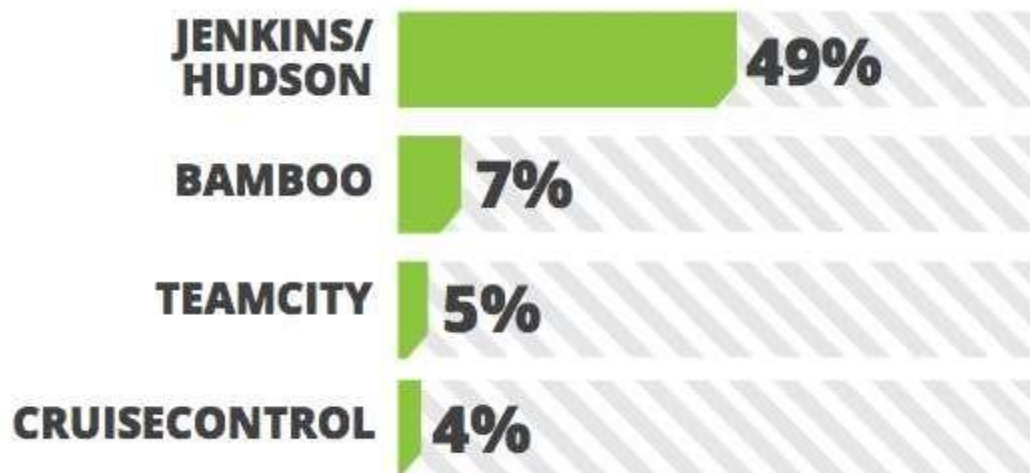
7) Sonar triggers the JUnit Tests and UI Automation tests (* UI Automation framework will be added later)

8) JUnit framework calls the functionality and verifies the result

9) UI Automation framework calls the UI functionality and verifies the result

10) JUnit framework and UI Automation framework returns the result to Sonar.

<u>Name</u>	<u>Platform</u>	<u>License</u>	<u>Windows builders</u>	<u>Java builders</u>	<u>Other builders</u>	Notification	<u>IDE Integration</u>	<u>Other Integration</u>
Bamboo	Servlet Container	Proprietary	MSBuild , NAnt , Visual Studio	Ant , Maven 1 , Maven 2 , Maven 3	custom script, command line, Bash	XMPP , Google Talk , E-mail , RSS , Remote API	IntelliJ , IDEA , Eclipse , Visual Studio	FishEye , Crowd , JIRA , Clover
CruiseControl	Cross-platform	BSD -style	NAnt , Rake , and Xcode	Phing , Apache Ant , Maven	catch-all 'exec'	E-mail , CCTray	Eclipse	Unknown
Go	Cross-platform	Apache 2.0	Yes	Yes	Cross-platform command-line	E-mail , CCTray	No	RESTful API
Jenkins/Hudson	Servlet Container	Creative Commons and MIT	MSBuild , NAnt	Ant , Maven 2 , Kundo	Cmake , Gant, Gradle, Grails , Phing , Rake , Ruby , SCons , Python , Shell script and Command Line	Android , E-mail , Google Calendar , IRC , XMPP , RSS , Twitter	Eclipse , IntelliJ , IDEA , NetBeans	Bugzilla , Google Code , JIRA , Redmine , FindBugs , Checkstyle , PMD and Mantis , Trac
TeamCity	Servlet Container	Proprietary	MSBuild , NAnt , Visual Studio , Duplicates finder for .NET	Ant , Maven 2/3 , IDEA ipr based, IDEA inspections, IDEA Duplicates finder, Gradle	Rake , FxCop , Command Line	E-mail , XMPP , RS S , IDE , SysTray	Eclipse , Visual Studio , IntelliJ , IDEA , RubyMine , PyCharm , PhpStorm , WebStorm	Jetbrains Youtrack, JIRA , Bugzilla , FishEye , FindBugs , PMD , dotCover , NCover
Team Foundation Server	Windows , VSTM	Proprietary	MSBuild	Ant, Maven	Custom script, Command line	E-Mail , SOAP	Visual Studio , Eclipse	Unknown



**“Continuous
Integration has
become a
mainstream
technique for
software
development”**



What is the value of continuous integration?



Reduce risk



Better project visibility



Greater software confidence



Deployable software anytime



Reduce repetitive manual processes





Image: <http://www.davistechnologiesllc.com/index-4.html>

The excuses for not continuously integrating...



Image: <http://www.davistechnologiesllc.com/index-4.html>

**Overhead to
maintain**



Image: <http://www.davistechnologiesllc.com/index-4.html>

**Too much
change required**



Image: <http://www.davistechnologiesllc.com/index-4.html>

**“The build
keeps failing...”**



Image: <http://www.davistechnologiesllc.com/index-4.html>

Additional hardware costs



Image: <http://www.davistechnologiesllc.com/index-4.html>

**Should be doing this
(manually) anyway**



Image: <http://www.daddyhogwash.com/2009/01/broken-window-theory-gets-a-boost-from-the-university-of-groningen/>

**DON'T PUT UP WITH
BROKEN WINDOWS!**