

Continuous Integration

Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly.

Martin Fowler

<http://martinfowler.com/articles/continuousIntegration.html>

Advantages

Continuous integration has many advantages:

- Rapid feedback on the "health" of the project
- Identifies defects early in the development cycle
- Enables project visibility at all levels
- Defects are often smaller, less complex and easier to fix
- The team has increased confidence to make changes
- Working software is always available.

*Interview with Paul Duvall, author of
Continuous Integration: Improving Software Quality and Reducing Risk*

Integration Servers

- CruiseControl
- CruiseControl.NET
- CruiseControl.rb
- CI Factory
- Dreambeat CI
- Tinderbox (1, 2, 3)
- BuildBot
- Anthill
- Anthill Professional
- Bamboo
- Luntbuild
- Luntbuild Professional
- Gump
- Draco.NET
- Continuum
- Sin
- Parabuild
- Pulse
- TeamCity
- Hudson

<http://confluence.public.thoughtworks.org/display/CC/CI+Feature+Matrix>

General Characteristics

- Supports you version control system
- Parallel, distributed builds
- Version control changes trigger build
- Inter-project dependencies
- Publish results (mail, RSS, etc.)
- Convenient interface (usually web)
- Historical perspective
- Supports you build mechanism
- Integrates with your issue tracker
- Integrates with your testing tools (e.g. JUnit, SilkCentral)
- Integrates with your IDE
- Integrates with your code inspector

The Practices

- Maintain a source repository
- Automate the build
- Make build self-testing
- Everyone commits every day
- Every commit (to mainline) should be built
- Keep the build fast
- Test in a clone of the production environment
- Make it easy to get the latest deliverables
- Everyone can see the results of the latest build
- Automate deployment

Example: Hudson

The screenshot shows the Hudson web dashboard in an Opera browser window. The browser title is "Dashboard [Hudson] - Opera" and the address bar shows "http://localhost:8080/". The dashboard has a blue header with the "Hudson" logo and a search bar. Below the header, there are navigation links: "New Job", "Manage Hudson", "People", and "Build History". A "Build Queue" section indicates "No builds in the queue." A "Build Executor Status" table shows two executors in an "Idle" state. The main content area displays a table of build jobs with columns for status, icon, job name, last success, last failure, and last duration. Two jobs are listed: "hello world" and "javacsv". A "Legend" section at the bottom right explains the icons used in the table.

Dashboard [Hudson] - Opera

File Edit View Bookmarks Widgets Tools Help

Dashboard [Hudson]

http://localhost:8080/

Hudson

ENABLE AUTO REFRESH

Welcome to Hudson (slow laptop) [edit description](#)

All fw +

S	W	Job ↓	Last Success	Last Failure	Last Duration
		hello world	1 day 23 hr (#2)	1 day 23 hr (#1)	3 sec
		javacsv	11 hr (#16)	N/A	38 sec

Icon: [S](#) [M](#) [L](#)

[Legend](#) [for all](#) [for failures](#) [for just last builds](#)

Build Queue
No builds in the queue.

Build Executor Status

No.	Status
1	Idle
2	Idle

Hudson ver. 1.252

100%

Install & Run

- Java installed
- Ant, Maven installed as appropriate
- \$JAVA_HOME, \$ANT_HOME, etc. set
- Download the latest Hudson [WAR file](#)
- `$> java -jar hudson.war`
- Go to <http://localhost:8080>
- Manage Hudson > Configure System
- Setup as appropriate (mostly JDK, mail server); Save

Simple Case

hello world [Hudson] - Opera

File Edit View Bookmarks Widgets Tools Help

hello world [Hudson]

http://localhost:8080/job/hello%20world/

Hudson

Hudson » hello world ENABLE AUTO REFRESH

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

Project hello world

The simplest job I could think of [edit description](#)

[Workspace](#)

[Recent Changes](#)

Build History		(trend)
	#2	Sep 13, 2008 9:53:20 AM
	#1	Sep 13, 2008 9:52:09 AM

[for all](#) [for failures](#)

Permalinks

- [Last build \(#2\), 1 day 23 hr ago](#)
- [Last stable build \(#2\), 1 day 23 hr ago](#)
- [Last successful build \(#2\), 1 day 23 hr ago](#)
- [Last failed build \(#1\), 1 day 23 hr ago](#)

Hudson ver. 1.252

100%

Advanced Case

javacsv [Hudson] - Opera

File Edit View Bookmarks Widgets Tools Help

javacsv [Hudson]

http://localhost:8080/job/javacsv/

Project javacsv

A simple Java project.

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

[Javadoc](#)

[Checkstyle Warnings](#)

[Coverage Report](#)

[Subversion Polling Log](#)

[edit description](#)

Test Result Trend

Build	Count
#12	100
#13	100
#14	100
#15	100
#16	100

[\(just show failures\)](#) [enlarge](#)

Checkstyle Trend

Build	Count
#13	1500
#14	2500
#15	1500

Build History [\(trend\)](#)

- #16 [Sep 14, 2008 9:42:13 PM](#)
- #15 [Sep 14, 2008 8:30:48 PM](#)
- #14 [Sep 14, 2008 3:01:06 PM](#)
- #13 [Sep 14, 2008 2:44:04 PM](#)

100%

Plugins

Over 90 plugins at:

- <http://hudson.gotdns.com/wiki/display/HUDSON/Plugins>

This example just uses two:

- Checkstyle: <http://checkstyle.sourceforge.net/> (static analysis)
- Cobertura: <http://cobertura.sourceforge.net/> (coverage analysis)

Questions?