



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

Sviluppo software in gruppi di lavoro complessi¹

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

Anno accademico 2017/18, I semestre



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

Lezione XII: Continuous integration & delivery



Continuous Integration

Configuration Management

Esplicitazione delle dipendenze

Test d'unità, d'integrazione, d'accettazione

Build automatizzate

Deployment simulato o automatizzato (domani)

↔ *Continuous Integration & Delivery*

Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

- Martin Fowler 2001-6 (<http://www.martinfowler.com/articles/continuousIntegration.html>)
- Tradizionalmente, l'integrazione è una delle parti più lunghe e rischiose dei progetti *software*
- CI \rightsquigarrow integrazione continua (incrementale) per minimizzare il rischio di fallimento
 - *Reduced Deployment Risk*
 - *Believable Progress*
 - *User Feedback*



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

- 1 Lavoro su una copia locale sulla *macchina di sviluppo*
- 2 *Build* vs. *Compile*: oltre alla costruzione esecuzione di test
- 3 *build* funzionante sulla *macchina di sviluppo*
- 4 Caricamento sulla *macchina d'integrazione*
- 5 *build* funzionante sulla *macchina d'integrazione*

Almeno una volta al giorno.



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

Cambiato drasticamente negli ultimi 10 anni. Fowler:

- “The current open source repository of choice is Subversion. (The older open-source tool CVS is still widely used, and is much better than nothing, but Subversion is the modern choice.)”
- “One of the features of version control systems is that they allow you to create multiple branches, to handle different streams of development. This is a useful, nay essential, feature — but it’s frequently overused and gets people into trouble. Keep your use of branches to a minimum.” (ma... suggerisce tecniche di *pending head* simili a quelle usate da *gitflow*)

Continuous Integration



Swigruppo

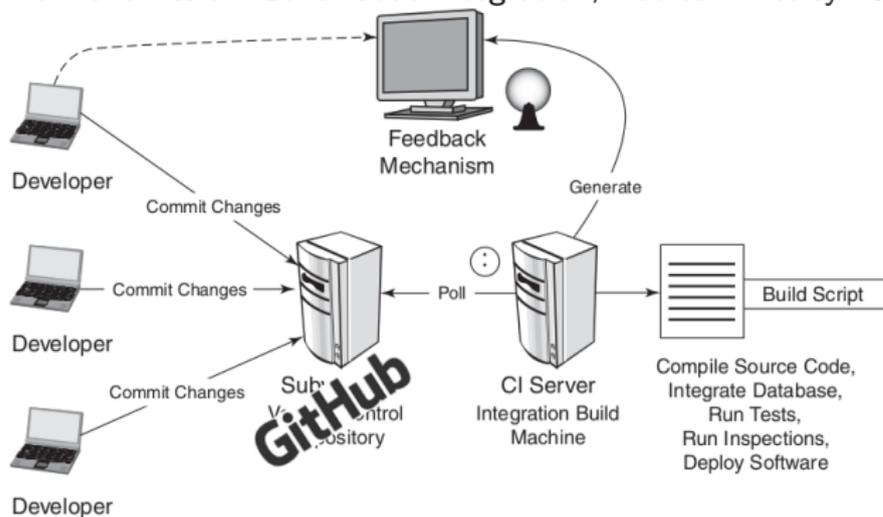
Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

Da Duvall et al. Continuous Integration, Addison Wesley 2007





Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

Il *build* non dovrebbe impiegare più di 10 minuti, altrimenti si perde il *feedback* immediato. Che fare se è necessario più tempo?

- *Deployment pipeline*: il *build* è spezzato in più fasi, *commit build*, *slower tests build*, ecc.
- Il *commit build* impiega meno di 10 min, il resto parte poi



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

- Il sistema dove avviene l'**integrazione** dovrebbe essere il più possibile "simile" a quello di **produzione**
- Il *deployment* verso l'ambiente di produzione può essere automatizzato (garantendo così un maggior controllo sulla effettiva configurazione in uso!)



Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

<https://git-scm.com/book/en/v2/Git-Tools-Submodules>

Permettono di tenere sotto controllo una 'dipendenza'

- una *sottodirectory* contiene un altro *repository*
- `git submodule add repo dirname`
- il *file* `.gitmodules` viene versionato
- `git clone --recursive`
- `(git submodule init) git submodule update` per ottenere la versione attuale del modulo
- ma nel progetto **congelò** una specifica versione `git add submoduledirname`



Cancellare un sottomodulo

Svigruppo

Monga

Riassunto

Continuous
Integration &
Delivery

Git
submodules

Non c'è un modo diretto, occorrono più operazioni:

- 1 Cancellare a mano in `.gitmodules`
- 2 Cancellare a mano in `.git/config`
- 3 `git rm --cached submoduledirname`
- 4 `commit` (e cancellare il sottomodulo)