UNIVERSITÀ DEGLI STUDI
DI MILANO

Sviluppo Software in Gruppi di Lavoro Complesso

prof. Carlo Bellettini

prof. Mattia Monga

a.a. 2020-21

Per fare domande:

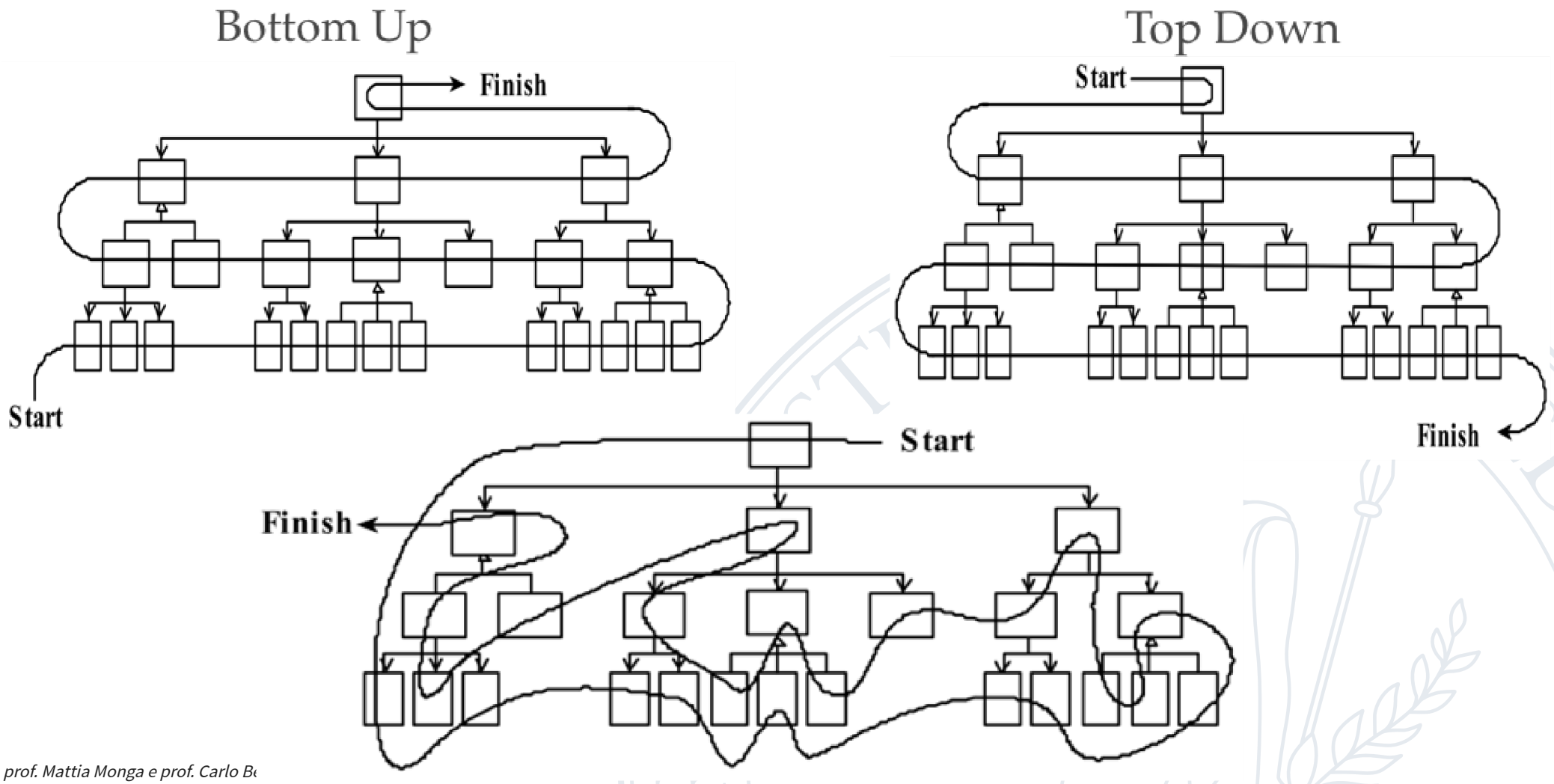
https://homes.di.unimi.it/bellettini/questions

codice di oggi 20102020

UNIVERSITÀ DEGLI STUDI
DI MILANO

7. Intro Continuous Integration

Integration



Unit vs Integration testing

Continuous Integration

Definizione:

> Allineamento frequente (es. "molte volte al giorno") dagli ambienti di lavoro degli sviluppatori verso l'ambiente condiviso (mainline)

- Non è più una fase... viene affogata nello sviluppo normale...
  - finita una parte (una feature, una patch, ...) si integra
  - l'integrazione diventa un NONEVENT

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.phdcomics.com/comics/archive.php?comicid=1531

Single Source Repository

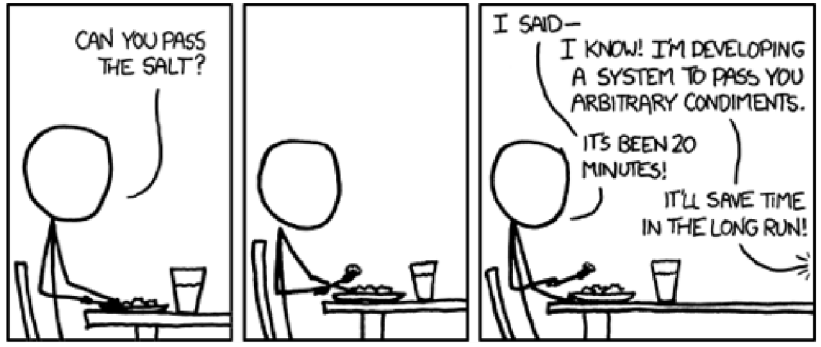
http://www.martinfowler.com/articles/continuousIntegration.html

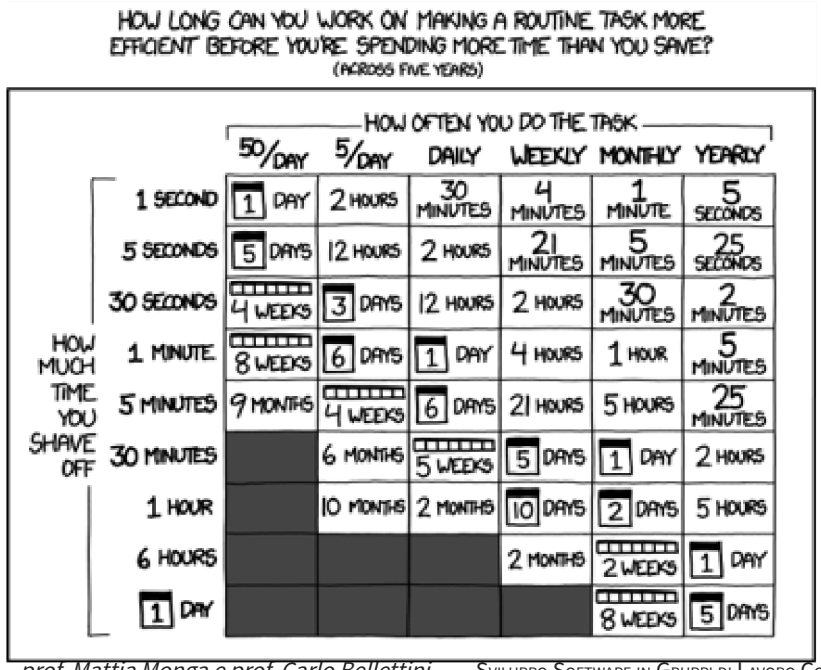
Capisaldi (Martin Fowler 2006)

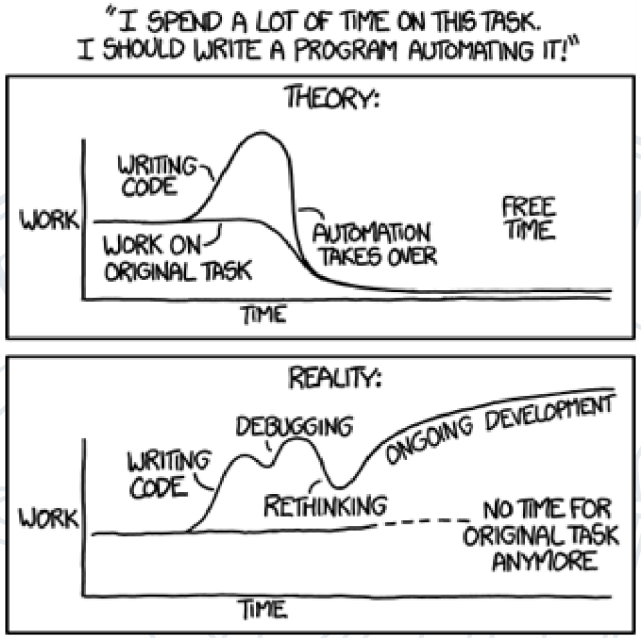
- Maintain a Single Source Repository
- Automate the Build
- Make Your Build Self-Testing
- Everyone Commits To the Mainline Every Day
- Every Commit Should Build the Mainline on an Integration Machine
- Fix Broken Builds Immediately
- Keep the Build Fast
- Test in a Clone of the Production Environment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

https://xkcd.com/974
https://xkcd.com/1205
https://xkcd.com/1319

Automate the build







http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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 Integration Machine
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

https://xkcd.com/303/

Keep the build fast

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

http://www.martinfowler.com/articles/continuousIntegration.html

Capisaldi (Martin Fowler 2006)

- 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
- Fix Broken Builds Immediately
- 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

Source code management (versioning)

> 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.
>
> M. FOWLER

Linus presenta git a Google

```
https://www.youtube.com/embed/4XpnKHJAok8
```