

# Configuration Management

---

## METODI E TECNOLOGIE PER LO SVILUPPO SOFTWARE

Nicola Bertazzo

nicola.bertazzo [at] unipd.it

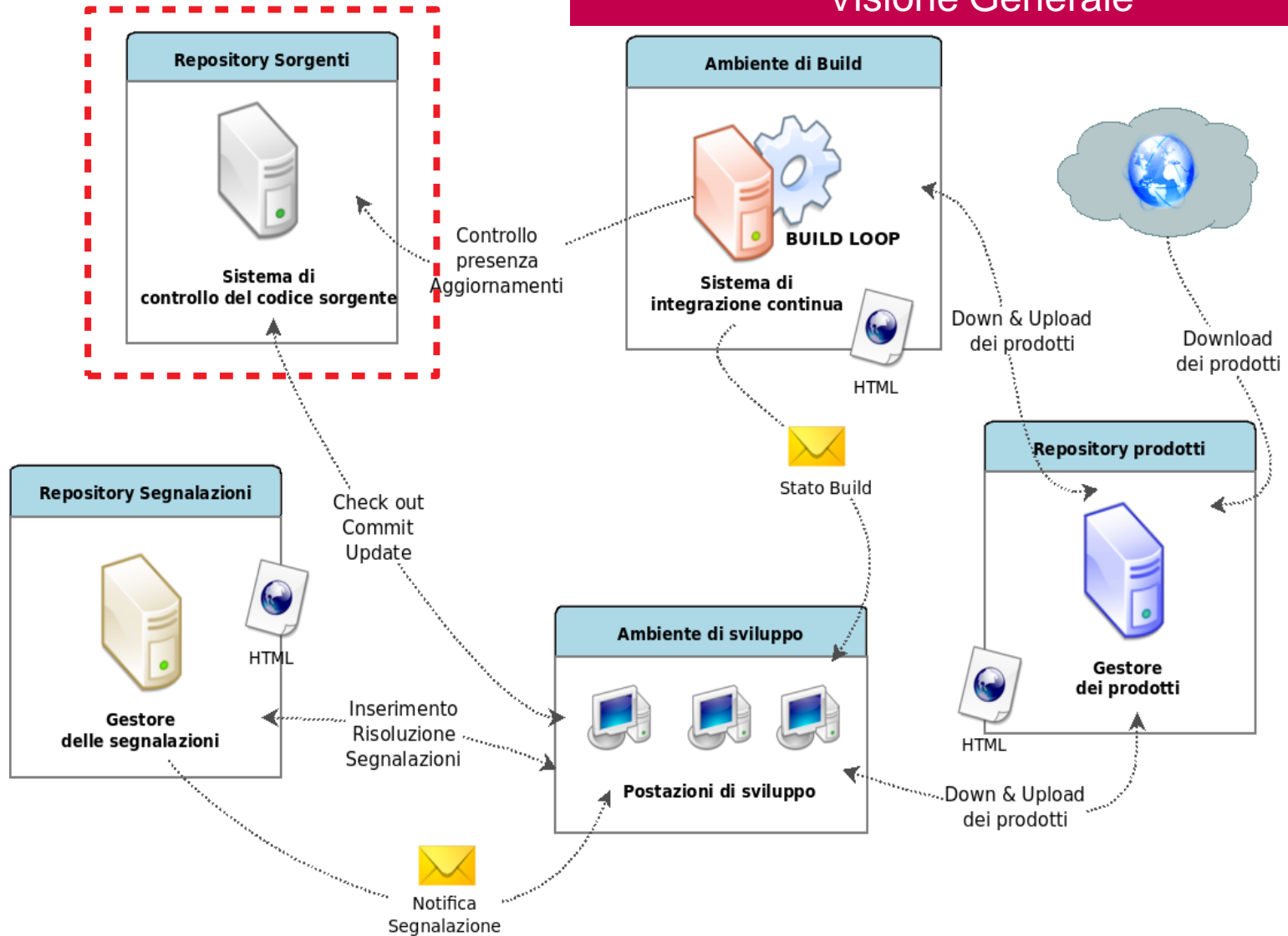
Università degli Studi di Padova

Dipartimento di Matematica

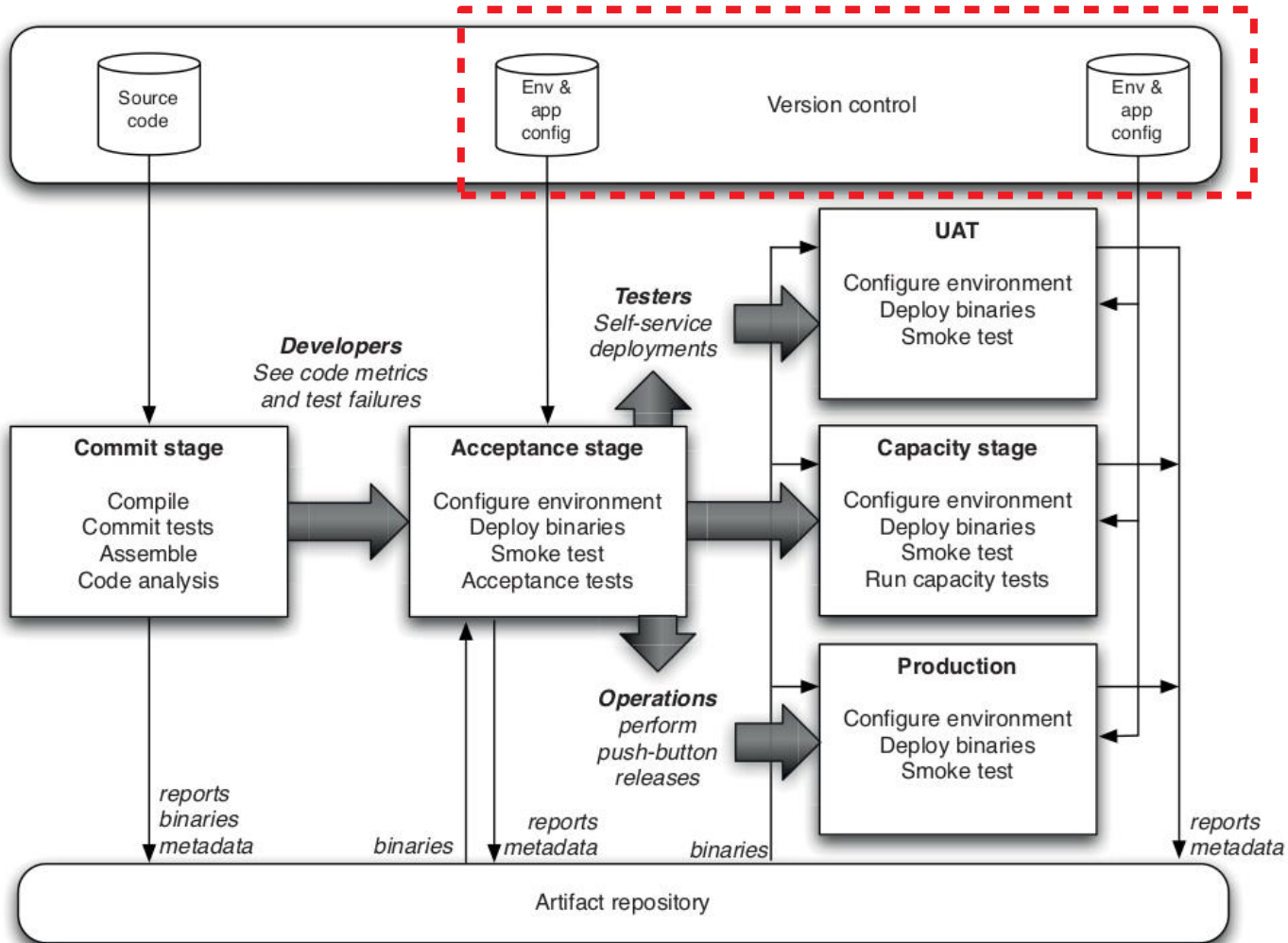
Corso di Laurea in Informatica, A.A. 2022 – 2023



Visione Generale



Visione Generale



<https://martinfowler.com/books/continuousDelivery.html> pag 111

## Definizione

**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**. The CM process is widely used by military engineering organizations to **manage changes throughout the system lifecycle of complex systems**, such as weapon systems, military vehicles, and information systems. Outside the military, the CM process is also used with **IT service management as defined by ITIL**, and with other domain models in the civil engineering and other industrial engineering segments such as roads, bridges, canals, dams, and buildings.



## Definizione (ITIL)

L'obiettivo del **Configuration Management** è di fornire un **modello logico dell'infrastruttura** attraverso l'identificazione, il controllo, la gestione e la verifica di tutte le versioni di "Configuration Items" esistenti.

**Il configuration item (CI)** è un unit di configurazione che puo' essere gestita individualmente (tipo: computer, routers, servers, software, etc...).

Un elemento chiave del processo è il Configuration Management Database (CMDB), che viene utilizzato per tracciare tutte le CI e le relazioni tra di loro (tipo: il server A ospita il Servizio B, etc...).

Alcuni benefici dell'implementare il processo di configuration management sono i seguenti:

- .Disponibilità di informazioni accurate sull'infrastruttura IT
- .Maggiore controllo sulle CI (potenzialmente costose)
- .Maggiore aderenza alle leggi (es. numero licenze software)
- .Miglior supporto al processo di Incident e Problem Management, soprattutto nella valutazione dell'impatto degli incidenti e nell'analisi della "root cause"

## Definizione (continuous delivery)

The traditional **software configuration management (SCM)** process is looked upon by practitioners as the best solution to **handling changes in software projects**. It identifies the **functional and physical attributes of software at various points in time**, and **performs systematic control of changes to the identified attributes for the purpose of maintaining software integrity and traceability** throughout the software development life cycle.

The SCM process further defines the need to trace changes, and the ability to verify that the final delivered software has all of the planned enhancements that are supposed to be included in the release. It identifies **four procedures** that must be defined for each software project to ensure that a sound SCM process is implemented. They are:

- 1) Configuration identification
- 2) Configuration control
- 3) Configuration status accounting
- 4) Configuration audits

## Obiettivi

**Reproducibility:** We should be able to **provision any environment in a fully automated fashion**, and know that **any new environment reproduced from the same configuration is identical**.

**Traceability:** We should be able to **pick any environment and be able to determine quickly and precisely the versions of every dependency used to create that environment**. We also want to be able to **compare previous versions** of an environment and see what has changed between them.

Vedi slide corso DevOps North Carolina State University:

<https://github.com/CSC-DevOps/Course/tree/Spring2020>

Slide:

<https://www.engineeringonline.ncsu.edu/course/csc-519-devops-modern-software-engineering-practices/>

[https://docs.google.com/presentation/d/1i18CWaZaiBBWPIT71iOuEPYYaZcK1VMvtdynwCLAVU0/edit#slide=id.g6e582f9f77\\_0\\_0](https://docs.google.com/presentation/d/1i18CWaZaiBBWPIT71iOuEPYYaZcK1VMvtdynwCLAVU0/edit#slide=id.g6e582f9f77_0_0)



## CM in AWS

**AWS OpsWorks** è un servizio di gestione delle configurazioni che fornisce istanze gestite di Chef e Puppet. Chef e Puppet sono piattaforme di automazione che permettono di utilizzare del codice per automatizzare le configurazioni dei tuoi server. OpsWorks utilizza Chef e Puppet per automatizzare il modo in cui vengono configurati, distribuiti e gestiti i server in tutte le istanze Amazon EC2 o in tutti gli ambienti di calcolo locali.

<https://aws.amazon.com/it/opsworks/>

**AWS CloudFormation** ti fornisce un linguaggio comune per descrivere ed effettuare il provisioning di tutte le risorse dell'infrastruttura nel tuo ambiente cloud. Con CloudFormation puoi usare un semplice file di testo per modellare ed effettuare il provisioning, in modo automatizzato e sicuro, di tutte le risorse necessarie alle tue applicazioni su tutte le regioni e tutti gli account. Questo file sarà l'unica sorgente del tuo ambiente cloud.

<https://aws.amazon.com/it/cloudformation/>

<https://cloudcraft.co/>

[https://en.wikipedia.org/wiki/Configuration\\_management](https://en.wikipedia.org/wiki/Configuration_management)

<https://continuousdelivery.com/foundations/configuration-management/>

<http://www.italia.com/itilconfigmanagement.htm>

<https://github.com/CSC-DevOps/Course/tree/Spring2018>

<https://github.com/CSC-DevOps/Course/tree/Spring2020>

[https://docs.google.com/presentation/d/1PO\\_QTieMkRvW9MDEIMVS0dD5bk50fK5fvSgj5zNyPfw/edit#slide=id.g117c3bc2e1\\_0\\_0](https://docs.google.com/presentation/d/1PO_QTieMkRvW9MDEIMVS0dD5bk50fK5fvSgj5zNyPfw/edit#slide=id.g117c3bc2e1_0_0)

<https://www.html.it/07/01/2019/5-configuration-management-tool-per-automatizzare-i-processi/>