投票 最多可选1项

设置

Are you ready?

A Yes

B No

提交

Review - 1

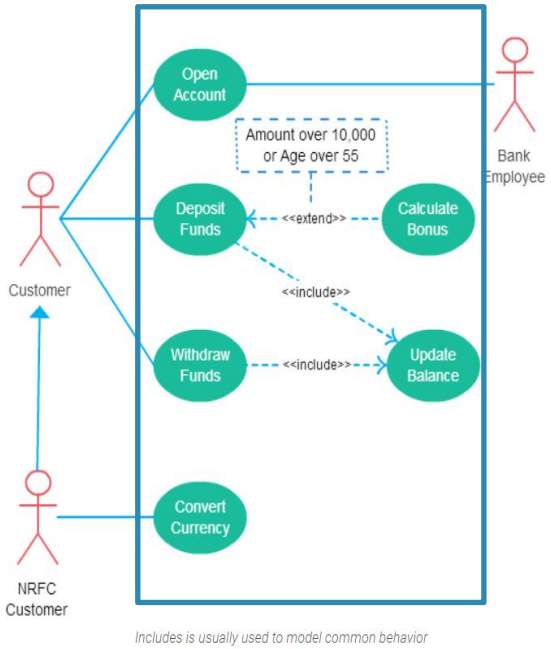
OO testing

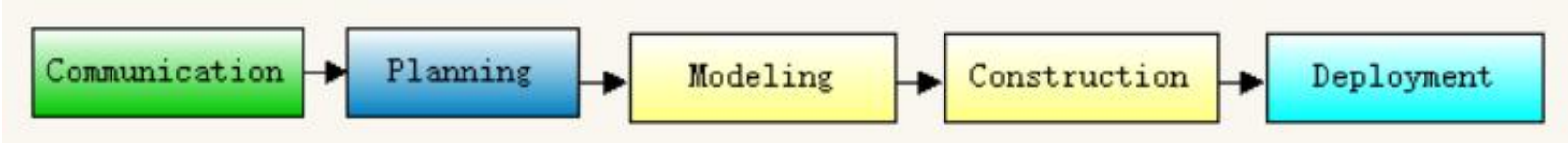
1. Unit testing: Intra-class testing
2. Integration testing[inter-class testing]: thread-based testing, use-based testing,cluster testing
3. Validation testing: use-case in requirement, black-box testing
4. Methods:
   - Partition: State-based partitioning, Attribute-base, Category-based
   - Inheritance: superclass and subclass
     - ✓ if change some method m() in a superclass, we need to retest m() inside all subclass inherit it
     - ✓ if we change a subclass, we need to retest all related methods inherited from its superclass
   - Sequence (Random testing)
   - Behavior (state change)

Some testing tools: Jmeter, Postman

Review - Framework Activities

- Communication
- Planning
- Modeling
  - Analysis of requirements
  - Design
- Construction
  - Code generation
  - Testing
- Deployment





Review - Umbrella Activities

- Software project tracking and control
- Risk management
- Software quality assurance
- Technical reviews
- Measurement
- Software configuration management
- Reusability management

Software Engineering

Part 3
Quality Management

Chapter 29
Software Configuration Management

Let's start our project with the professional model!

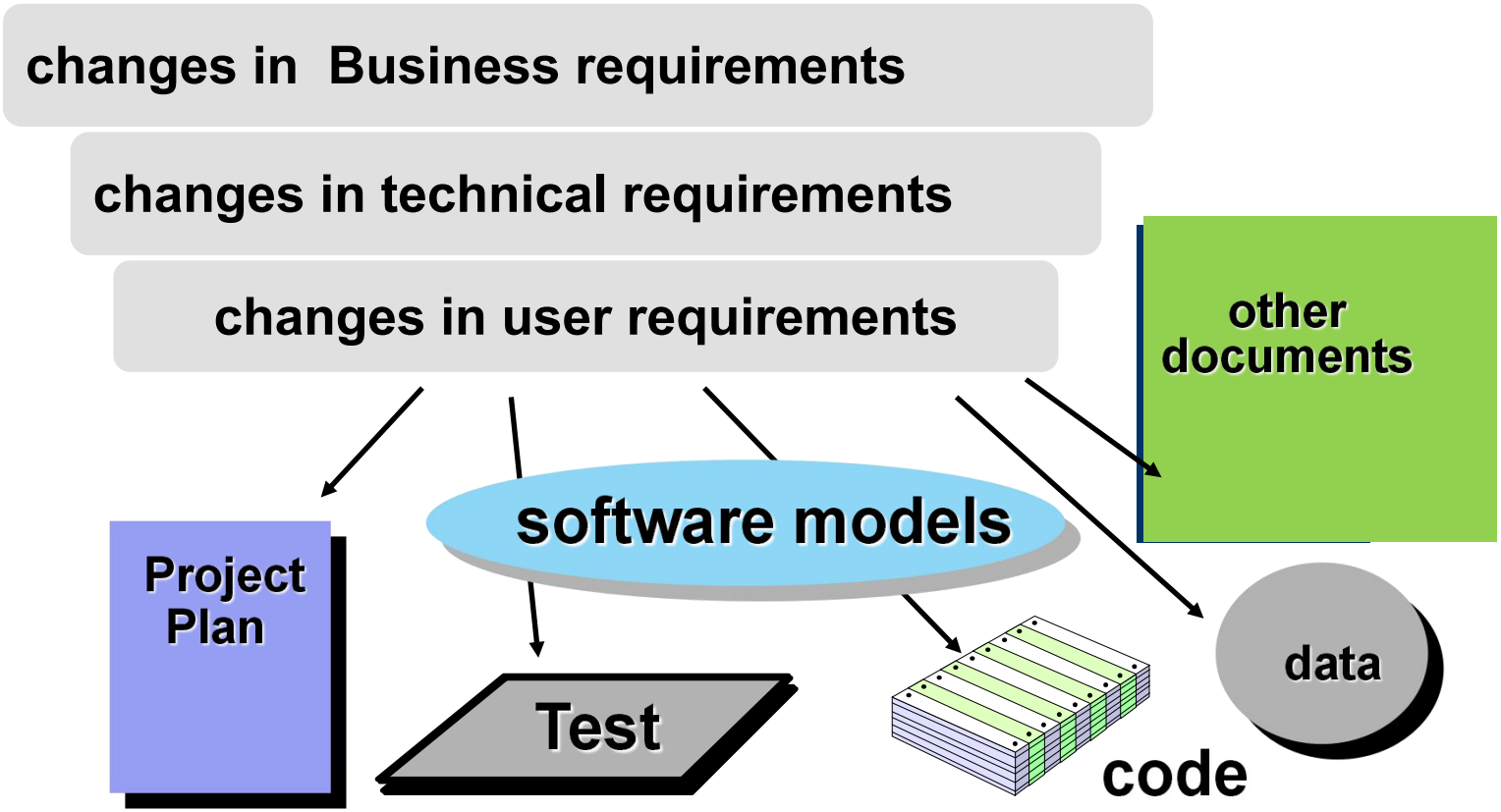
Communication - Plan – Model – Construction - Deployment

Contents

29.1 The “First Law”

No matter where you are in the system life cycle, the system will change, and the desire to change it will persist throughout the life cycle.

29.1.1 What Are These Changes?



主观题 10分

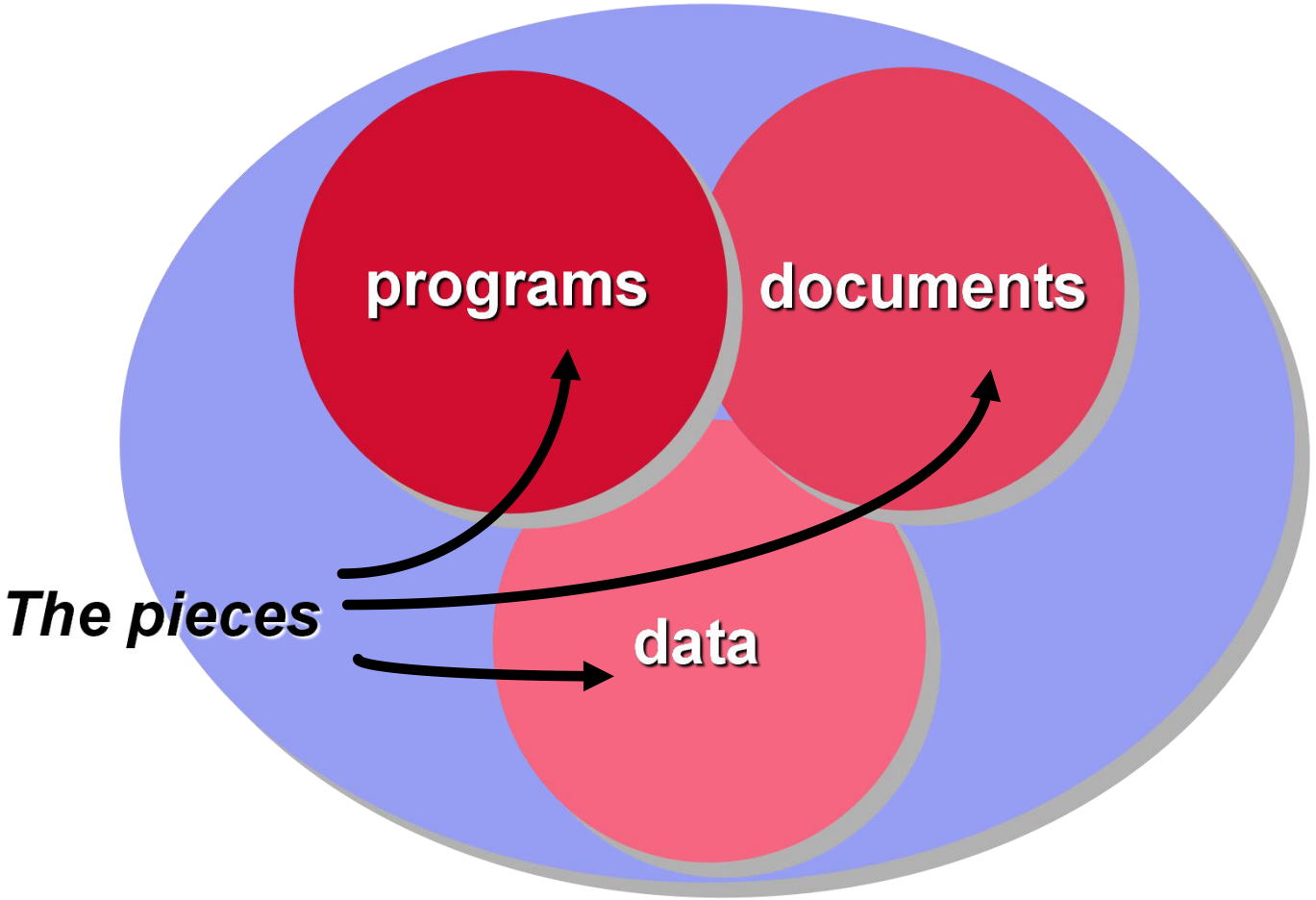
设置

How to do the backup when you modify your documents or your code? Have you used some tools for your version management?

正常使用主观题需2.0以上版本雨课堂

作答

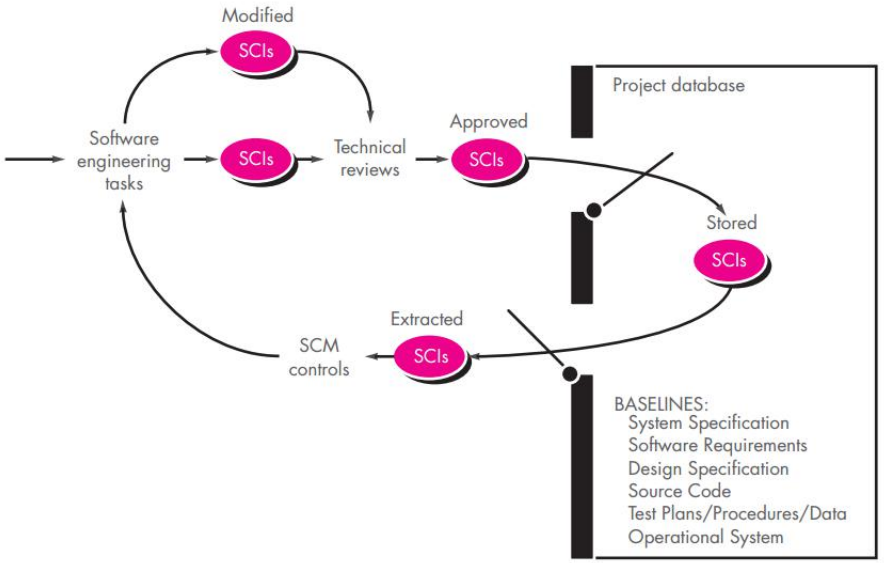
29.1.2 Elements of a Software Configuration Management System



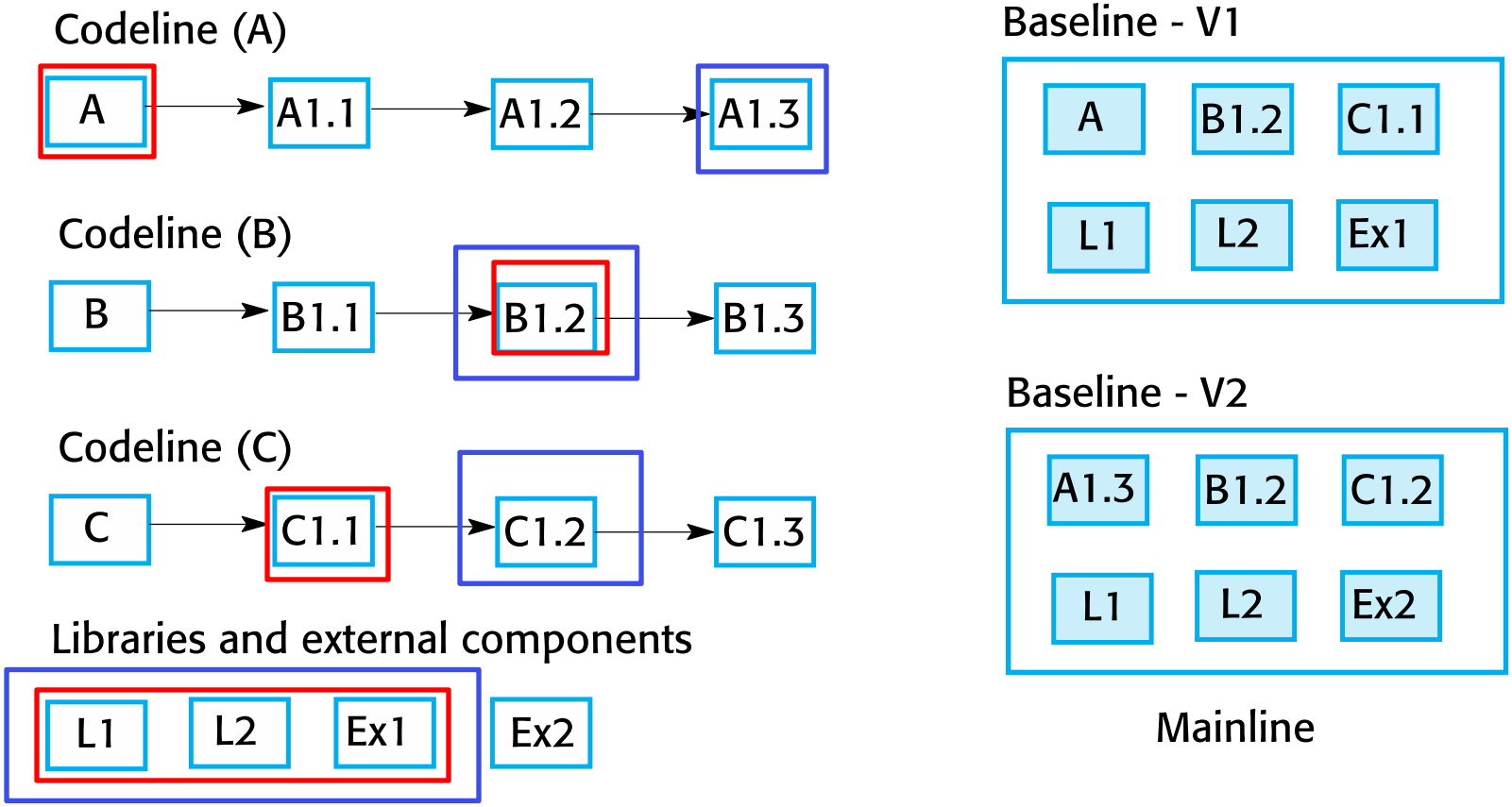
29.1.3 Baselines

- Baseline (IEEE Std. No. 610.12-1990) :

  A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures.



29.1.3 Baselines



Cite from Chapter 25 Configuration management in Software Engineering (10th Edition)

多选题 1分

设置

此题未设置答案，请点击右侧设置按钮

Which ones will be managed in SCM?

A Design Specification

B Data Model

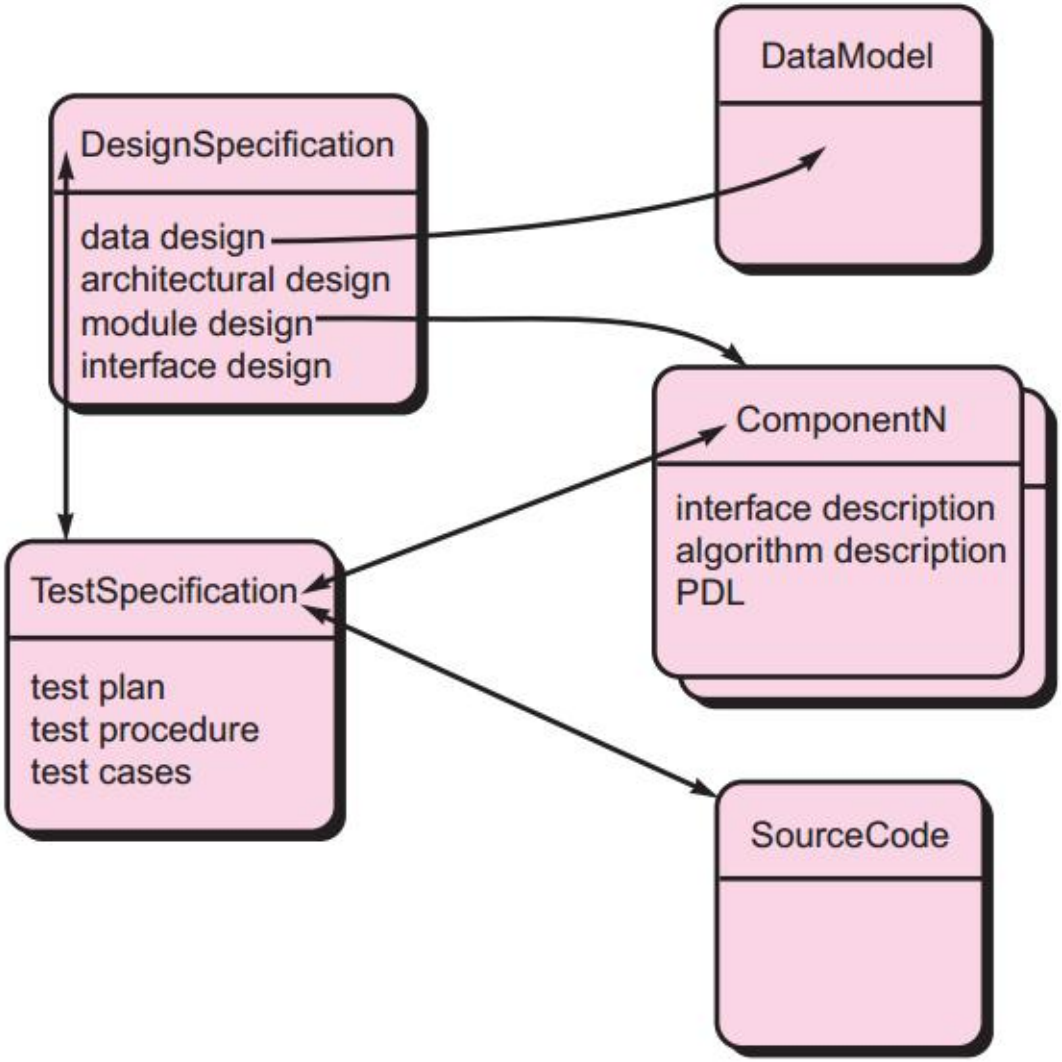
C Component

D Source code

E Test specification

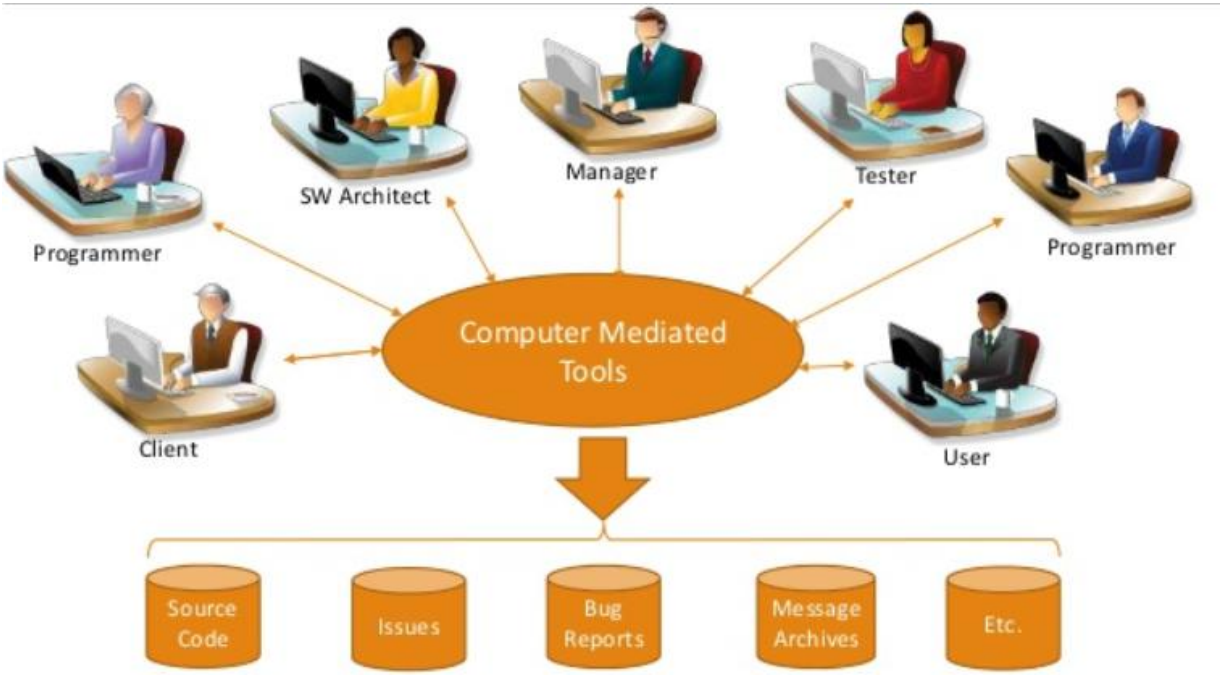
提交

29.1.4 Software Configuration Objects



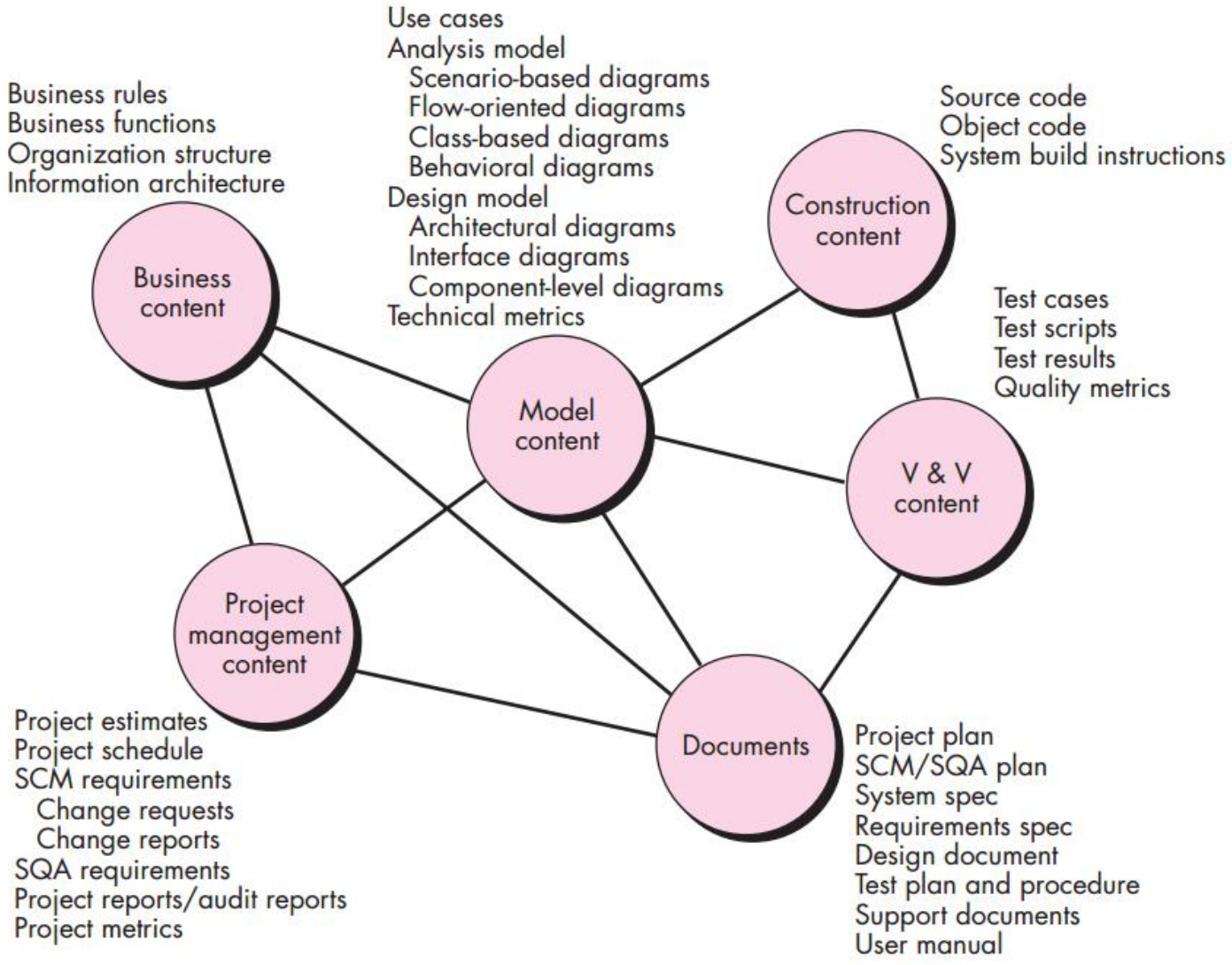
29.2 SCM Repository

- The SCM **repository** is the set of mechanisms and data structures that allow a software team to manage change in an effective manner.



Current and historical artifacts and interactions are registered in software repositories

29.2 SCM Repository



29.2 SCM activity

- **Version management**
  - Keeping track of the multiple versions of system components and ensuring that changes made to components by different developers do not interfere with each other.
- **System building**
  - The process of assembling program components, data and libraries, then compiling these to create an executable system.
- **Change management**
  - Keeping track of requests for changes to the software from customers and developers, working out the costs and impact of changes, and deciding the changes should be implemented.
- **Release management**
  - Preparing software for external release and keeping track of the system versions that have been released for customer use.

29.2 SCM Co-Ordinator (activity)

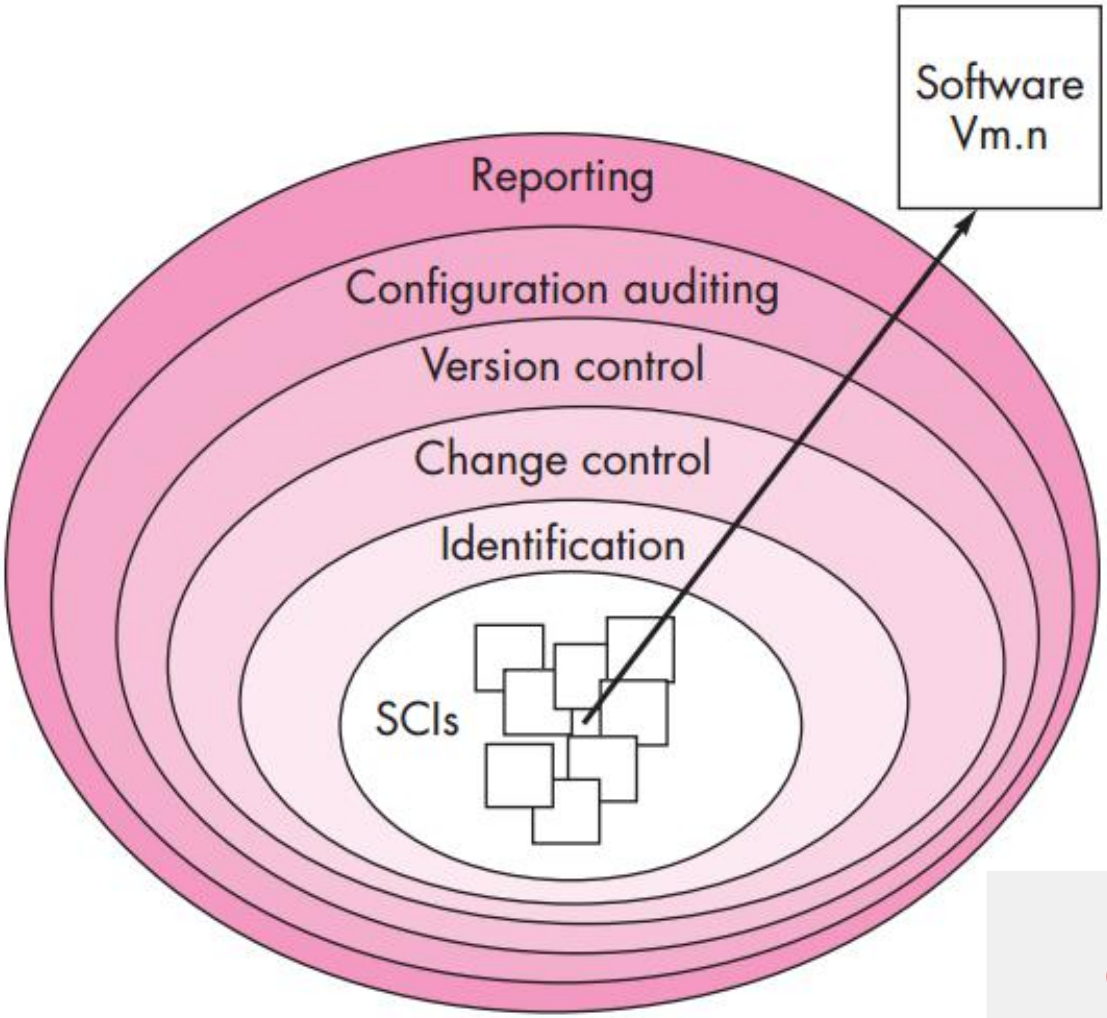
Let's watch a video to learn more about SCM.

https://www.youtube.com/watch?v=AaHaLjuzUm8

Take a break

Five minutes

29.3 The SCM Process



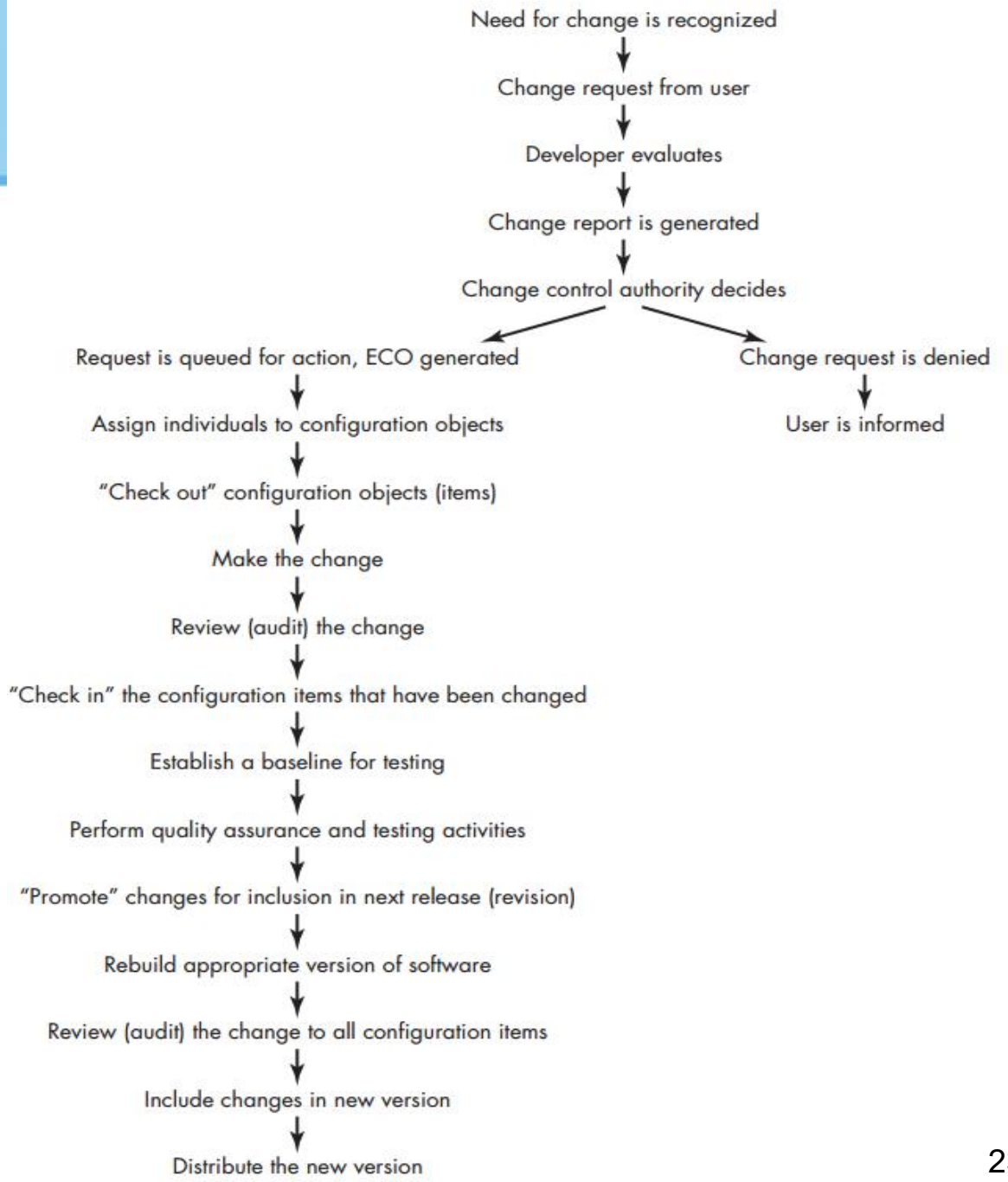
SCI：Software Configuration Item

29.3.2 Version Control

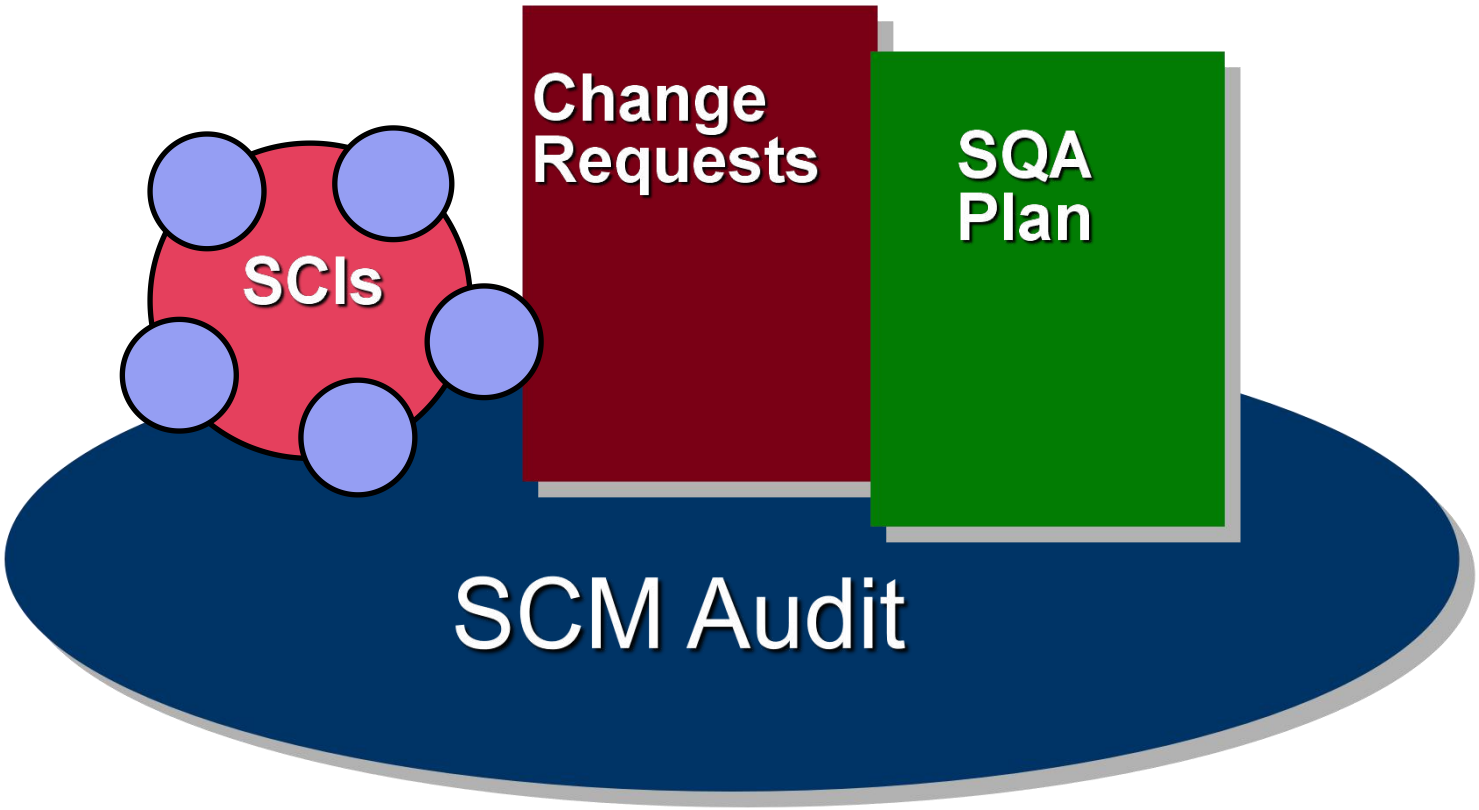
- Version control combines procedures and tools to manage different versions of configuration objects that are created during the software process.

- A version control system :
  - a *project database (repository)* that stores all relevant configuration objects
  - a *version management* capability that stores all versions of a configuration object (or enables any version to be constructed using differences from past versions);
  - a *make facility* that enables the software engineer to collect all relevant configuration objects and construct a specific version of the software.
  - an *issues tracking* (also called *bug tracking*) capability that enables the team to record and track the status of all outstanding issues associated with each configuration object.

29.3.3 Change Control Process

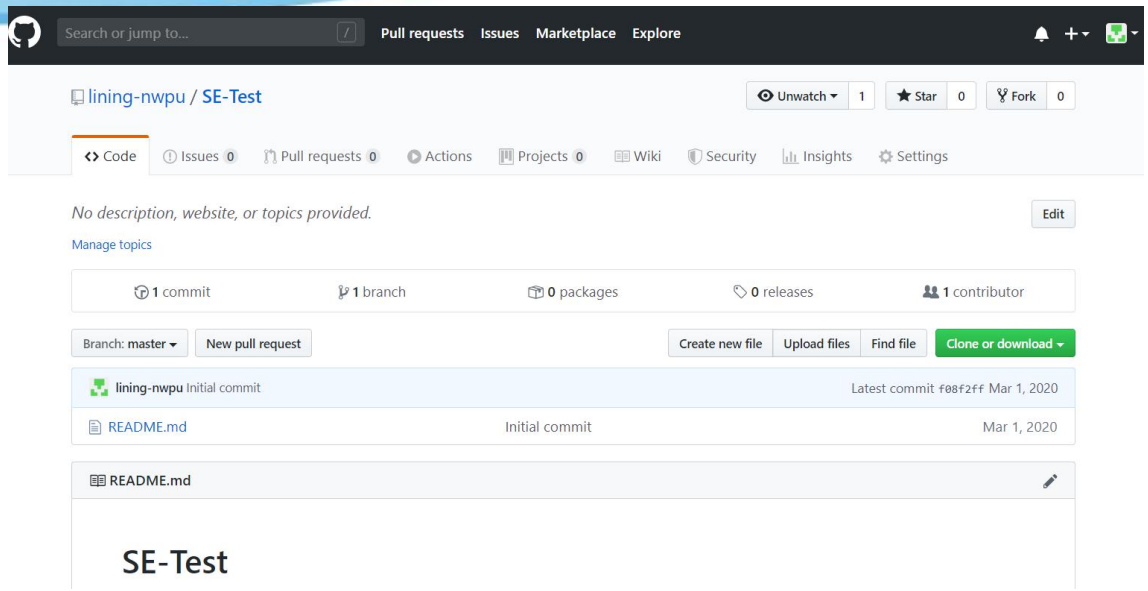


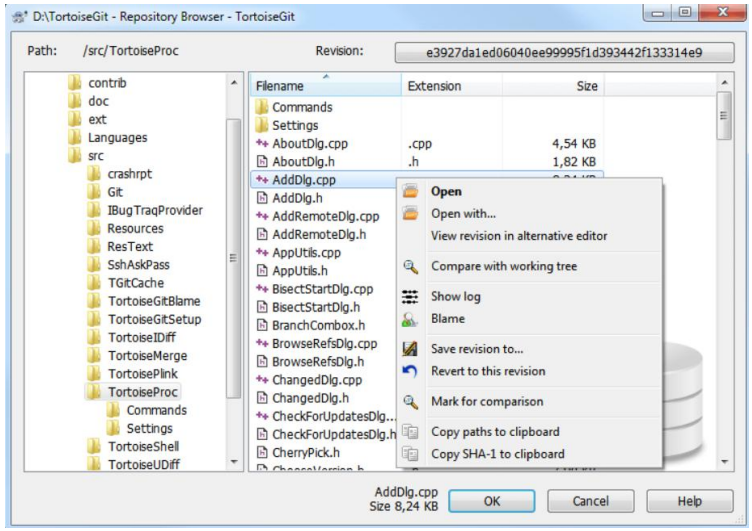
29.3.4 Auditing



29.4 SCM Tool

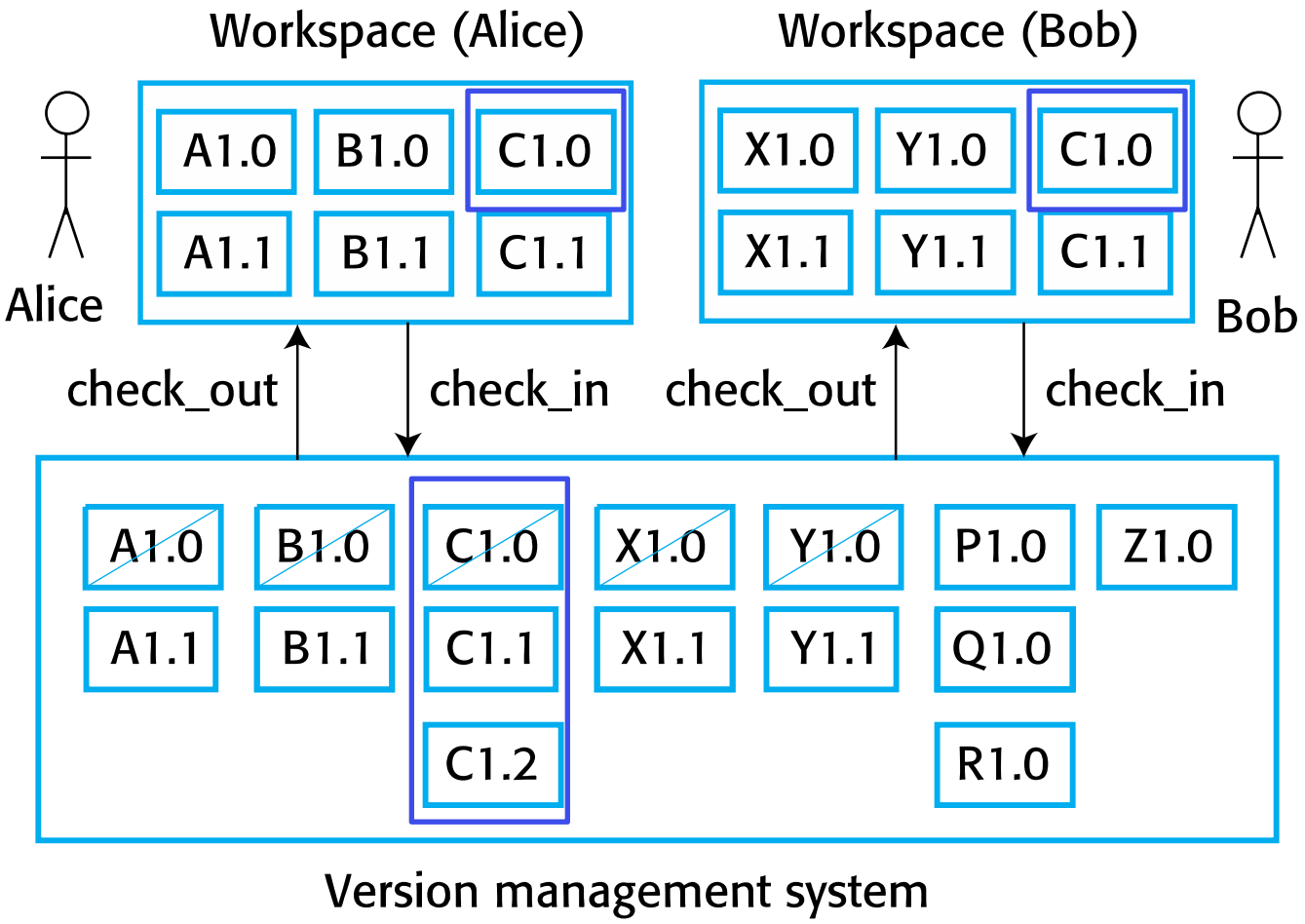
- Github
- SVN
- VSS
- ....





29.4 Repository Check-in/Check-out

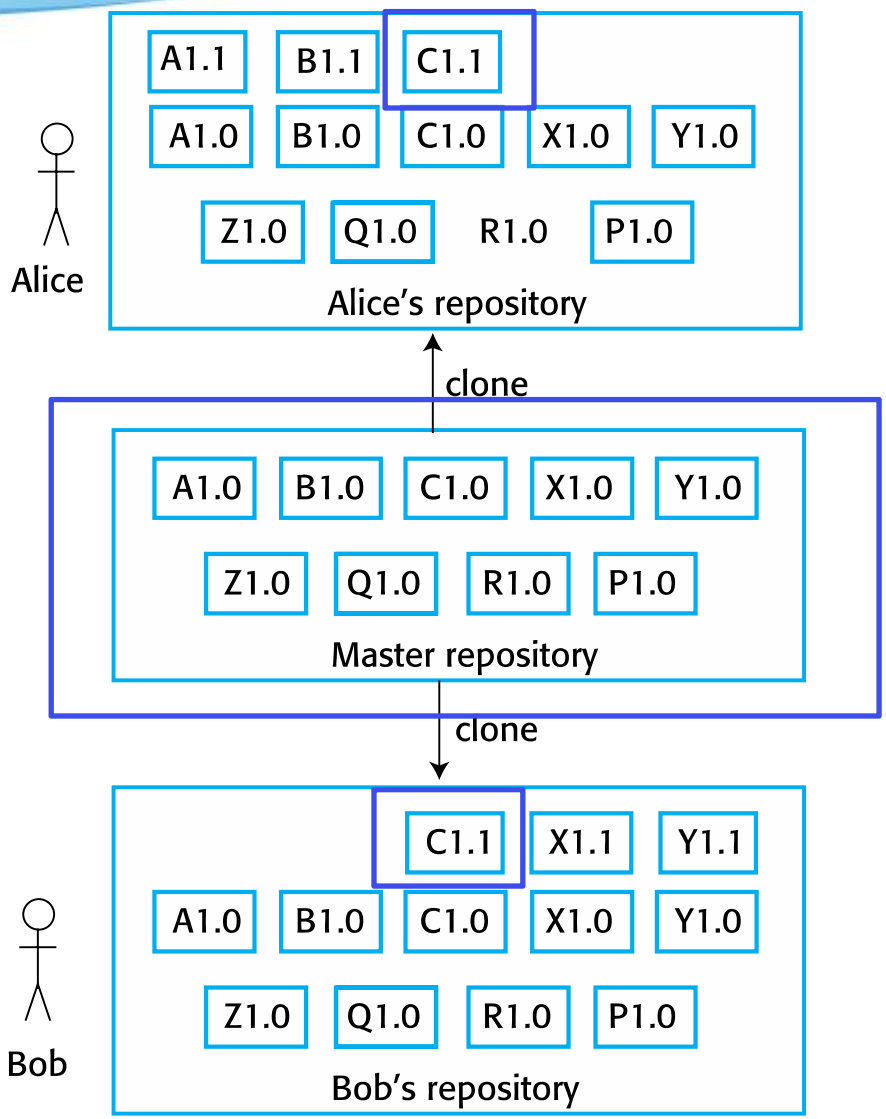
Centralized version control



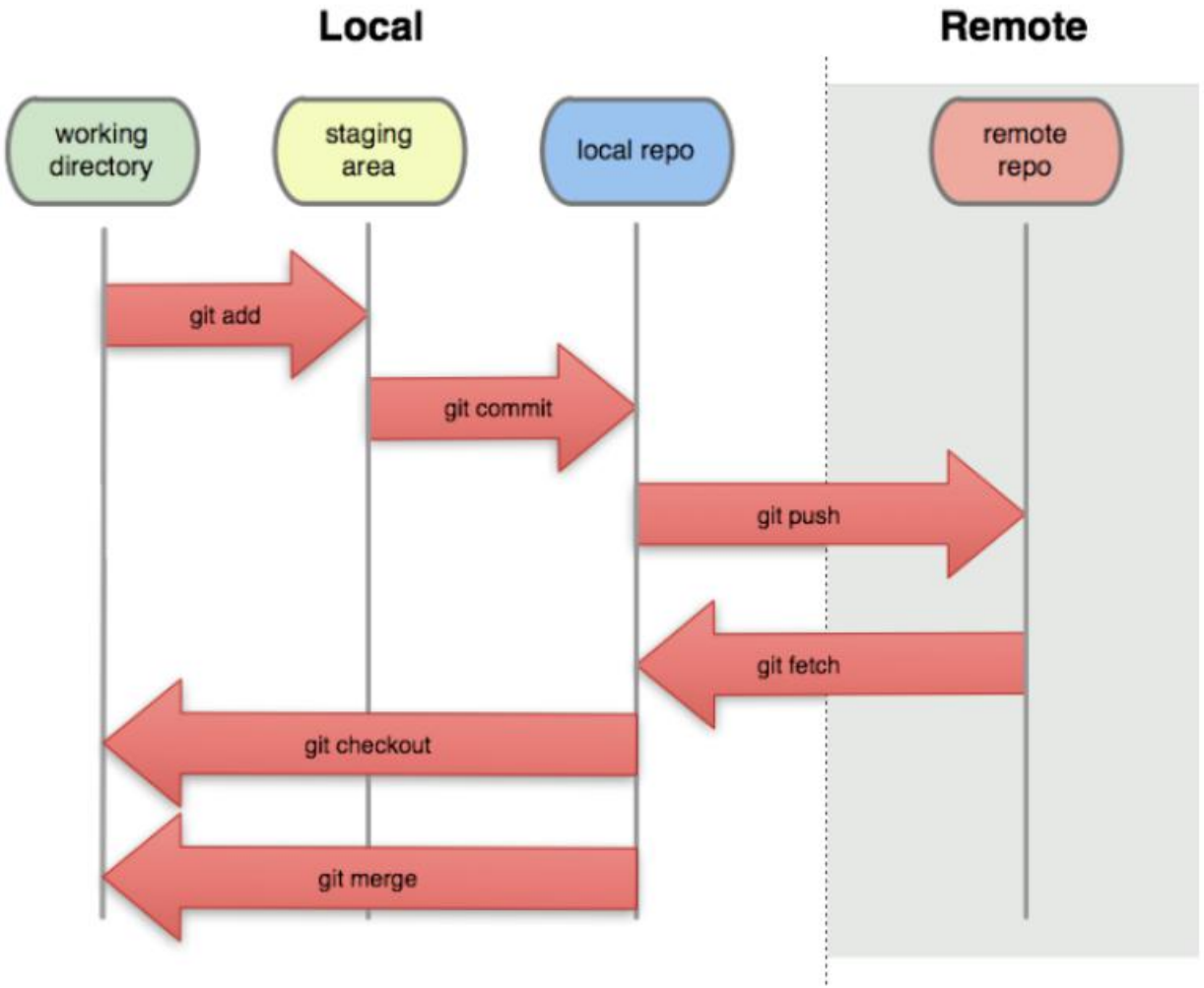
29.4 Repository Check-in/Check-out

Distributed version control

Repository cloning



29.4 SCM Tool - Github



From: https://stackoverflow.com/questions/13072111/gits-local-repository-and-remote-repository-confusing-concepts

29.4 SCM Tool

1. Let's try to use GitHub!
2. Let's watch a video to learn more git and GitHub.

: Introduction - Git and GitHub for Poets

https://www.bilibili.com/video/av46599796?p=2

29.4 SCM Tool

GitHub vs Bitbucket vs GitLab: What are the differences?

- They are code collaboration and version control tools offering repository management.
- GitLab is open source, though all three support open source projects. GitHub offers free public repositories; Bitbucket also offers free private repositories; GitLab offers a Community Edition which is entirely free.

Cloud-Hosted Pricing Comparison

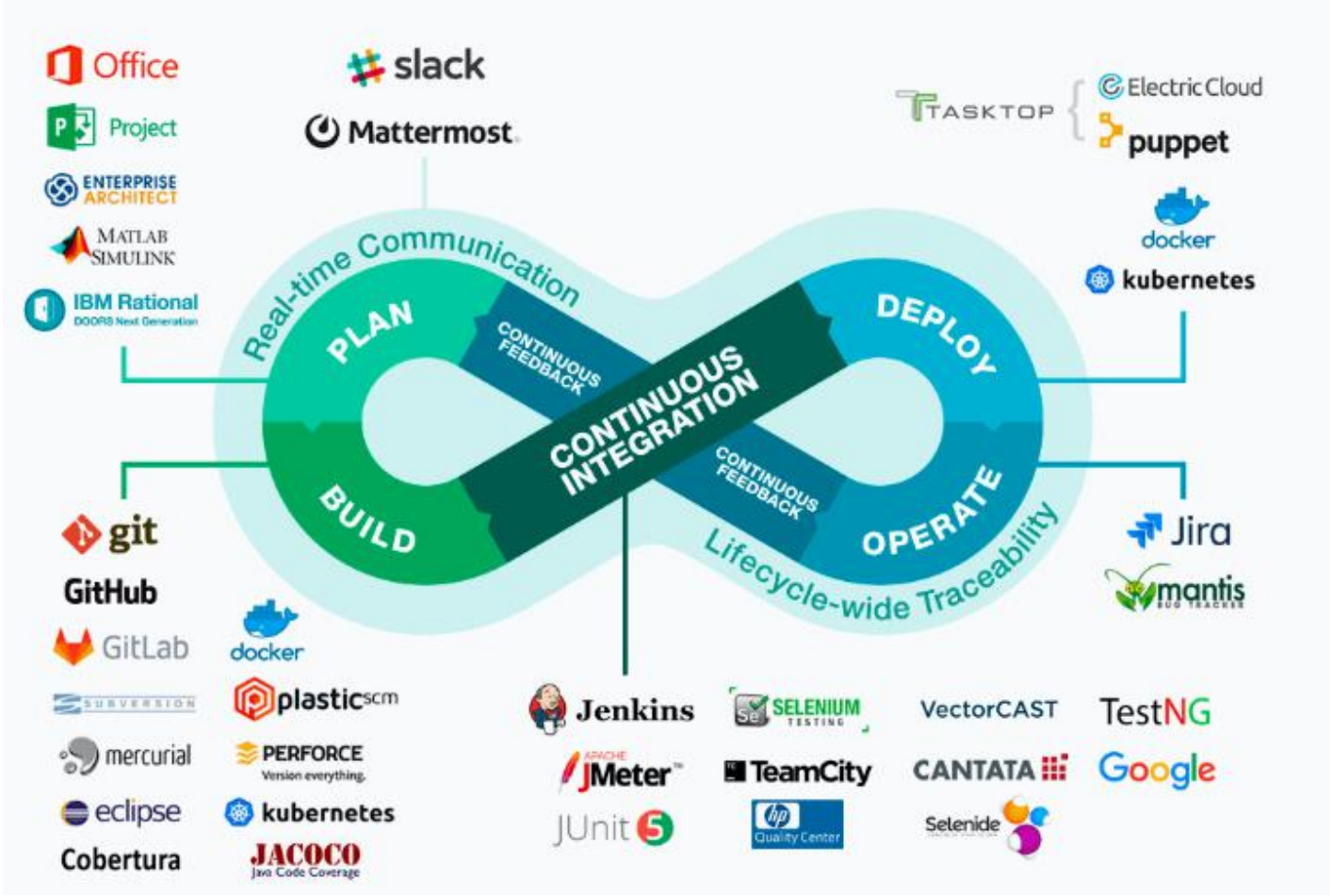
If your organization is able to use the lowest-tier cloud-hosted version of each platform, your costs will typically grow as a function of team size and whether or not the code you're hosting is open source.

		Github	Bitbucket	GitLab
Open Source	Up to 5 users	$0	$0	$0
	> 5 users	$0	$2/user/mo	$0
Private Repos	Up to 5 users	$25/mo	$0	$0
	> 5 users	$9/user/mo	$2/user/mo	$0

**Note: Github also includes a $7/month plan for individuals who want private repositories.*

Cited from https://stackshare.io/stackups/bitbucket-vs-github-vs-gitlab / https://peakd.com/git/@urisan/difference-between-git-vs-github-vs-gitlab

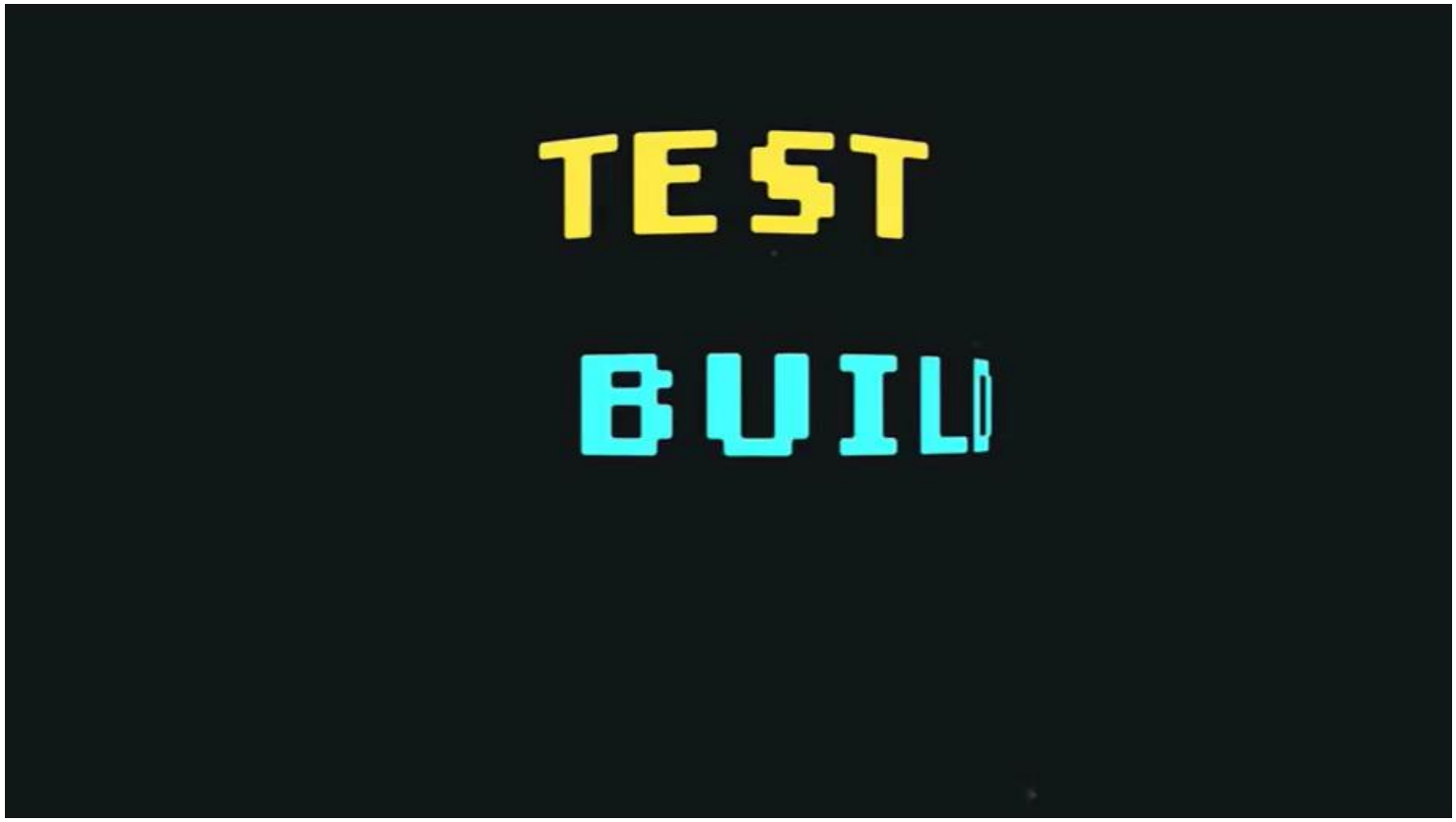
29.4 SCM Tool



Cited from : https://intland.com/codebeamer/devops-it-operations/

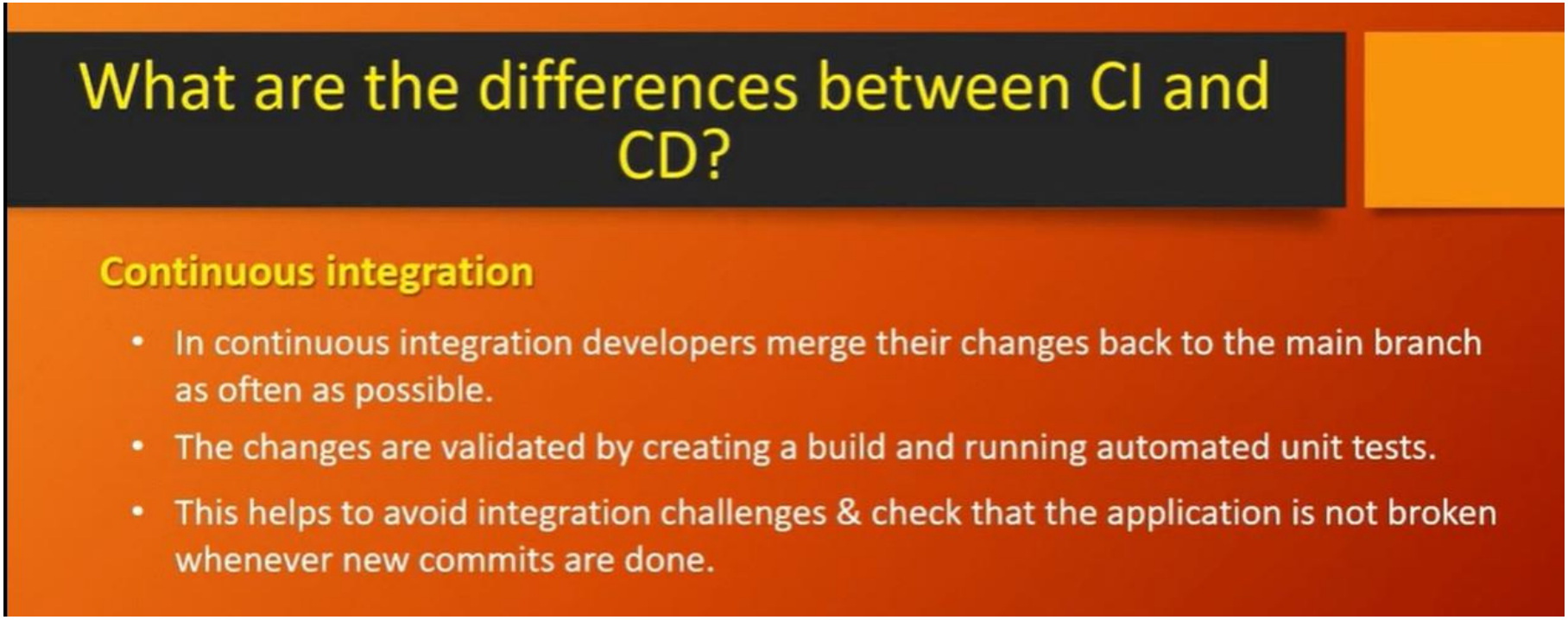
29.4 SCM Tool

Let's watch a video to learn more about DevOps.



Cited from : https://intland.com/codebeamer/devops-it-operations/

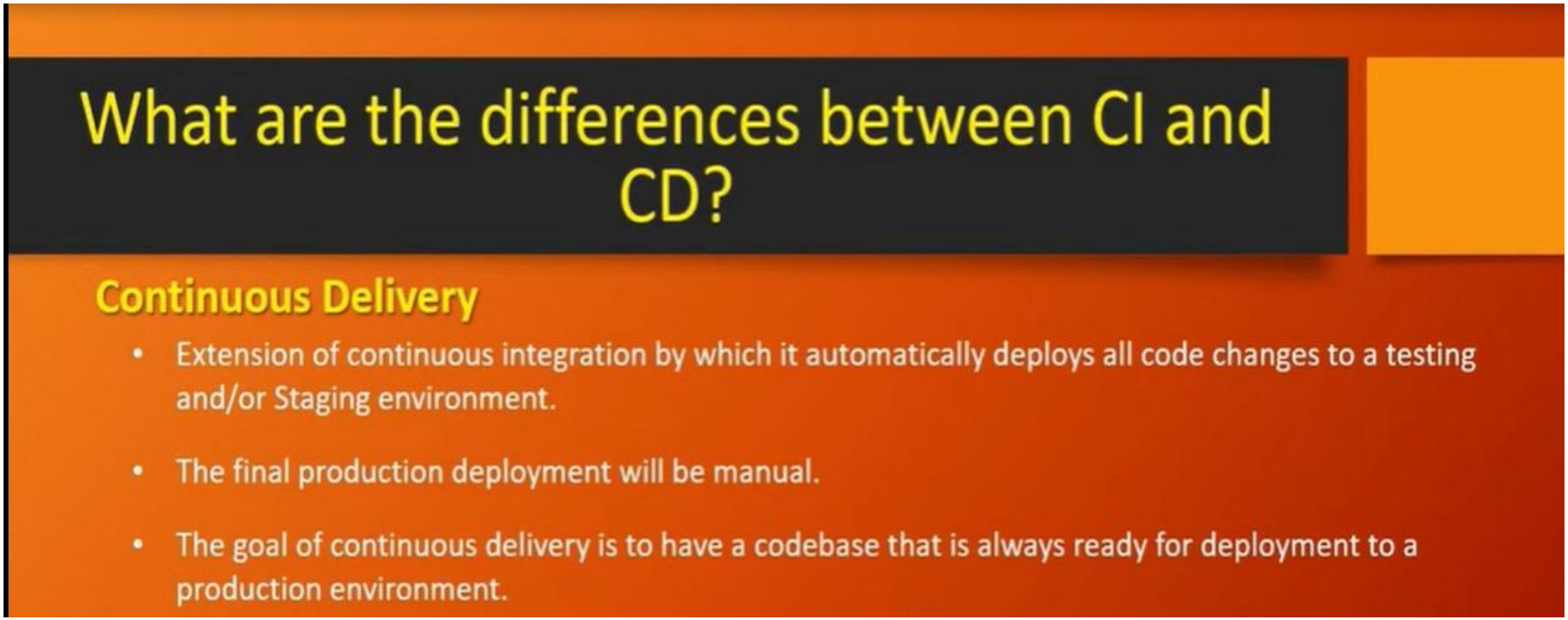
29.4 SCM Tool



Cited from : https://www.youtube.com/watch?v=jUiKi6FWYrg

Ref: https://docs.gitlab.com/ee/development/cicd/

29.4 SCM Tool



Cited from : https://www.youtube.com/watch?v=jUiKi6FWYrg

29.4 SCM Tool

What are the differences between CI and CD?

Continuous Delivery

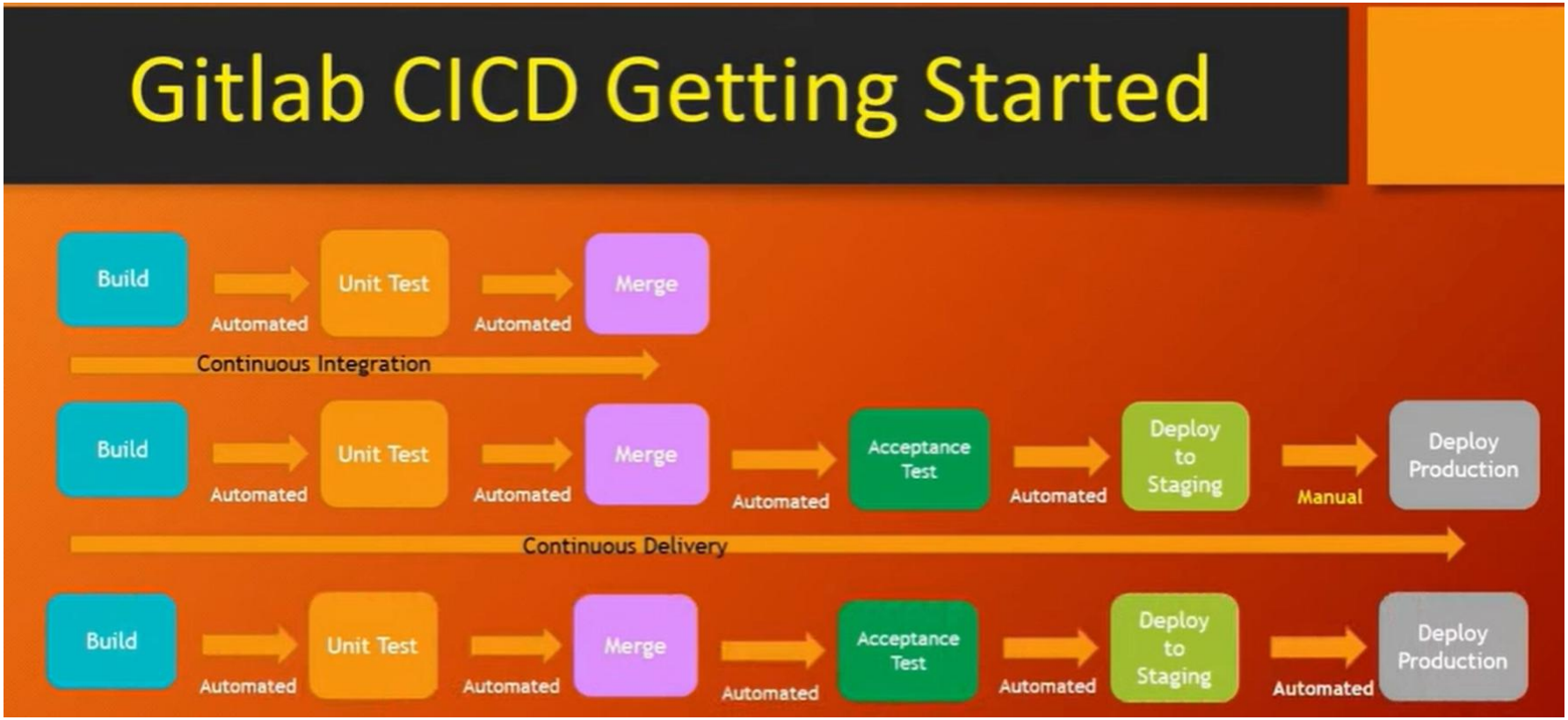
- Extension of continuous integration by which it automatically deploys all code changes to a testing and/or Staging environment.
- The final production deployment will be manual.
- The goal of continuous delivery is to have a codebase that is always ready for deployment to a production environment.

Continuous Deployment

- Every change that passes all stages of your production pipeline is released to your customers/Production environment.
- No human intervention, completely automated.

Cited from : https://www.youtube.com/watch?v=jUiKi6FWYrg

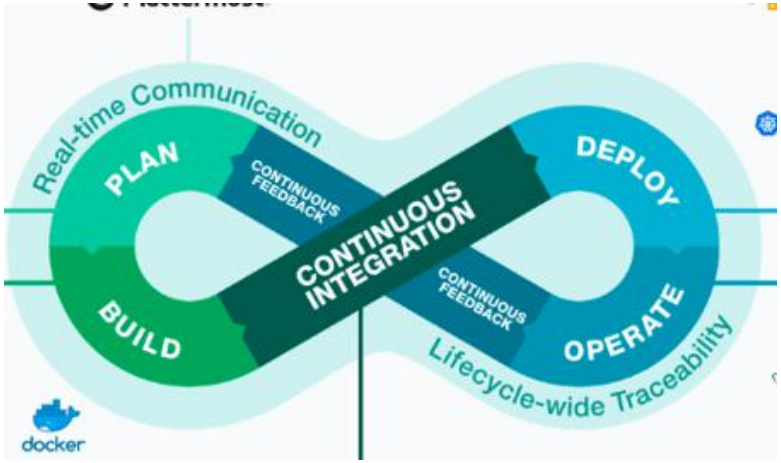
29.4 SCM Tool



Cited from : https://www.youtube.com/watch?v=jUiKi6FWYrg

Summary

- **Software Configuration Objects**: docs, code, test, design models, etc
- **Version control**
  - Centralized: SVN,
  - Distributed: **git** (github, gitlab,...)
- **DevOps: CI & CD**



29.4 SCM Tool

Practice:

Try to use Github or Gitlab!

THE END