

CZ3002 - Advanced Software Engineering

Release Management

Faculty : Dr Althea Liang
School : School of Computer Science and Engineering
Email : qhliang@ntu.edu.sg
Office : N4-02c-107

Learning Objectives

At the end of the lesson, you should be able to:

- ▶ Explain activities of software release management
- ▶ Describe how the version control system works and use it
- ▶ Describe important practices of release management



Where Is Release Management Positioned

Examples of Sub-Disciplines of Software Engineering:

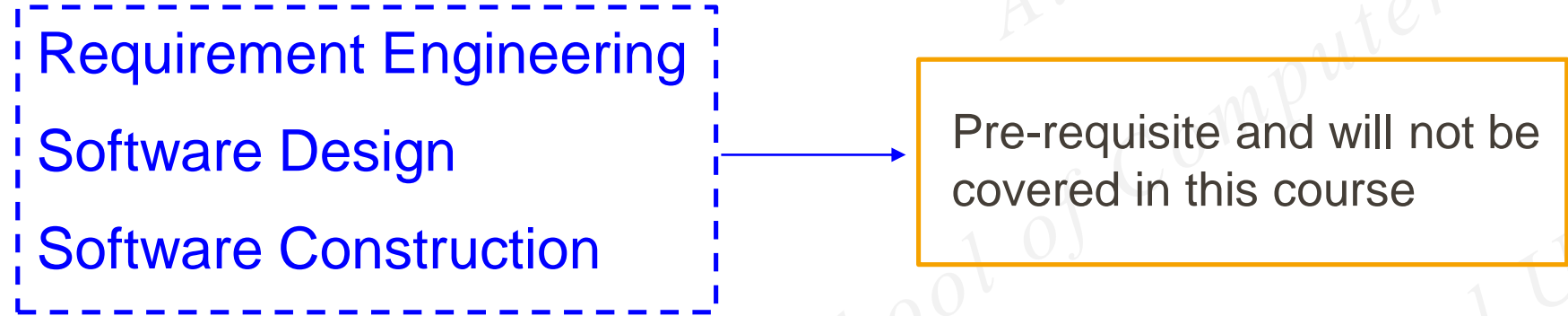
- ▶ Software Quality Management
- ▶ Software Engineering Management
- ▶ Software Configuration Management

Althea LIANG
School of Computer Science
& Engineering,
Nanyang Technological University

Where Is Release Management Positioned

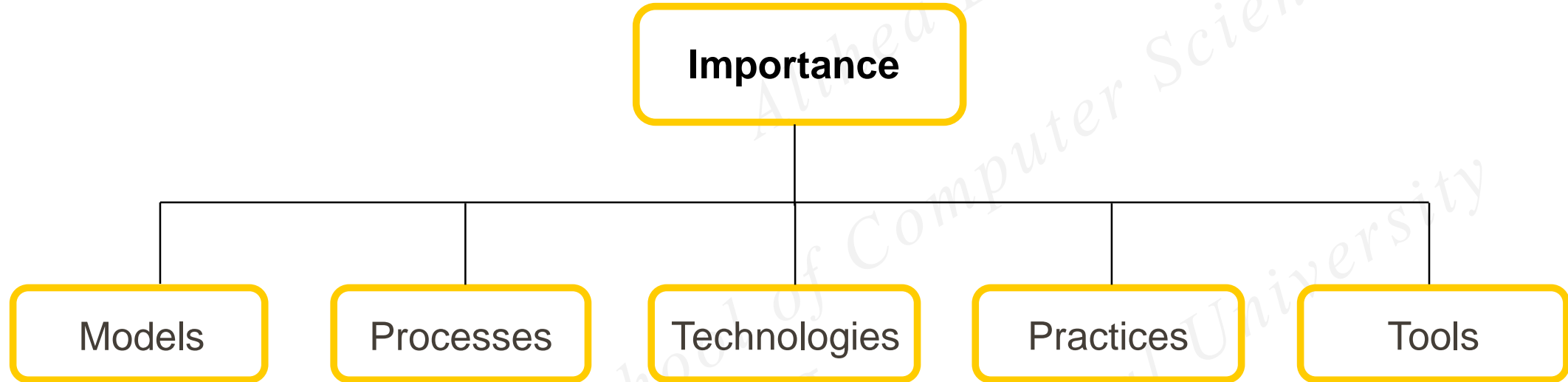
Examples of Sub-Disciplines of Software Engineering:

- ▶ Software Maintenance
- ▶ Software Testing
- ▶ Requirement Engineering
- ▶ Software Design
- ▶ Software Construction



Pre-requisite and will not be covered in this course

What Are Important in Release Management



Release Management

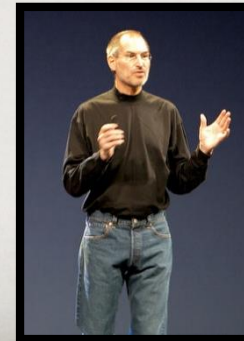
- Release management, a functional area of software configuration management.

THE NEWS

To address the security issue in iPhone, Apple Inc is going to release iOS 10.3.2 on 15 May 2017, the enhanced version.

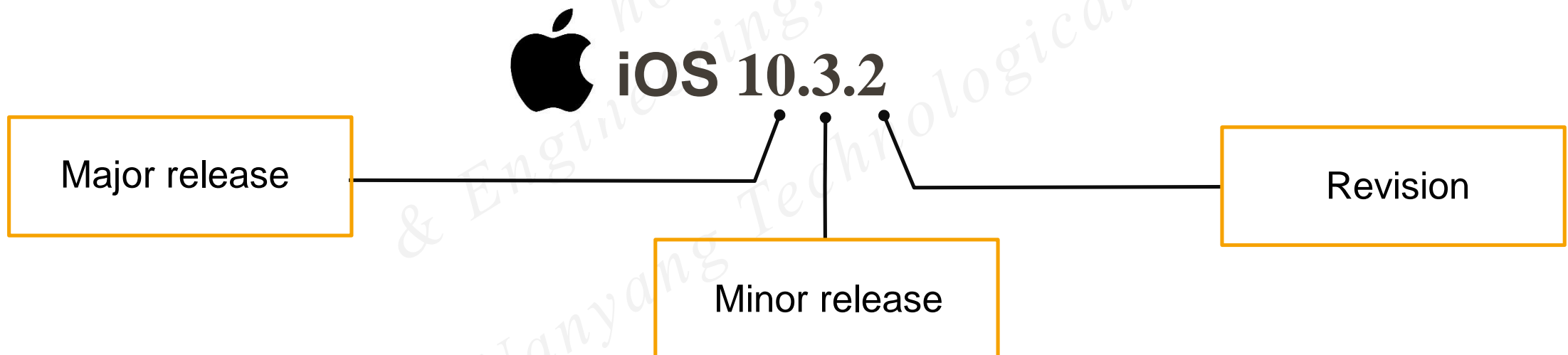
What is a release?

What do these numbers represent?



► A Release

- ❖ a **tested** and **approved baseline** that is usually installed at a **client site** or **packaged for purchase**
- ❖ has a **unique ID**
- ❖ e.g. iOS 10.3.2, Android 6.0.1, Windows 10, version 1607

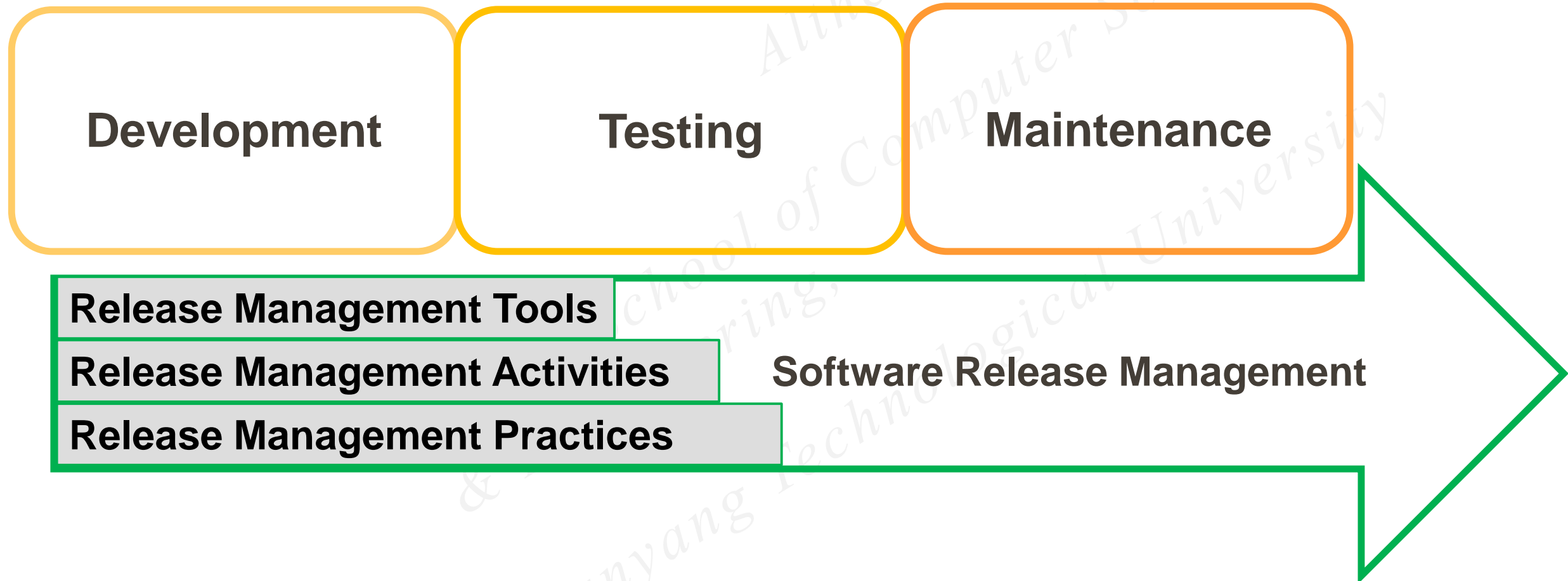


► A Patch

- ❖ is a “minor release” - not a whole **build** but some subset of a build
- ❖ generally done to fix one or more significant bugs
- ❖ has a unique ID (e.g. “Release 2.1-1” or “Patch 2.1-1”)

Software Release Management – Roadmap

Suggested Roadmap of Systematic Software Release Management

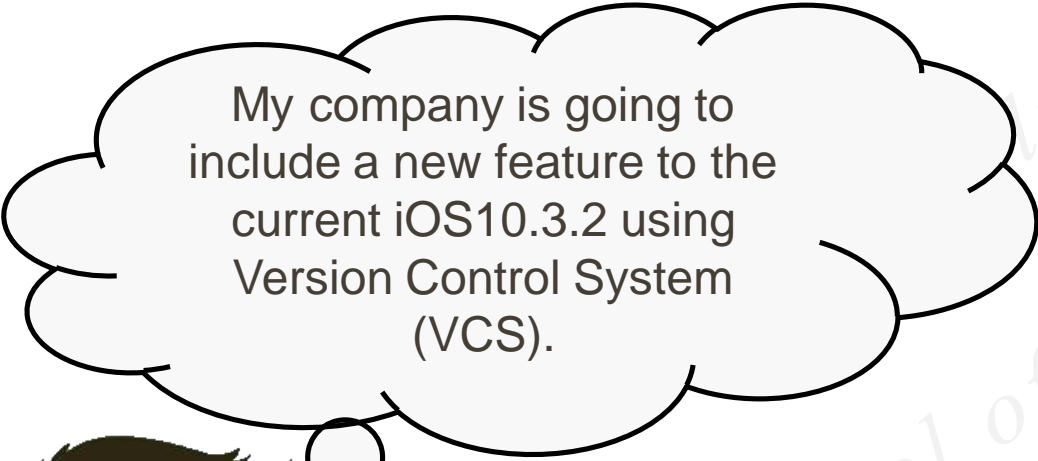


Tools Used for Release Management

- ▶ **Version Control Software (or Version Control Systems)** can be used to manage releases
- ▶ A good way to produce releases is through using various branches meant for release in VCS (version management tools)
- ▶ More or less in the context of VCS
- ▶ Examples of VCS: Mercurial, Git, Subversion

VCS Graph

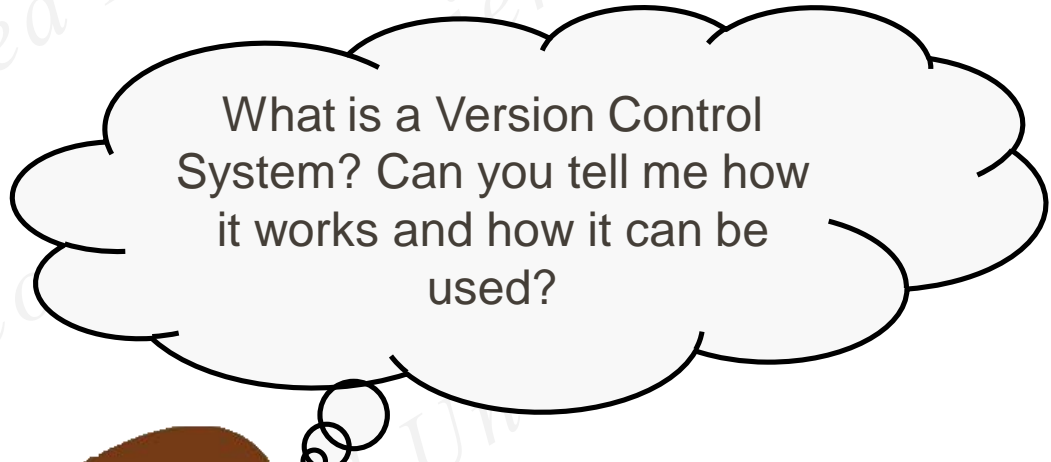
Below is the conversation between William, a senior software engineer from Apple Inc and Marcus, a newbie who is enthusiastic about software engineering.



My company is going to include a new feature to the current iOS10.3.2 using Version Control System (VCS).



William

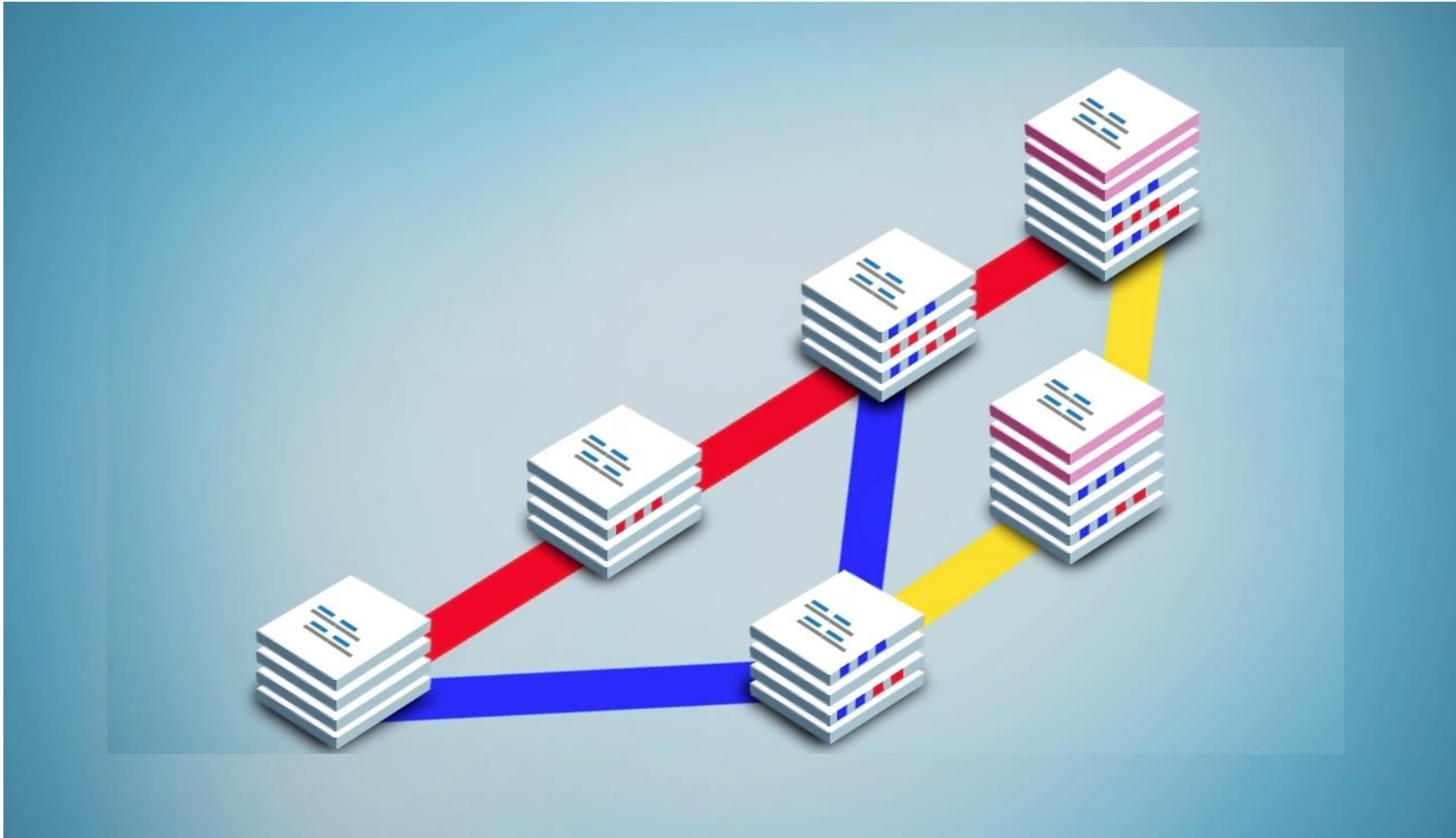


What is a Version Control System? Can you tell me how it works and how it can be used?



Marcus

VCS Graph

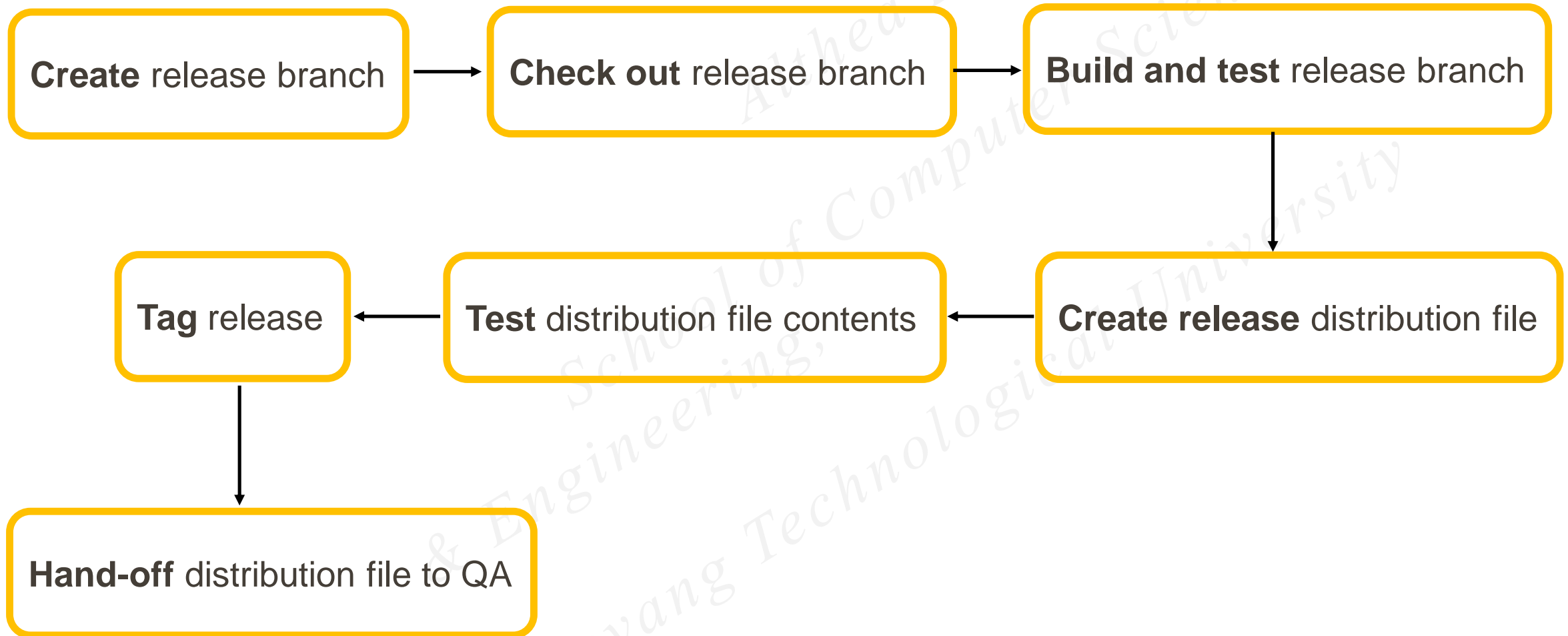


Branch Dynamics in VCS

- ▶ There is a main branch or **trunk**.
- ▶ Developers **commit** all new work to the **trunk**.
- ▶ Developers create a **release branch**.
- ▶ The trunk is **copied** to a “**release**” **branch** and teams continue to work in **parallel**.
- ▶ The branch is **tagged and released**.
- ▶ The branch is **maintained** over time.

Release Preparation in VCS

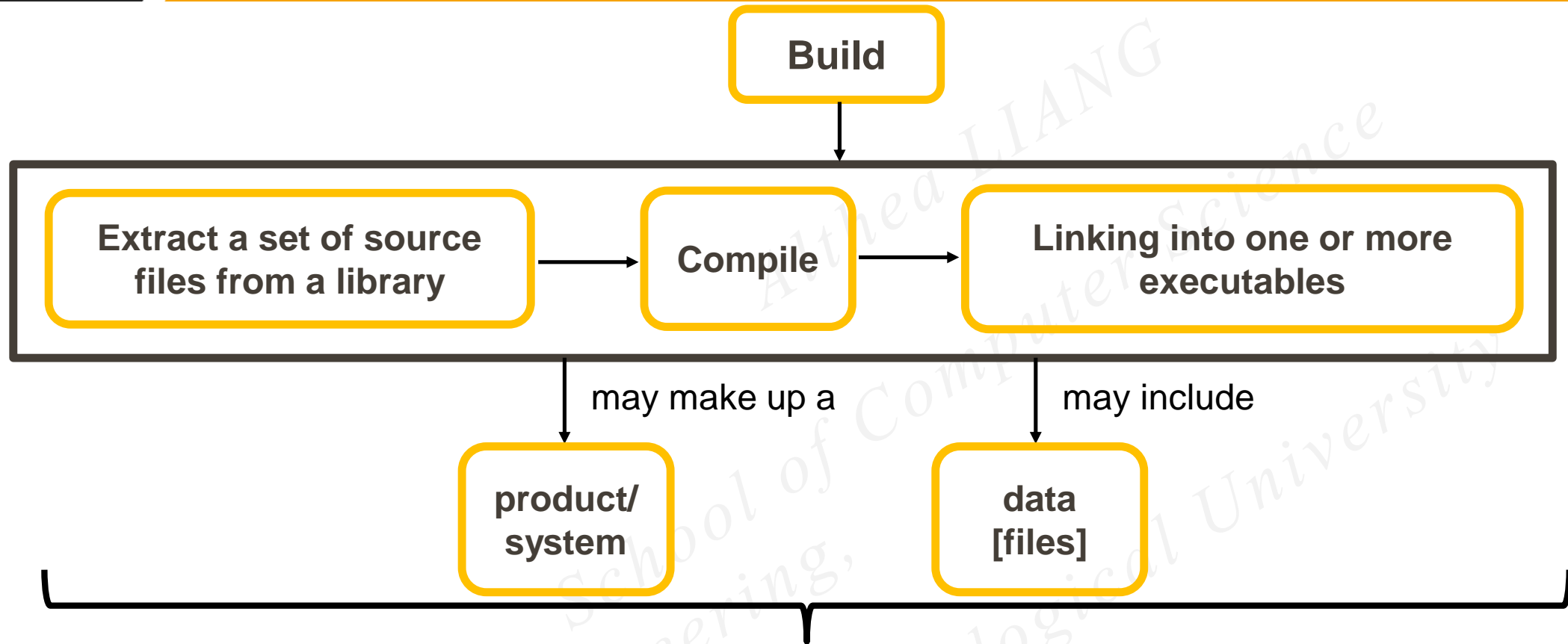
Through Branches:



Software Release Management Activities

- ▶ Distribution of software configuration item **outside** development activity.
- ▶ Encompasses the **identification**, **packaging** and **delivery** of the elements of the product release, e.g. executable program, documentation, release notes, configuration data.
- ▶ Release management must consists of processes for **Internal (test team)** or **External (customer) releases**.
 - ❖ Internal releases a.k.a. promotions
- ▶ Release management process must be designed to control a release that may consist of **variants** for different platforms.

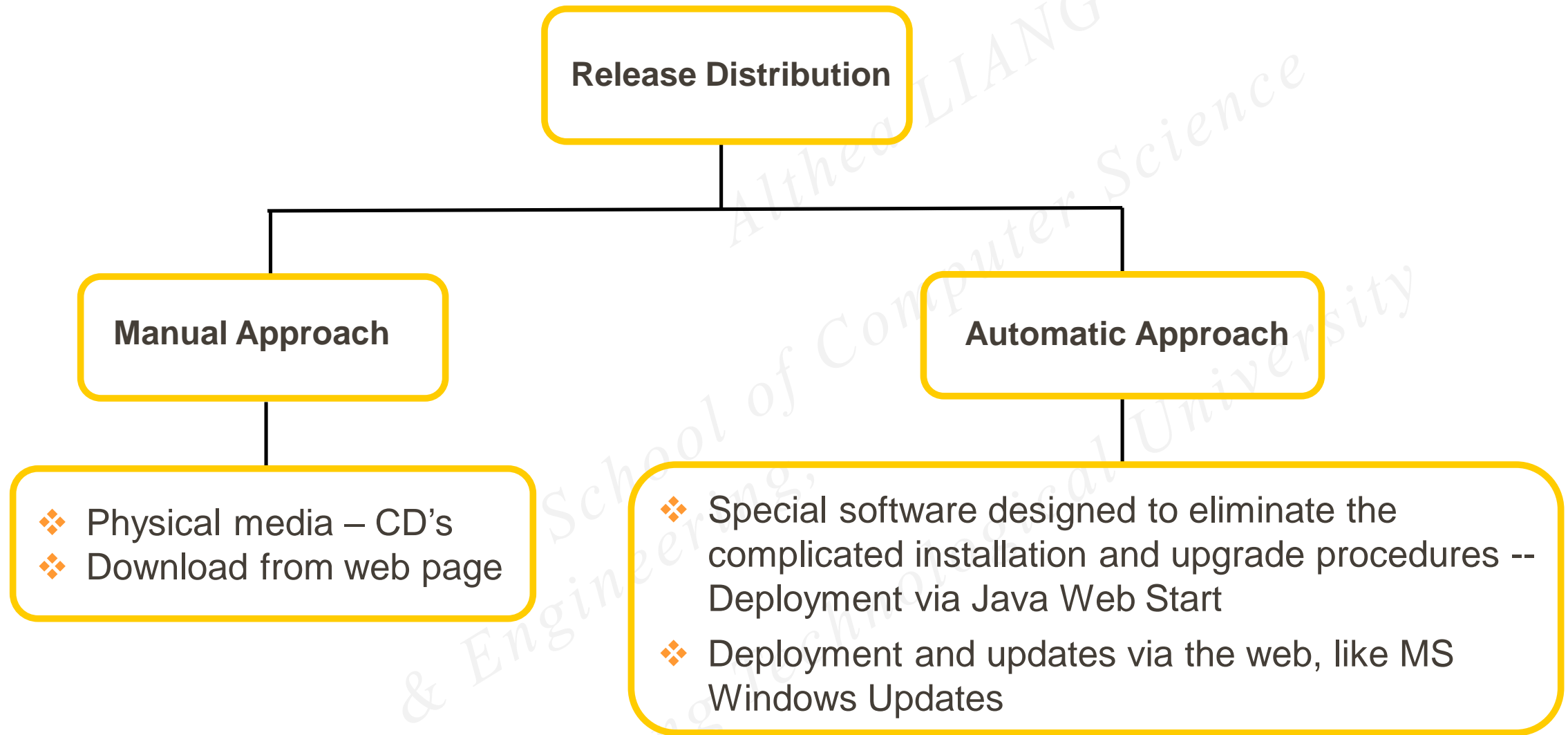
Builds and Software Building



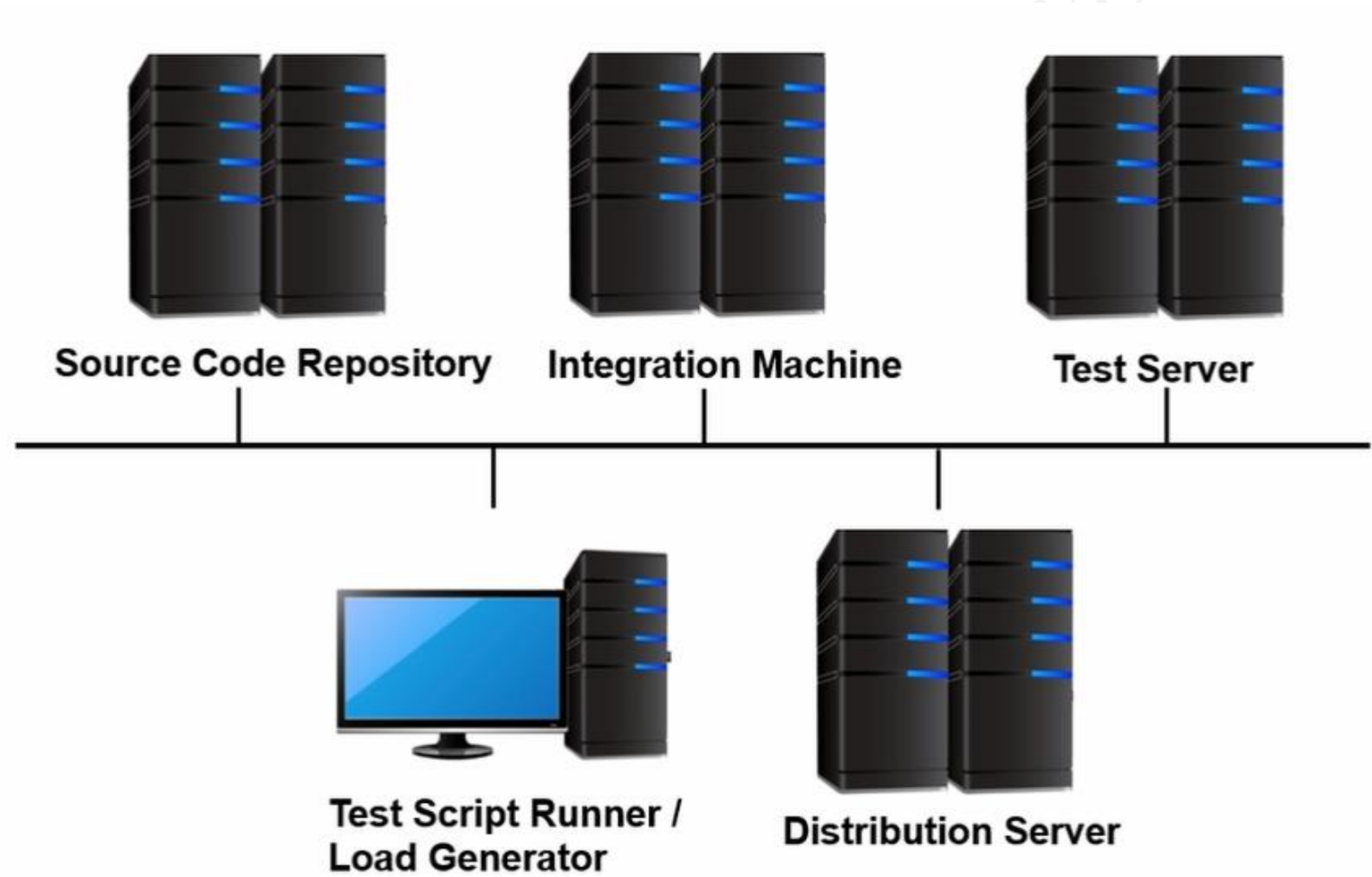
- ❖ Given a unique, often sequential ID (e.g. Build 123,124,125,...)
- ❖ Done on some regular basis (overnight, once a week)
- ❖ Comprises of a build list which identifies all files and which revision of each to include in a build
- ❖ Correspond to a node in VCS graph

- ▶ Activity of **combining correct versions** of software configuration items, using the **appropriate data**, into an executable programme for delivery usually to customer or test team
 - ❖ Complete – self-sufficient
 - ❖ Repeatable – automatic, consistent
 - ❖ Informative – provides feedback
 - ❖ Schedulable – auto-triggered
 - ❖ Portable – independent (of IDE)

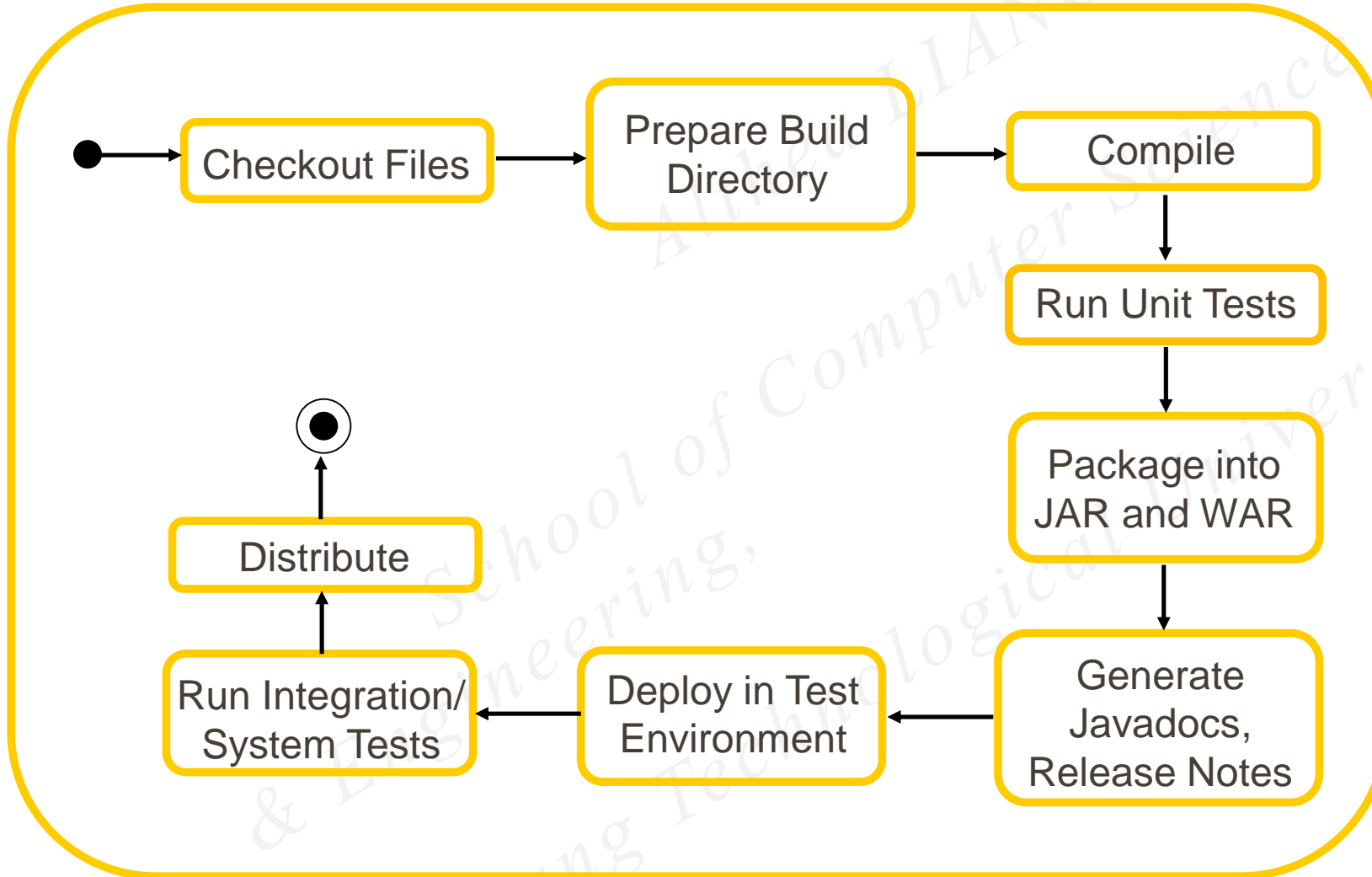
Release Distribution



Build-Package-Deploy Configuration



Build-Package-Deploy Process



Practice of Continuous Integration

- ▶ Integration components or subsystems
- ▶ Continuous along the way versus one-big-bang

- ▶ Definition:

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.

Practice of Continuous Integration

Combining “promotion – internal release management” and “integration testing”

- ❖ Automate build and make it self-testing
- ❖ All developers should commit frequently

Automate deployment and test in production-clone environment



Build frequently on an Integration Server

Outcome of best practices:

- ❖ Anyone can get the latest build easily
- ❖ Everyone can see what's happening

Post view: Best Practice Organisation in this Lesson

Best Practices/Tools	Slides
Continuous Integration	Practice of Continuous Integration - 1 Practice of Continuous Integration - 2
VCS Software	VCS Graph Branching and Merging

Summary

Now you should be able to:

- ▶ Explain activities of software release management
- ▶ Describe how the version control system works and use it
- ▶ Describe important practices of release management

*Nanyang Technological University
School of Computer Science
& Engineering,*

Special Thanks to Kydon during the TEL Efforts of the Lecture

End of Release Management

Faculty : Dr Althea Liang
School : School of Computer Science and Engineering
Email : qhliang@ntu.edu.sg
Office : N4-02c-107