

CONFIGURATION MANAGEMENT

Why Software Configuration Management ?

- ♦ The problem:
 - ♦ Multiple people have to work on software that is changing
 - ♦ More than one version of the software has to be supported:
 - ♦ Released systems
 - ♦ Custom configured systems (different functionality)
 - ♦ System(s) under development
 - ♦ Software must run on different machines and operating systems

⚡ *Need for coordination*

- ♦ Software Configuration Management
 - ♦ manages evolving software systems
 - ♦ controls the costs involved in making changes to a system

What is Software Configuration Management?

- ◆ Definition:

- ◆ A set of management disciplines within the software engineering process to develop a baseline.

- ◆ Description:

- ◆ Software Configuration Management encompasses the disciplines and techniques of initiating, evaluating and controlling change to software products during and after the software engineering process.

- ◆ Standards (approved by ANSI)

- ◆ IEEE 828: Software Configuration Management Plans
 - ◆ IEEE 1042: Guide to Software Configuration Management

Software Configuration Management is a Project Function

- ♦ SCM is a Project Function with the goal to make technical and managerial activities more effective.
- ♦ Software Configuration Management can be administered in several ways:
 - ♦ **A single software configuration management team for the whole organization**
 - ♦ **A separate configuration management team for each project**
 - ♦ **Software Configuration Management distributed among the project members**
 - ♦ **Mixture of all of the above**

Terminology

- ◆ We will define the following terms
 - ◆ **Configuration Item**
 - ◆ **Baseline**
 - ◆ **SCM Directories**
 - ◆ **Version**
 - ◆ **Revision**
 - ◆ **Release**

⚡ The definition of the terms follows the IEEE standard.

Terminology: Configuration Item

- ❖ Software configuration items are not only program code segments but all type of documents according to development, e.g.
 - ⚡ **all type of code files**
 - ⚡ **drivers for tests**
 - ⚡ **analysis or design documents**
 - ⚡ **user or developer manuals**
 - ⚡ **system configurations (e.g. version of compiler used)**
- ❖ In some systems, not only software but also hardware configuration items (CPUs, bus speed frequencies) exist!

Tasks for the Configuration Managers

Define configuration items

Possible Selection of Configuration Items

- ◆ Problem Statement
- ◆ Software Project Management Plan (SPMP)
- ☞ Requirements Analysis Document (RAD)
- ☞ System Design Document (SDD)
- ◆ Project Agreement
- ☞ Object Design Document (ODD)
- ◆ Dynamic Model
- ◆ Object model
- ◆ Functional Model
- ☞ Unit tests
- ◆ Integration test strategy
- ☞ Source code
- ◆ API Specification
- ☞ Input data and data bases
- ◆ Test plan
- ☞ Test data
- ☞ Support software (part of the product)
- ◆ Support software (not part of the product)
- ◆ User manual
- ◆ Administrator manual

Once the Configuration Items are selected, they are usually organized in a tree

Terminology: Version

- ◆ The initial release or re-release of a configuration item associated with a complete compilation or recompilation of the item. Different versions have different functionality.

Terminology: Baseline

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

Examples:

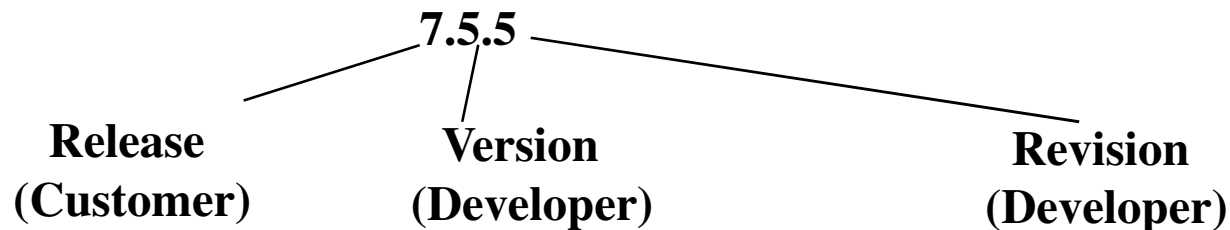
Baseline A: All the API have completely been defined; the bodies of the methods are empty.

Baseline B: All data access methods are implemented and tested.

Baseline C: The GUI is implemented.

More on Baselines

- ◆ As systems are developed, a series of baselines is developed, usually after a review (analysis review, design review, code review, system testing, client acceptance, ...)
 - ◆ *Developmental baseline*
 - ◆ **Goal: Coordinate engineering activities.**
 - ◆ *Functional baseline* (**first prototype, alpha release, beta release**)
 - ◆ **Goal: Get first customer experiences with functional system.**
 - ◆ *Product baseline* (**product**)
 - ◆ **Goal: Coordinate sales and customer support.**
- ◆ Many naming scheme for baselines exist (1.0, 6.01a, ...)
- ◆ A 3 digit scheme is quite common:



Terminology: Version vs. Revision vs. Release

♦ Version:

- ♦ **An *initial* release or re-release of a configuration item associated with a *complete compilation* or recompilation of the item. Different versions have different functionality.**

♦ Revision:

- ♦ ***Change* to a version that corrects only errors in the design/code, but does not affect the documented functionality.**

♦ Release:

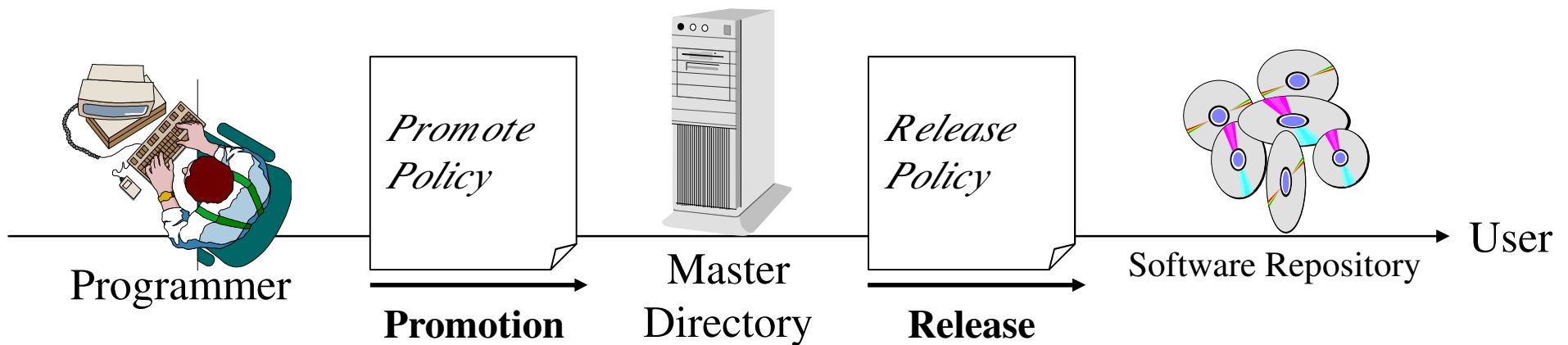
- ♦ **The *formal distribution* of an approved version.**

Change management

- ♦ Change management is the handling of change requests
 - ♦ **A change request leads to the creation of a new release**
- ♦ General change process
 - ♦ **The change is requested (this can be done by anyone including users and developers)**
 - ♦ **The change request is assessed against project goals**
 - ♦ **Following the assessment, the change is accepted or rejected**
 - ♦ **If it is accepted, the change is assigned to a developer and implemented**
 - ♦ **The implemented change is audited.**
- ♦ The complexity of the change management process varies with the project. Small projects can perform change requests informally and fast while complex projects require detailed change request forms and the official approval by one more managers.

Controlling Changes

- ♦ Two types of controlling change:
 - ♦ *Promotion*: The internal development state of a software is changed.
 - ♦ *Release*: A changed software system is made visible outside the development organization.



- ♦ Approaches for controlling change (Change Policy)
 - ♦ Informal (good for research type environments and promotions)
 - ♦ Formal approach (good for externally developed CIs and for releases)

Standard SCM Directories

- ◆ Programmer's Directory
 - ◆ (IEEE Std: "Dynamic Library")
 - ◆ Completely under control of one programmer.
- ◆ Master Directory
 - ◆ (IEEE Std: "Controlled Library")
 - ◆ Central directory of all promotions.
- ◆ Software Repository
 - ◆ (IEEE Std: "Static Library")
 - ◆ Externally released baselines.

