SIEMENS
Ingenuity for life

PLM and Innovation Excellence

Learning Campus
Your partner for Business Learning

Siemens Core Learning Program

Configuration and Change Request Management

Authors: Christian Bub, CT | Franz Kudorfer, CT | Rüdiger Kreuter, CT | Christian Hahn, CT

Keeping the overview

The future of search is finding answers, not links.

[Michael Brady]

SIEMENS
Ingenuity for life

Configuration and Change Request Management

Learning objectives

- Understand important concepts of configuration management
- Understand the specific aspects and concepts of disciplines
  - Software
  - Hardware / Mechanics
  - Systems
- Understand important concepts of change request and error management

Configuration and Change Request Management

Agenda

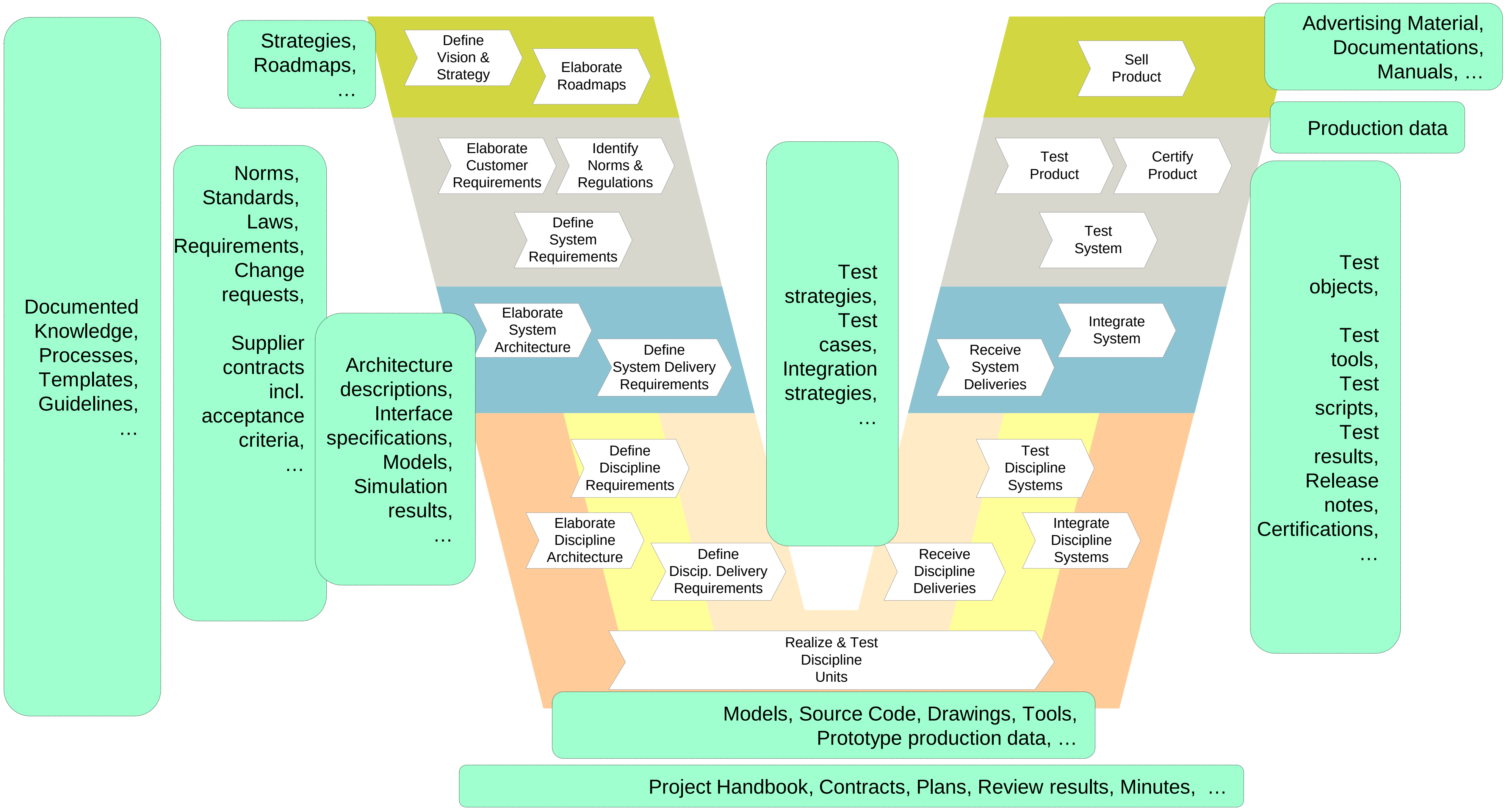
Intro

Configuration Management

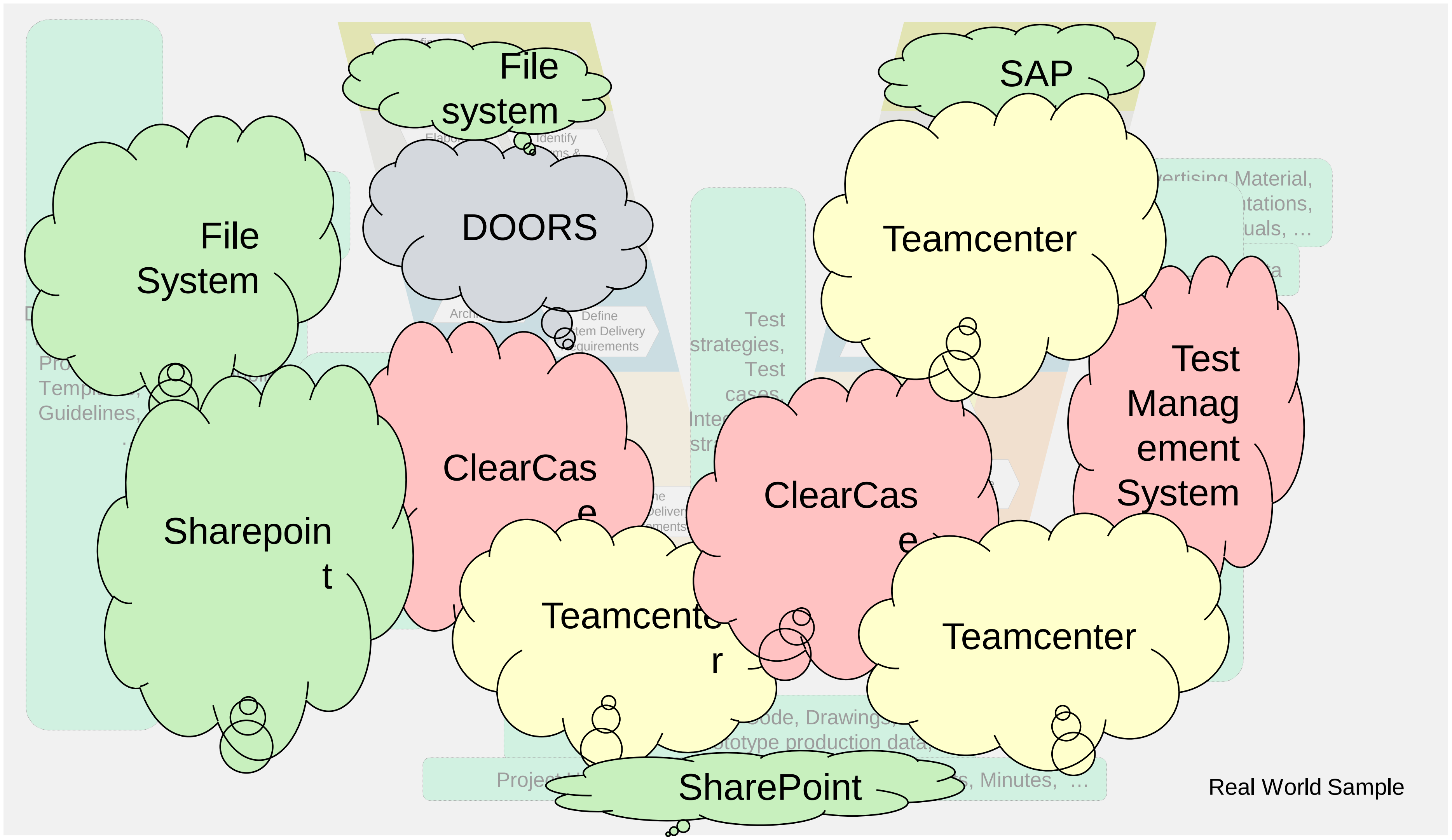
Change Request Management

Summary

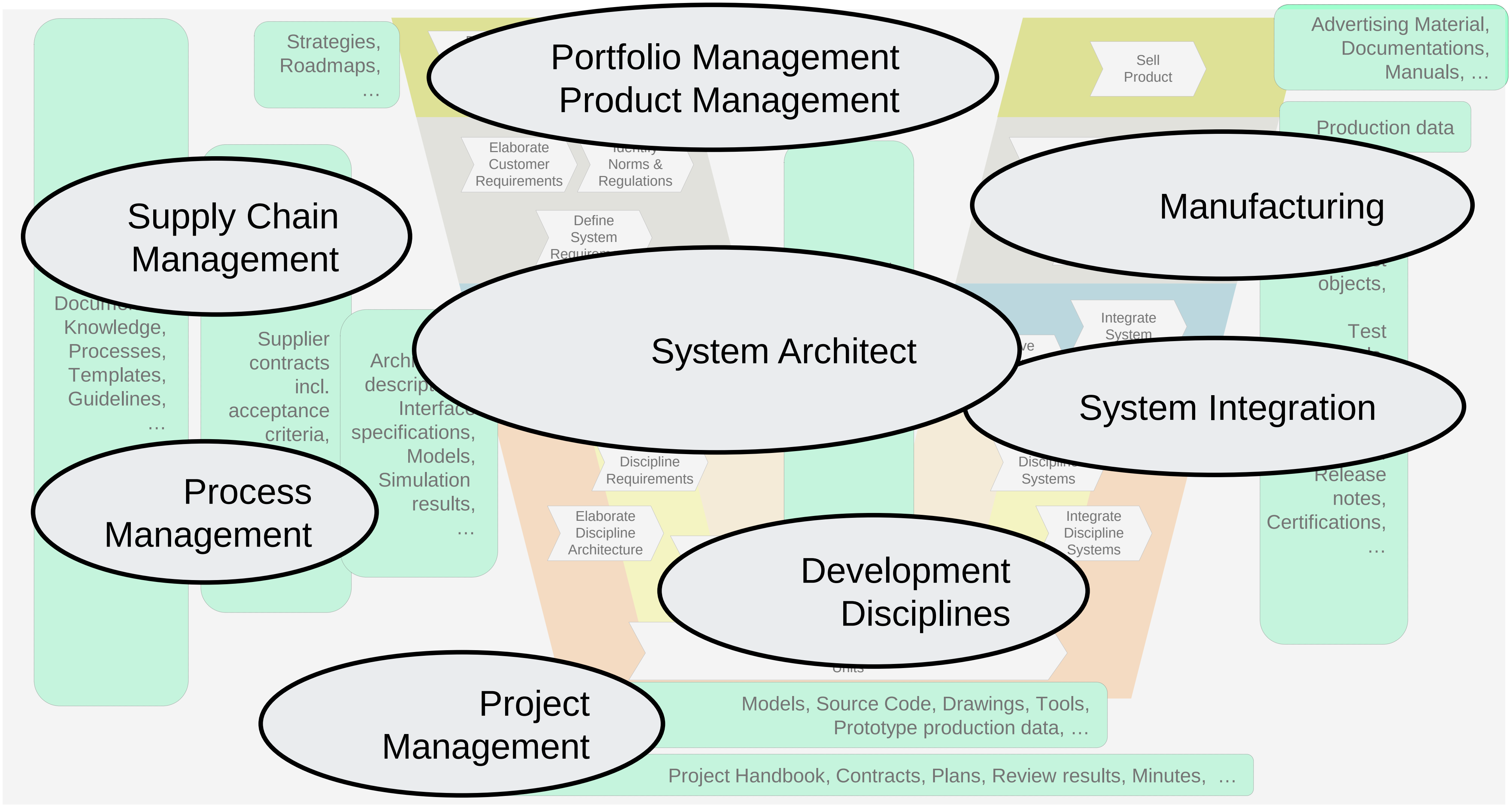
Numerous artifacts are created during a development project


Strategies, Roadmaps, …
Define Vision & Strategy
Elaborate Roadmaps
Sell Product
Advertising Material, Documentations, Manuals, …
Production data
Documented Knowledge, Processes, Templates, Guidelines, …
Norms, Standards, Laws, Requirements, Change requests,
Supplier contracts incl. acceptance criteria, …
Elaborate Customer Requirements
Identify Norms & Regulations
Define System Requirements
Test Product
Certify Product
Test System
Test strategies, Test cases, Integration strategies, …
Architecture descriptions, Interface specifications, Models, Simulation results, …
Elaborate System Architecture
Define System Delivery Requirements
Receive System Deliveries
Integrate System
Test objects,
Test tools, Test scripts, Test results, Release notes, Certifications, …
Define Discipline Requirements
Elaborate Discipline Architecture
Define Discip. Delivery Requirements
Receive Discipline Deliveries
Test Discipline Systems
Integrate Discipline Systems
Realize & Test Discipline Units
Models, Source Code, Drawings, Tools, Prototype production data, …
Project Handbook, Contracts, Plans, Review results, Minutes, …

Artifacts
are stored in different tools



Artifacts are under different responsibilities



Strategies, Roadmaps, …
Portfolio Management Product Management
Sell Product
Advertising Material, Documentations, Manuals, …
Production data
Elaborate Customer Requirements
Identify Norms & Regulations
Define System Requirements
Supply Chain Management
Manufacturing
Documents, Knowledge, Processes, Templates, Guidelines, …
Supplier contracts incl. acceptance criteria,
Architecture descriptions, Interface specifications, Models, Simulation results, …
System Architect
Integrate System
objects,
Test
System Integration
Discipline Requirements
Discipline Systems
Release notes, Certifications, …
Process Management
Elaborate Discipline Architecture
Integrate Discipline Systems
Development Disciplines
Units
Project Management
Models, Source Code, Drawings, Tools, Prototype production data, …
Project Handbook, Contracts, Plans, Review results, Minutes, …

Typical issues with artifact management

Several different versions of an artifact may be needed in parallel;
e.g. a customer specific version besides the main line.

Several people may work on the same artifact;
e.g. an architecture document.

Artifacts are managed using different tools

Data corruption (or a similar accident),
enforces a fall back to an older version.

Comparison between (changed) versions may be necessary;
e.g. for audits.

Professional artifact / configuration management is necessary for keeping control

SIEMENS
Ingenuity for life

Configuration Management (CM)

… fosters consistency between product requirements and the released product

… ensures that changes to requirements or artifacts are controlled and traceable

… ensures that the (most current) artifacts are known and can be found easily throughout their lifecycle

- ➔ Fostering completeness and integrity
- ➔ Active monitoring of changes instead of being driven by changes
- ➔ Fast information retrieval
- ➔ Protection against information loss
- ➔ Support for efficient project management

Elements
of a professional Configuration Management

SIEMENS
Ingenuity for life

Configuration item identification

Repository structures

Defect correction

Product liability

Decision

Status model

Naming conventions

Impact analysis

Baseline

Long term maintenance

Version management

New baseline

Change Request

Configuration item creation

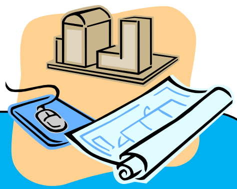
Defect detection

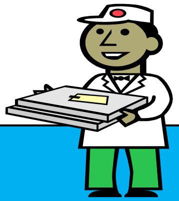
Correction, verification

SIEMENS

Ingenuity for life

Typical Situations in Daily Work





Decompose system

Create, review and release artifact

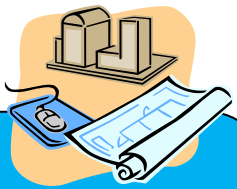
Deliver component or system

Archive artifact

Manage Changes

Manage Defects

Typical Situations in Daily Work related to the elements of Configuration Management

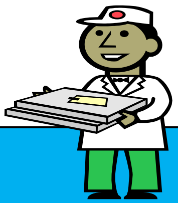


Decompose system

- Repository structures
- Naming conventions
- Configuration item identification
- Version management

Create, review and release artifact

- Configuration item creation
- Status model

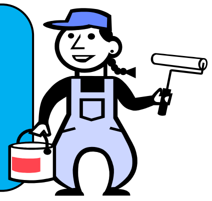


Deliver component or system

- Status model
- Baseline

Archive artifact

- Product liability
- Long term maintenance



Manage Changes

- Change Request
- Impact analysis
- Decision
- Implement, New baseline

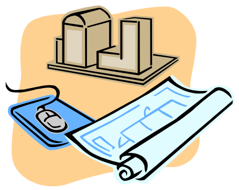
Manage Defects

- Defect detection
- Defect correction
- Regression testing
- New baseline

Configuration and Change Request Management

Agenda

Decompose System: Identify Configuration Items



SIEMENS
Ingenuity for life

A development project may produce thousands of artifacts.

Any of these artifacts may be brought under CM control and become a "Configuration Item“ (CI).

But, not every artifact needs to be!

➔ Select Configuration Items carefully

➔ Select the time when CM control starts

- starting late may cause risk
- starting too early produces overhead

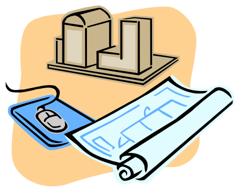
Note :

- Only human made artifacts should be identified as configuration items.
- It is not necessary to apply configuration control for "derived" artifacts; that is, for artifacts that could be easily reproduced by tools at any time

  Exception: released derived items when transferred to a different configuration area (e.g. software to system level or from development to production)

CM	Configuration Management

Decompose System: Apply Naming Conventions



SIEMENS
Ingenuity for life

Each configuration item (CI) must be uniquely identifiable.

- Standardized names of configuration items enable
  - Recognition by human beings
  - Retrieval by categories beside storage structures
  - Identification of items for baselines

Naming examples:

[system]_[doc-type]_[.xxx]

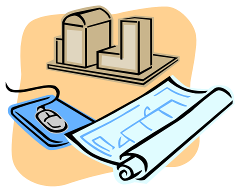
R-[pool_id]-[number]

- CM tools create internal identifiers (ID)
  (e.g. item ID in Teamcenter, requirement ID in DOORS)

➔ Use date/time stamps, versions, etc. in names only when not stored in a CM tool that supports identification and versioning!

CI	Configuration Item
CM	Configuration Management
ID	Identifier

Decompose System:
Further Configuration Item Attributes



SIEMENS
Ingenuity for life

Good practice: maintain a CI List, that includes all CIs in the project.

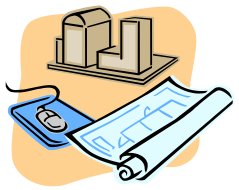
- Name
- Identificator
- Responsible owner
- Access rights
  (e.g. groups / individuals; read / write / delete)
- Storage type
  (e.g. paper, hard drive / server, CM System, document management system)
- Storage path / address
- Control level
- Start time (e.g. a certain milestone)
- Baselines, to which the CI shall belong

If subject to archiving:

- Storage type
  (e.g. paper, tape, disc, archive system, …)
- Storage path / address
- Archiving time
  (e.g. a certain milestone)
- Archiving period
  (e.g. 20 years)

CI	Configuration Item
CM	Configuration Management

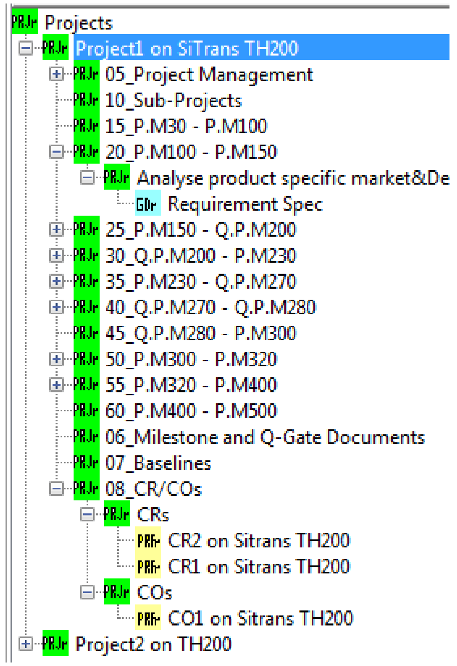
Decompose System: Define Repository Structure



The repository structure follows the purpose of the CM system.

Project oriented

Principle: phases/milestones



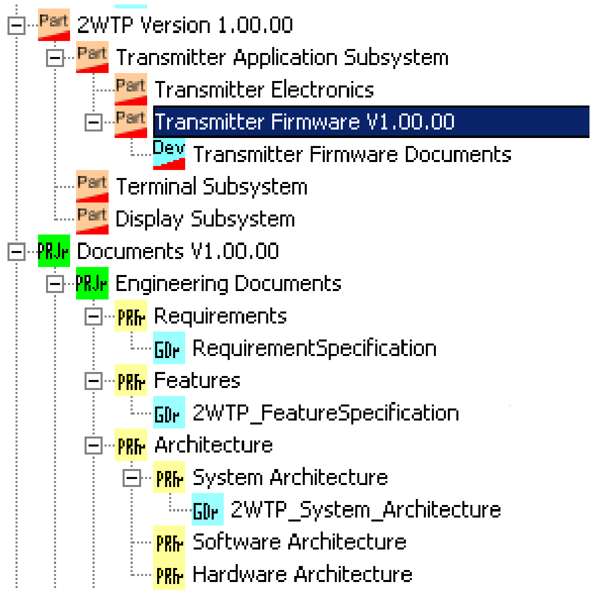
Projects
- Project1 on SiTrans TH200
  - 05_Project Management
  - 10_Sub-Projects
  - 15_P.M30 - P.M100
  - 20_P.M100 - P.M150
    - Analyse product specific market&De
      - Requirement Spec
  - 25_P.M150 - Q.P.M200
  - 30_Q.P.M200 - P.M230
  - 35_P.M230 - Q.P.M270
  - 40_Q.P.M270 - Q.P.M280
  - 45_Q.P.M280 - P.M300
  - 50_P.M300 - P.M320
  - 55_P.M320 - P.M400
  - 60_P.M400 - P.M500
  - 06_Milestone and Q-Gate Documents
  - 07_Baselines
  - 08_CR/COs
    - CRs
      - CR2 on Sitrans TH200
      - CR1 on Sitrans TH200
    - COs
      - CO1 on Sitrans TH200
- Project2 on TH200

Consistency rules

(manual or tool)

System/product oriented

Principle: architecture/break-down



- 2WTP Version 1.00.00
  - Transmitter Application Subsystem
    - Transmitter Electronics
    - Transmitter Firmware V1.00.00
      - Transmitter Firmware Documents
  - Terminal Subsystem
  - Display Subsystem
- Documents V1.00.00
  - Engineering Documents
    - Requirements
      - RequirementSpecification
    - Features
      - 2WTP_FeatureSpecification
    - Architecture
      - System Architecture
        - 2WTP_System_Architecture
      - Software Architecture
      - Hardware Architecture

CI	Configuration Item
CM	Configuration Management

Create, review and release artifact: Use status model

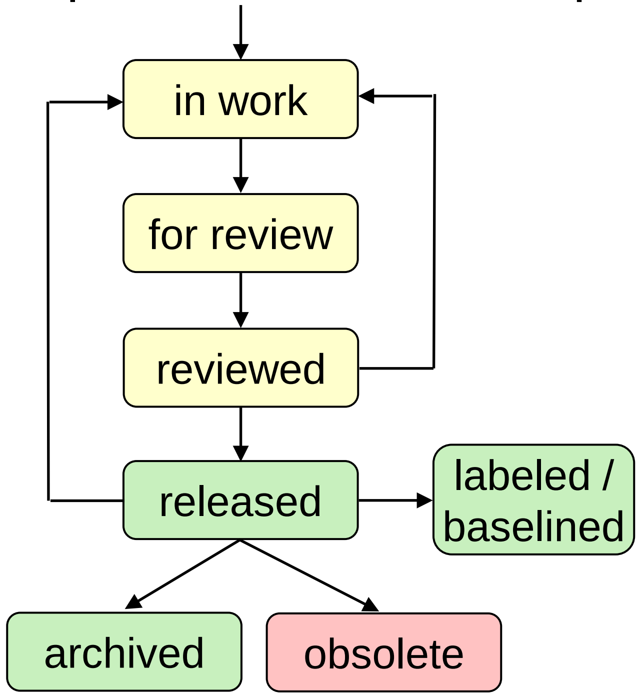
SIEMENS

Ingenuity for life

An artifact is produced over a time span with different roles working on it.

- Author sets up draft
- Author and experts contribute (chapters, additions, modules)
- Author submits for review
- Reviewers inspect
- Author implements review comments
- Author releases
- Configuration Manager defines baseline
- …

Simple state model example:



Excursion: Discipline Specific Aspects

SIEMENS
Ingenuity for life

Discipline specific processes impose discipline specific CM features.

Hardware

- Integration with CAx and simulation tools
- Structure derived from architecture (BOM)
- Physical test objects (samples), are usually managed manually

System

- Structure BOM oriented
- Software executable is a "part"
- Different BOM Views
  - Engineering BOM
  - Manufacturing BOM
  - Product variant management

Software

- Integration with software development tools
- High number of generated modules
- Frequent creation of executables (e.g. "daily build")
- Concurrent work on modules (e.g. bug-fix vs. feature) needs branching, merging concepts

BOM	Bill of Material
CAx	Computer Aided tools for design, engineering, manufacturing, etc.
CM	Configuration Management
PLM	Product Lifecycle Management

Excursion:
Discipline Specific Aspects

SIEMENS
Ingenuity for life

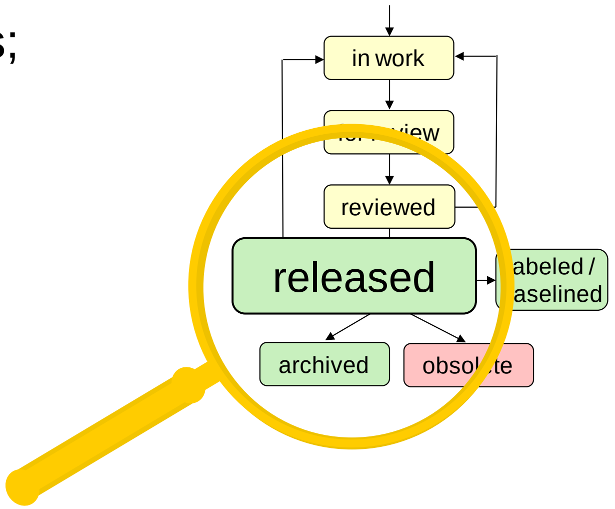
	Software	Hardware	System
Integration of development and CM tools	Strong	Strong	Weak
Frequency of generated objects	High (e.g. daily build)	Low	Low
Support for parallel development / maintenance lines	Branching / merging concept	Not required	Required but not well supported by tools
Other	Generated source code, high number of configuration items	Physical test objects	BOM Views (development, manufacturing, …)

BOM	Bill of Material
CM	Configuration Management

Deliver a Component or System: Define Baseline (I)

A formally released set of Configuration Items, become a “Baseline" for further development.

- Baselines have to be **planned and controlled** by the project.
- Baselines have to be **checked for integrity and consistency**.
- A configuration item may belong to several baselines; e.g. a system may be
  - released for verification
  - released for validation
  - released for customer A
  - released for customer B
  - …
- Baselines may be consecutive or parallel



Deliver a Component or System: Define Baseline (II)

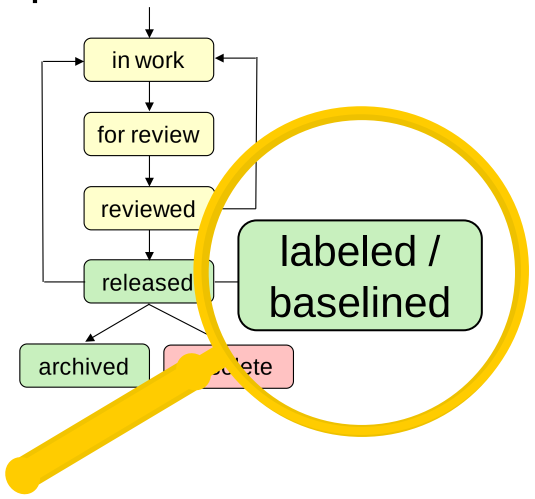
Define an exact configuration of (a part of) the system at a certain time.

Requirements for baselines:

- Completeness
  Document all CIs of a baseline with their versions and status.
- Reproducibility
  Ensure that all derived objects can be reproduced at any time.
- Immutability
  Protect all CIs of a baseline against any change.
  (changes trigger new baselines!)

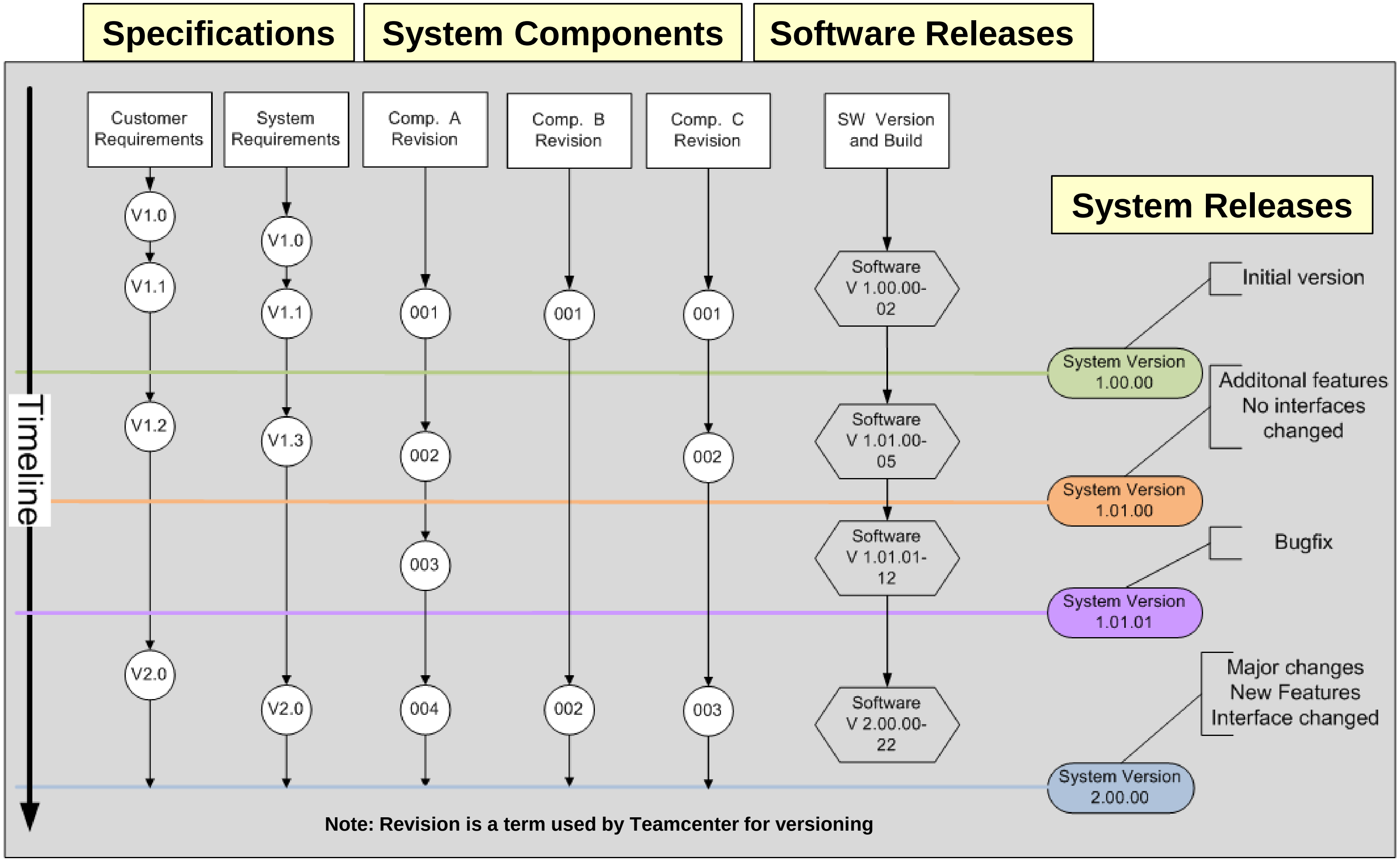
Automated support for baseline creation:

- Labels (e.g. used by Software CM tools)
- BOM definitions (e.g. used by PLM tools)
- Write protection



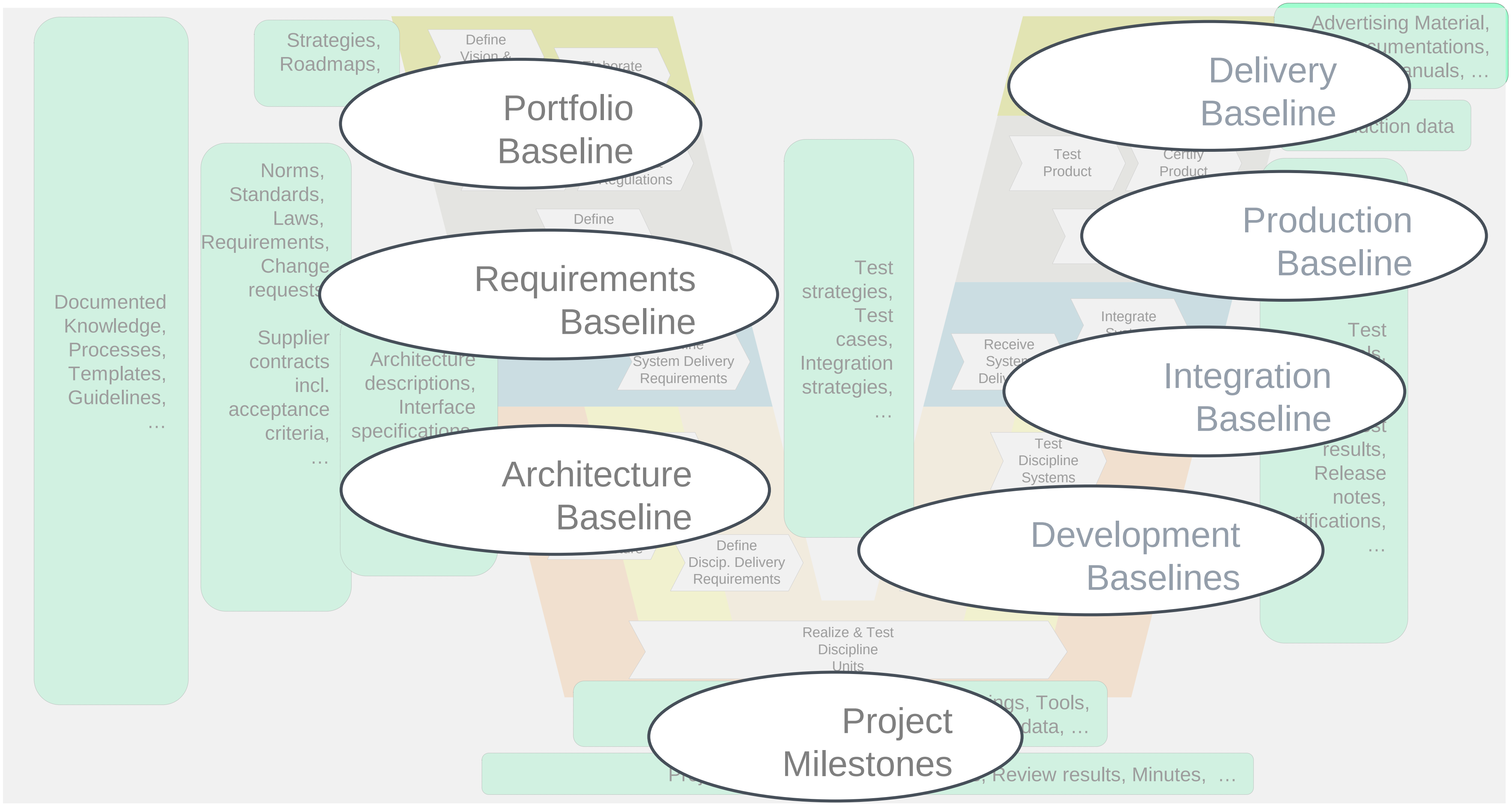
CI	Configuration Item
CM	Configuration Management
BOM	Bill of Material
PLM	Product Lifecycle Management

Deliver a Component or System: Baselines and System Releases (Example)



Note: Revision is a term used by Teamcenter for versioning

Different Baselines are Defined During the Project and the System Lifecycle


Documented Knowledge, Processes, Templates, Guidelines, …
Strategies, Roadmaps,
Norms, Standards, Laws, Requirements, Change requests
Supplier contracts incl. acceptance criteria, …
Architecture descriptions, Interface specifications
Define Vision &
Elaborate
Regulations
Define
System Delivery Requirements
Define Discip. Delivery Requirements
Test strategies, Test cases, Integration strategies, …
Realize & Test Discipline Units
Test Discipline Systems
Receive System Deliv
Integrate
Test Product
Certify Product
Advertising Material, Documentations, Manuals, …
Production data
Test results, Release notes, Certifications, …
Drawings, Tools, data, …
Review results, Minutes, …
Portfolio Baseline
Requirements Baseline
Architecture Baseline
Delivery Baseline
Production Baseline
Integration Baseline
Development Baselines
Project Milestones

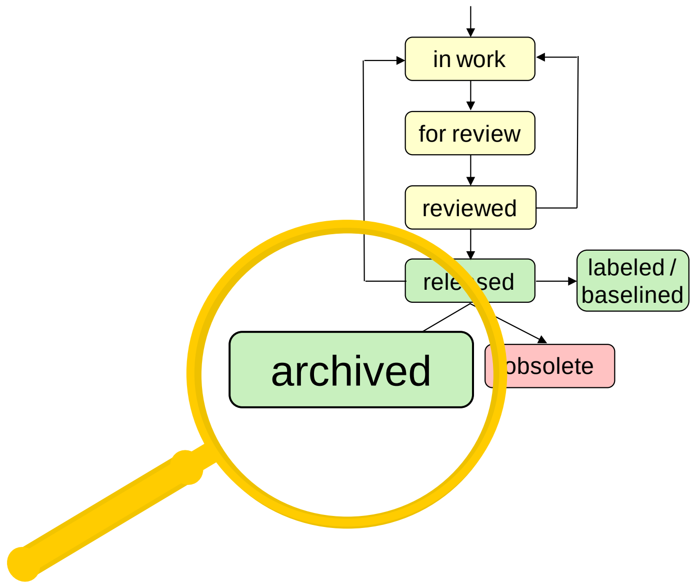
Archive Artifacts

All artifacts that are needed beyond project lifetime shall be archived for long term use.

Long term use may be triggered by

- Version development
- Product liability
- Long term maintenance obligations
- Re-use of documented knowledge

Architectures are usually subject to archiving!



Note1: Error fixing requires not only the artifact to fix, but also the necessary software and hardware tools! Therefore it may be necessary to archive the development environment, too. (“obsolescence management”)

Note2: Long-term archiving does not replace back-up mechanisms of CM tools!

SIEMENS
Ingenuity for life

Attributes of Professional Configuration Management

- Standard artifact repository structures
- Naming conventions
- Status Model
- List of (identified) configuration items

Issue of organizational and project process

- Version management / control for frequently changing artifacts
- Change control for important artifacts, at least for released artifacts
- Baselining for (sets of) released artifacts
- Access right management (e.g. read, write, delete)
- Access synchronization (e.g. check out / in, branch / merge)
- Appropriable tool and IT support
- Managed backups

Provided or supported by a CM tool

- Archiving for artifacts (Documents, SW, HW) that are needed beyond project lifetime

Archive

Configuration and Change Request Management

Agenda

Change Management & Defect Management

In an ideal project**:**

- Technical objectives are once agreed and persist until project closure.
- All implemented deliverables comply straightaway with the intended outcome.

A **real project** needs:

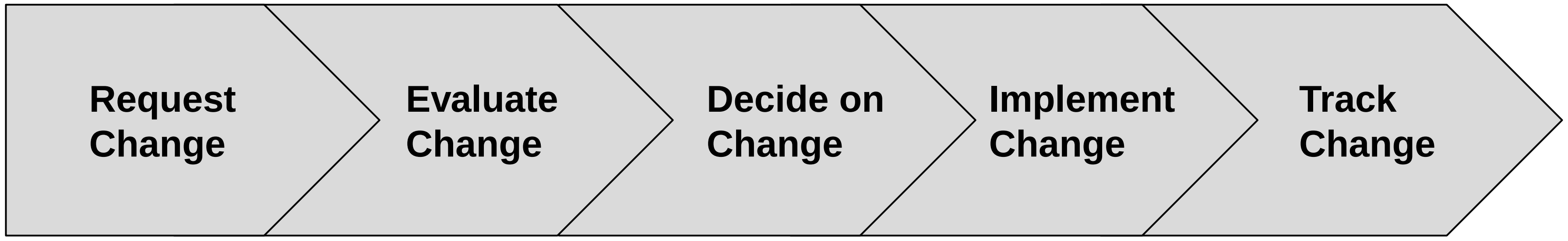
Change Management	because previously agreed requirements are incomplete, erroneous or ambiguous, needing amendments or adaptation
Defect Management	because tests reveal defects that need correction or rework

Change Management:
Change Request Process

SIEMENS
Ingenuity for life

No project without changes!

No improvement without changes!



Change Request Management: Request Change

Requestor can be any internal stakeholder, or a customer, or a supplier.

Basic attributes for Change Requests:

- Identifier
- Name
- Requestor
- Problem or wish
- Change proposal

Advanced attributes:

- Impact analysis result
- Approval / rejection dates
- Due date for implementation
- Trace link to related requirements
- Trace link to implementation
- If necessary signatures

Best Practice: Introduce a Single Entry Point for change requests

Change Request Management: Evaluate and Decide on Change

SIEMENS
Ingenuity for life

Understand impact on the system and all disciplines or sub-systems.

Decision by Change Control Board (CCB)

A group of persons with assigned responsibility and authority to make decisions on the change.

Typically includes

- Product Manager
- Project Manager
- (System) Architect

Impact Analysis

- Rough initial evaluation by the CCB to decide on further action
- Deep analysis by experts
  - on architecture, cost, schedule
  - all affected disciplines
  - all relevant sub-systems
- Propose further proceeding (accept / defer / reject)

Change Request Management: Implement and Track Change

Change of artifacts leads to definition of a new baseline.

Implement Change

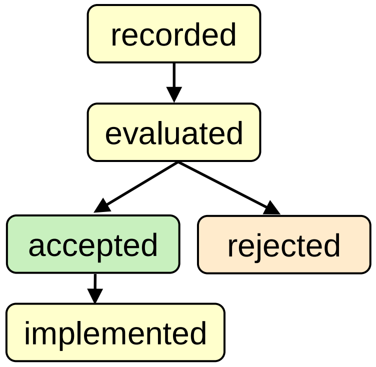
- Done by configuration item owner
- Review and release
- Create new baseline

→ Usually a new baseline is defined for a number of changes.

→ Agile methodologies support ease of change by regular baselines (iterations / sprints).

Track Change

- Change requests are recorded in a change request system
- A status model applies; e.g.:

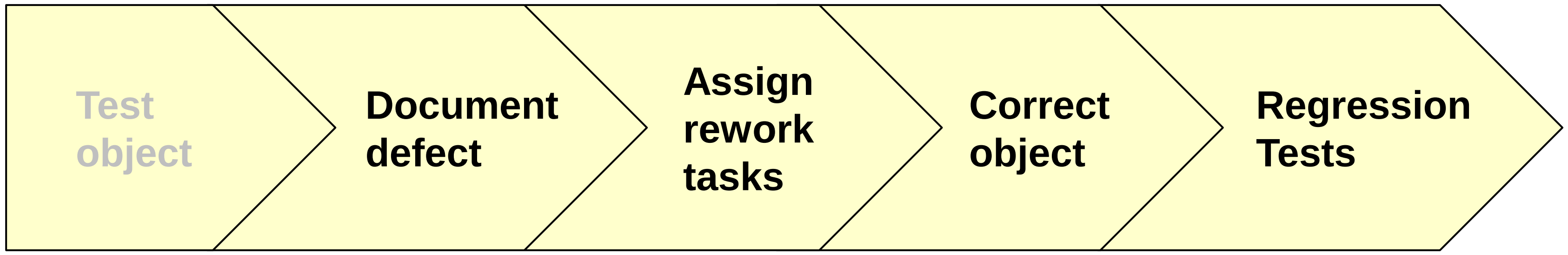


Defect Management:
Defect Management Process

SIEMENS
Ingenuity for life

No project without defects!

No improvement without errors!



Best practice: Common tool and process for changes and defects (e.g. CHARMs at Healthcare)

SIEMENS

Ingenuity for life

Configuration and Change Request Management

Agenda

What we have learned

A development project produces numerous artifacts, many of them are subject to change.

A professional artifact management is necessary to allow for concurrent development.

Important artifacts are Configuration Items and thus subject to version, change and baseline control.

Change request management helps to keep the overview and to reduce cost.

Change management and defect management include similar aspects.

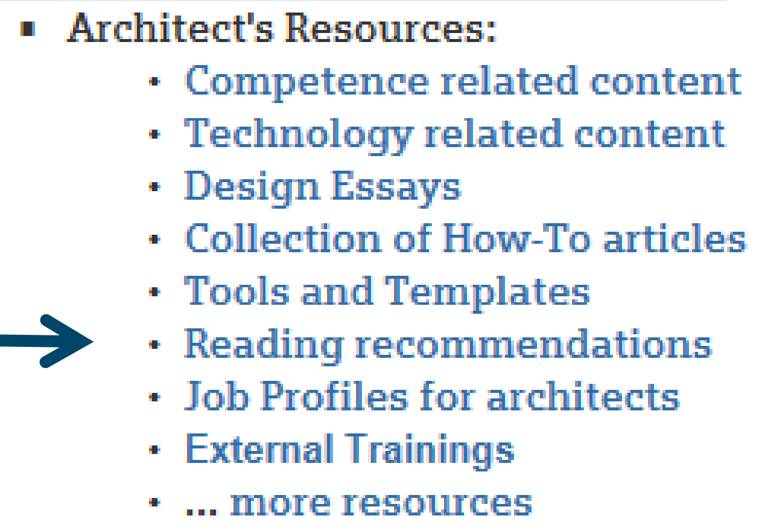
SIEMENS

Ingenuity for life

Further readings

Use the SSA Wiki :
https://wiki.ct.siemens.de/x/fReTBQ

and check the “Reading recommendations”:
https://wiki.ct.siemens.de/x/-pRgBg



- Architect's Resources:
  - Competence related content
  - Technology related content
  - Design Essays
  - Collection of How-To articles
  - Tools and Templates
  - Reading recommendations
  - Job Profiles for architects
  - External Trainings
  - ... more resources