SE345

Atılım University
Dept of Software Engineering

Asst. Prof. Dr. Aylin AKCA-OKAN

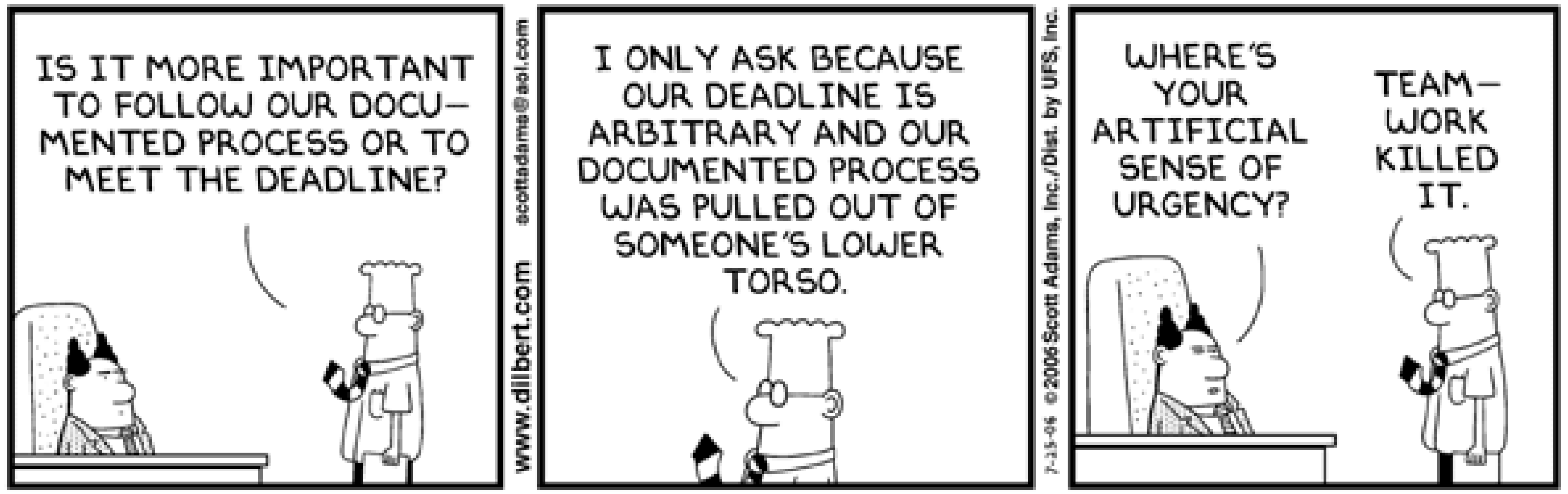
Tentative Course Schedule

Wk	Subjects	Chapter
1.	Introduction to Software Quality and Assurance	Chapter 1
2.	Introduction to Software Quality and Assurance	Chapter 1
3.	Software Quality Factors	Chapter 3
4.	Overview of Components of the SQA System	Chapter 4
5.	Overview of Components of the SQA System	Chapter 4
6.	Integrating Quality Activities in Project Life Cycle	Chapter 7
7.	Midterm	
8.	Reviews, Inspection and Audits	Chapter 8
9.	Software Quality Metrics	Chapter 21
10.	Procedures and Work Instructions	Chapter 14
11.	Software Change Process	Chapter 18
12.	Presentations	
13.	SQA Process Standards	Chapter 23

Procedures and Work Instructions

Infrastructure

Infrastructure components are tools employed to prevent software errors and promote the quality level of the entire organisation.


IS IT MORE IMPORTANT TO FOLLOW OUR DOCUMENTED PROCESS OR TO MEET THE DEADLINE?
I ONLY ASK BECAUSE OUR DEADLINE IS ARBITRARY AND OUR DOCUMENTED PROCESS WAS PULLED OUT OF SOMEONE'S LOWER TORSO.
WHERE'S YOUR ARTIFICIAL SENSE OF URGENCY?
TEAMWORK KILLED IT.
www.dilbert.com scottadams@aol.com
7-15-06 ©2006 Scott Adams, Inc./Dist. by UFS, Inc.

© Scott Adams, Inc./Dist. by UFS, Inc.

Typical infrastructure components

- Procedures and work instruction.
- Quality support devices like templates and checklists.
- Staff SQA training and certification activities.
- Preventive and corrective actions.
- Software configuration management.
- Documentation and quality records control.

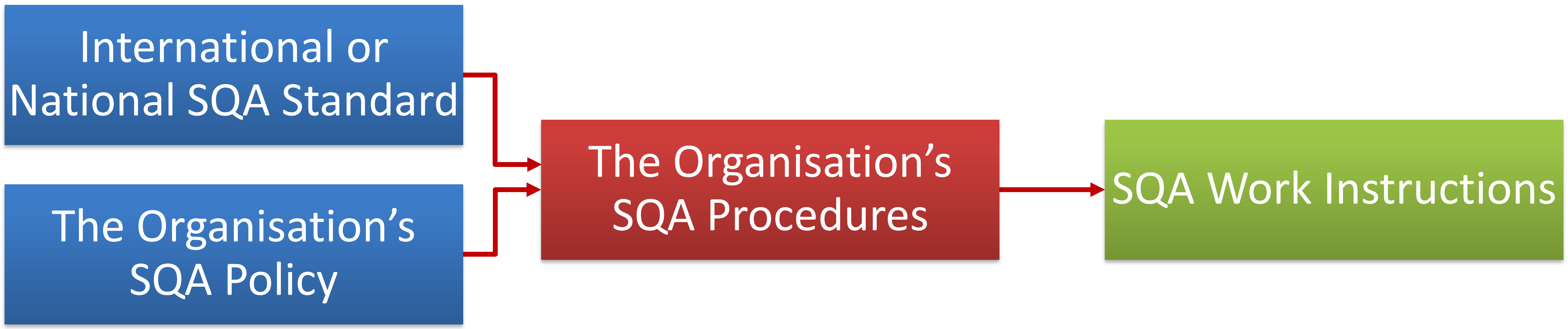
Questions

- Why **should we use** SQA procedures and work instructions?
- **Wouldn't it be better** if every professional relied on his own experience and performed the task the best way he knows?
- What are the **organisational benefits** of forcing me to perform a task exclusively in the way specified by the company?

Objectives & Benefits

- Performance of tasks, processes, or activities in the **most effective and efficient way** without deviating from quality requirements.
- **Effective and efficient communication** between the different teams involved in the development and maintenance of software systems.
- **Uniformity** in performance, achieved by conformity with procedures and work instructions, reduces misunderstandings that lead to software errors.
- **Simplified coordination between tasks and activities** performed by the various bodies of the organisation. Better coordination translates into fewer errors.

Conceptual hierarchy of procedures and work instructions


International or National SQA Standard
The Organisation's SQA Policy
The Organisation's SQA Procedures
SQA Work Instructions

- procedure - general methodology
- work instructions - details of how to implement those procedures for this project
- template - a standard Table of Contents
- checklist - list of items that are part of an activity

Which Philosophy is correct?

- True greatness is measured by how much freedom you give to others, not by how much you can coerce others to do what you want.

Larry Wall
Creator of Perl
March 1999

- Standardisation is the key.

paraphrase of ISO 9000-3

The Five Ws: Issues Resolved by Procedures

What activities should be performed?

Ho**W** should the activity be performed?

When should the activity be performed?

Where should the activity be performed?

Who should perform the activity?

Procedures

> “a particular way of accomplishing something or of acting”
> (Webster's New College Dictionary).

- **Detailed activities or processes to be performed**, according to a given method, for the purpose of accomplishing a task.
- The procedures adopted by an organisation are considered to be **binding for the organisation employees**, meaning that each employee is to perform his or her tasks according to **the steps appearing in the relevant procedure document** .
- Procedures also tend to be **universal within an organisation**, meaning that they are **applied whenever the task is performed,** irrespective of the person performing the task, or the organisational context.

Documenting your SQA Procedures

The collection of SQA procedures is usually called the "**SQA Procedure Manual**"

A good reference for how to build a SQA Manual for software companies is **ISO 90003**

ISO 90003
Software Quality Management Standard

- ISO IEC 90003:2018 is an update of ISO 90003:2014 and is based on the ISO 9001:2015 quality management standard.
- Both standards are organised in the same way and use basically the same numbering system.
- ISO 9001 is part of the ISO 9000 family of standards, which provides a framework for quality management systems (QMS). It focuses on meeting customer needs and enhancing satisfaction through effective system implementation.
- ISO/IEC 90003, on the other hand, is tailored specifically for software engineering. It offers guidelines for organisations seeking to apply ISO 9001 principles to software development processes, emphasising the unique challenges and requirements of this domain.
- ISO/IEC 90003 provides guidance for organisations in the application of ISO 9001:2015 to the acquisition, supply, development, operation and maintenance of computer software and related support services. It does not add to or otherwise change the requirements of ISO 9001:2015.

- *https://www.iso.org/standard/74348.html*
- *https://www.walturn.com/insights/understanding-iso-9001-and-90003-for-software-quality-management*
- *https://cdn.standards.iteh.ai/samples/74348/4d03d0c03f9048209d7eb232becb4fdf/ISO-IEC-IEEE-90003-2018.pdf*

How to meet the standard

- **ISO 9001:2015 Requirements (*must*)**
- **ISO IEC 90003:2018 Guidelines**
  - Recommendations (*should*)
  - Suggestions (*may*)

Items your Quality Manual should cover

- 4.1 Management responsibilities
- 4.2 Quality system requirements
- 4.3 Contract review requirements
- 4.4 Product design requirements
- 4.5 Document and data control
- 4.6 Purchasing requirements
- 4.7 Customer-supplied products
- 4.8 Product identification and tracing
- 4.9 Process control requirements
- 4.10 Product inspection and testing
- 4.11 Control of inspection equipment
- 4.12 Inspection and test status of products
- 4.13 Control of nonconforming products
- 4.14 Corrective and preventive action
- 4.15 Handling, packaging, and delivery
- 4.16 Control of quality records
- 4.17 Internal quality audit requirements
- 4.18 Training requirements
- 4.19 Servicing requirements
- 4.20 Statistical techniques

according to ISO 9001

Procedures Manual

The contents of an organisation's procedures manual vary according to:

- The types of software development and maintenance activities carried out by the organisation.
- The range of customers (e.g., internal/customers of custom-made software/COTS software customers) and suppliers (e.g., self-development and maintenance/subcontractors/suppliers of COTS software and reused software modules).
- Organisations seeking a higher quality assurance level usually achieve it by **adopting a national or international quality assurance standard**.
- An organisation that adopts a quality standard has to adapt its procedures to the standard's requirements
  - where the standard defines <u>“what has to be performed”</u> and
  - the relevant organisation's procedure determines <u>“how this requirement is to be fulfilled.”</u>

The Fixed Table of Contents for Procedures

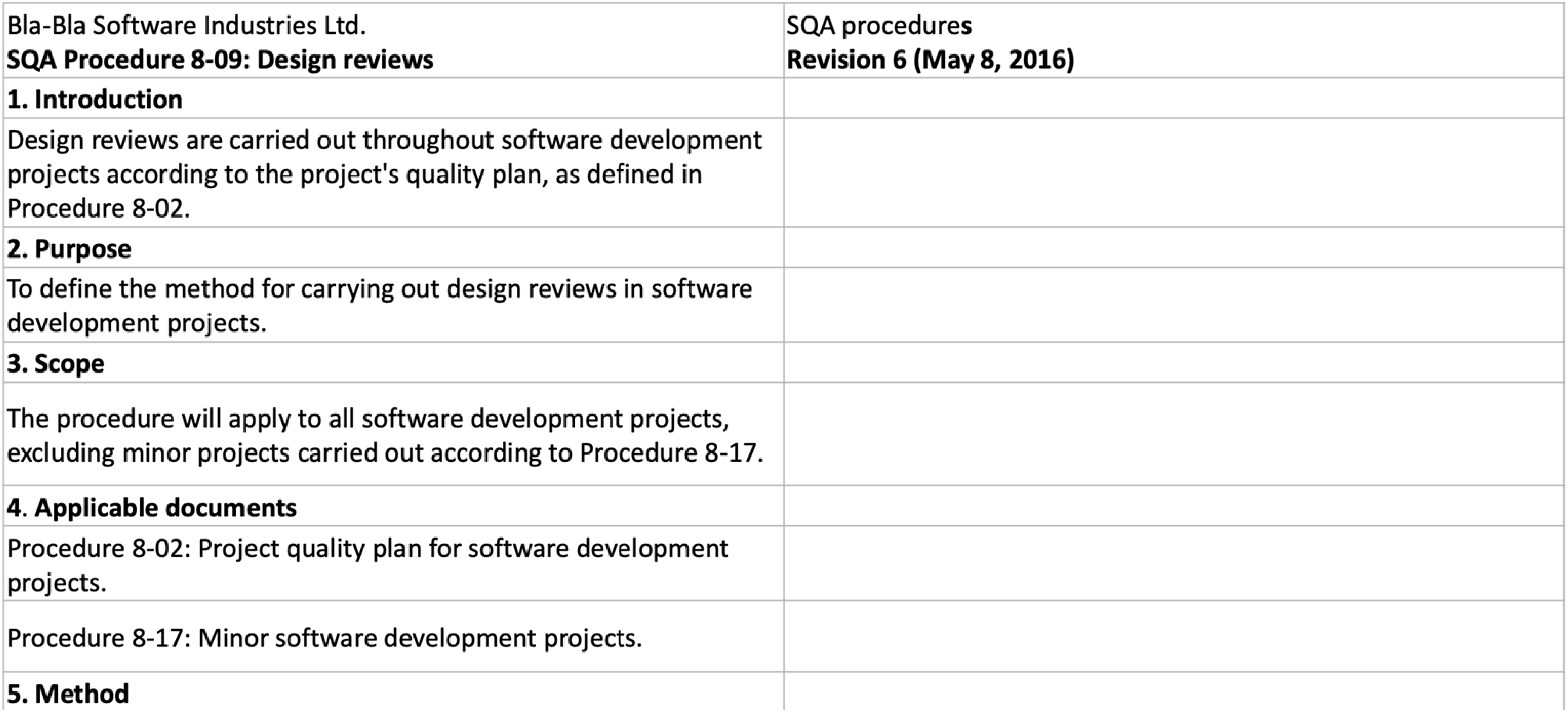
Standardisation – the application of a fixed format and structure – is the principle applied to all SQA procedures

1. Introduction *
2. Purpose
3. Terms and abbreviations *
4. Applicable documents
5. Method
6. Quality records and documentation
7. Reporting and follow up *
8. Responsibility for implementation *
9. List of appendices *

Appendices *

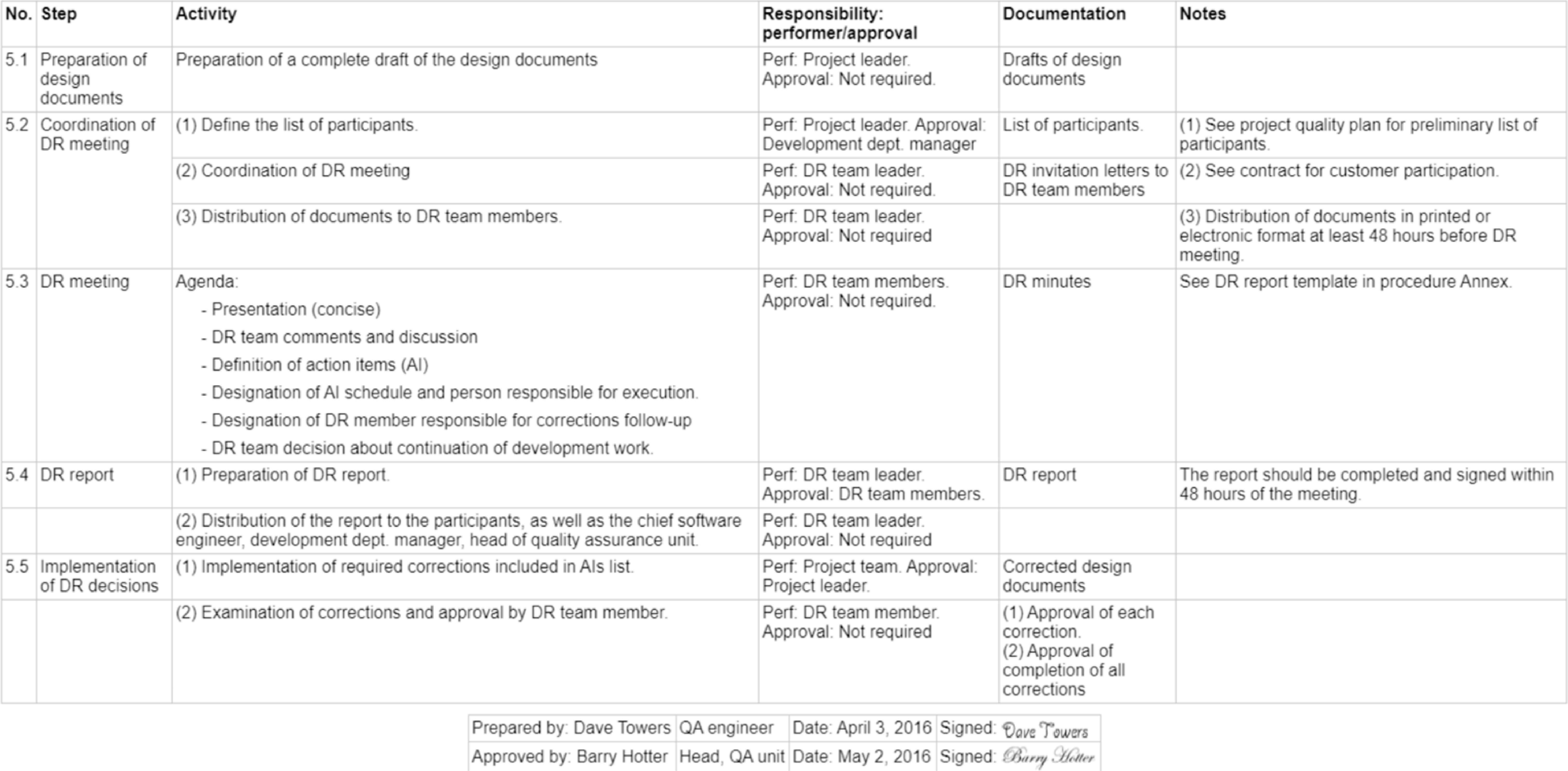
* Sections included only if applicable

A Design Review Procedure - Example



Bla-Bla Software Industries Ltd. **SQA Procedure 8-09: Design reviews**	SQA procedures **Revision 6 (May 8, 2016)**
1. Introduction	
Design reviews are carried out throughout software development projects according to the project's quality plan, as defined in Procedure 8-02.	
2. Purpose	
To define the method for carrying out design reviews in software development projects.	
3. Scope	
The procedure will apply to all software development projects, excluding minor projects carried out according to Procedure 8-17.	
4. Applicable documents	
Procedure 8-02: Project quality plan for software development projects.	
Procedure 8-17: Minor software development projects.	
5. Method	

A Design Review Procedure - Example



No.	Step	Activity	Responsibility: performer/approval	Documentation	Notes
5.1	Preparation of design documents	Preparation of a complete draft of the design documents	Perf: Project leader. Approval: Not required.	Drafts of design documents	
5.2	Coordination of DR meeting	(1) Define the list of participants.	Perf: Project leader. Approval: Development dept. manager	List of participants.	(1) See project quality plan for preliminary list of participants.
		(2) Coordination of DR meeting	Perf: DR team leader. Approval: Not required.	DR invitation letters to DR team members	(2) See contract for customer participation.
		(3) Distribution of documents to DR team members.	Perf: DR team leader. Approval: Not required		(3) Distribution of documents in printed or electronic format at least 48 hours before DR meeting.
5.3	DR meeting	Agenda: - Presentation (concise) - DR team comments and discussion - Definition of action items (AI) - Designation of AI schedule and person responsible for execution. - Designation of DR member responsible for corrections follow-up - DR team decision about continuation of development work.	Perf: DR team members. Approval: Not required.	DR minutes	See DR report template in procedure Annex.
5.4	DR report	(1) Preparation of DR report.	Perf: DR team leader. Approval: DR team members.	DR report	The report should be completed and signed within 48 hours of the meeting.
		(2) Distribution of the report to the participants, as well as the chief software engineer, development dept. manager, head of quality assurance unit.	Perf: DR team leader. Approval: Not required		
5.5	Implementation of DR decisions	(1) Implementation of required corrections included in AIs list.	Perf: Project team. Approval: Project leader.	Corrected design documents	
		(2) Examination of corrections and approval by DR team member.	Perf: DR team member. Approval: Not required	(1) Approval of each correction. (2) Approval of completion of all corrections	

Prepared by: Dave Towers	QA engineer	Date: April 3, 2016	Signed: Dave Towers
Approved by: Barry Hotter	Head, QA unit	Date: May 2, 2016	Signed: Barry Hotter

What is the motivation behind updating procedures?

External changes

- Technological changes in development tools, hardware, communication equipment, etc.
- Changes in legal requirements
- Changes in the organisation's areas of activity

Procedural reasons

- Termination of a version lifetime, after which an update review is mandatory

Proposals for process improvements

- User and SQA team proposals for improvement
- Analysis of failures as well as successes
- Proposals for improvements initiated by internal audit reports
- Learning from the experience of other organisations

Preparation of New Procedures

Appointment of an ad hoc committee

The ad hoc committee is comprised of professionals working in the units involved, SQA unit members, and experts in the respective topics to be dealt with.

Assessment of the existing process

The assessment will be based on an observation of the process, documents review, study of quality problems, and client complaints. The findings will be summarized in a report.

Preparation of a procedure draft

The committee will prepare the proposed draft.

Review of the proposed draft

The draft should be reviewed by the leaders of the teams responsible to implement the procedure, the managers of the unit/department, and other persons involved in the procedure.

Preparation of the final draft for approval

A satisfactory final draft will be prepared by the committee, based on the comments and suggestions

Approval of the proposed procedure by the authorised person(s)

It should be noted that the approval is not automatic, and in a number of cases, the senior employee, serving as the authorising person, initiates further changes according to his/her comments.

Implementation of New or Revised Procedures

Distribution of the procedure material in a printed form, intranet site, and so on to ensure availability to team members and all persons involved in software development and maintenance.

Supporting/coaching the users in implementation by providing explanations and solving user difficulties in performing the required processes.

Follow-up of implementation in order to identify users, who do not perform correctly the new procedure, and instruct them as needed.

Contribution of procedures to software quality assurance

SQA procedures, when developed and maintained properly, are assumed to reflect the most adequate method known to date **for the performance of design and maintenance activities**.

SQA procedures that are up-to-date and fully implemented by developers and maintenance teams ensure **conformity of their activities to the software's quality requirements and performance of the associated activities in an efficient and effective way.**

Uniform development and maintenance enables **easier and more effective professional reviews** together with better communication with maintenance teams.

It **facilitates cooperation and coordination** between all bodies, internal and external, involved in the project.

Work Instructions

Work instructions **deal with the application of procedures**, adapted to the requirements of a specific project team, customer, or other relevant party.

General methodology is defined in a procedure, the precise details that allow its application to a specific project or unit are often laid out in a work procedure.

Work instructions should never contradict their parent procedure, although several instructions may be associated with another procedure.

Are specific to a team or department; they supplement procedures by providing explicit details that are suitable solely to the needs of one team, department, or unit.

Used mainly in cases when a uniform method of performing the task throughout the organisation is crucial to its success.

Example Types of Work Instructions

- audit process for subcontractors
- design documentation templates
- C++ programming instructions
- coordination and cooperation with the customer
- follow-up of beta site reporting
- monthly progress reporting

What should go in coding work instructions?

- **design change request procedure**
- **backup procedure**
- **version control**
- adherence to coding standards
  - comments
  - variable names
  - error checking
  - etc

What is the value of templates and checklists?

- **Templates**
  - documents are more complete
  - easier to review documents that are standardised
  - easier to find information

- **Checklists**
  - helps with self-checks
  - more efficient and effective reviews

Case Study

Software Solutions Inc. is a medium-sized software development company that has traditionally followed a waterfall model for all its projects. Recently, the management decided to transition to DevOps practices to improve efficiency and reduce time-to-market for their products. The change involves implementing new procedures and work instructions that affect various teams, including development, operations, and quality assurance.

The implementation plan includes the development of new work instructions for continuous integration and continuous deployment (CI/CD), automated testing protocols, and regular, smaller updates to systems. These instructions require significant changes in how teams collaborate and how tasks are accomplished.

1. What risks are associated with the preparation of new procedures and work instructions for adopting DevOps practices at Software Solutions Inc.?
2. How can Software Solutions Inc. mitigate the risk of implementation failure when introducing these new procedures?
3. What are the long-term benefits and potential drawbacks of transitioning to DevOps practices through these new procedures at Software Solutions Inc.?

Case Study

What risks are associated with the preparation of new procedures and work instructions for adopting DevOps practices at Software Solutions Inc.?

Resistance to Change: Employees may resist the shift from a well-understood waterfall model to DevOps, which could hinder the adoption of new practices.
Training Needs: There is a risk that the current workforce might lack the necessary skills for DevOps, necessitating extensive training and potentially leading to delays.
Integration Challenges: Integrating new tools and practices with existing systems could lead to technical difficulties and disruptions in current projects.
Documentation Overload: Inadequate or excessively complex documentation could make it difficult for teams to understand or follow the new instructions effectively.
Inexperience: Not knowing the DevOps methods in details, not enough familiarity in the domain.

How can Software Solutions Inc. mitigate the risk of implementation failure when introducing these new procedures?

Stakeholder Engagement: Engage with all stakeholders, including developers, testers, and operations teams, from the outset to ensure buy-in and gather input on the procedures.
Pilot Testing: Implement the new procedures in a controlled, pilot environment before a full-scale rollout. This allows for addressing potential issues in a smaller, manageable context.
Training Programs: Develop comprehensive training programs to equip all relevant employees with the necessary skills and knowledge to adapt to the new DevOps practices.
Incremental Implementation: Roll out the changes gradually rather than all at once, to minimize disruption and allow time for adjustment.
Expert Opinion: Support and consultancy from DevOps experts

What are the long-term benefits and potential drawbacks of transitioning to DevOps practices through these new procedures at Software Solutions Inc.?

Benefits:
Increased Efficiency: Continuous integration and deployment can lead to quicker releases and faster feedback from end users.
Better Product Quality: Frequent testing and updates can improve the quality of software as issues are identified and resolved quickly.
Enhanced Collaboration: Breaking down silos between departments can lead to better communication and more innovative solutions.
Drawbacks:
Ongoing Maintenance: The need for continuous monitoring and updating of CI/CD pipelines can increase the operational overhead.
Dependency on Tools: Heavy reliance on specific tools and platforms for DevOps practices can lead to challenges if those tools fail or become obsolete.

Software Change Process

Some Questions

- What is the correct version of the software module that I have to continue its coding?
- Who can provide me with an accurate copy of the last year’s version 4.1 of the software package?
- What version of the design document matches the software version we are adapting to a new customer?
- What version of the software system is installed at ABC Industries?
- What changes have been introduced in the version installed at the ABC Industries’ site?
- What changes have been introduced in the new version of the software?
- Where can I find the full list of customers that use version 6.8 of our software?
- Can we be sure that the version installed at Top Com Ltd. does not include undocumented changes?

- **Which is more important?:**
  - ○ **Stability or**
  - ○ **Progress**

- **Why is change potentially dangerous?**

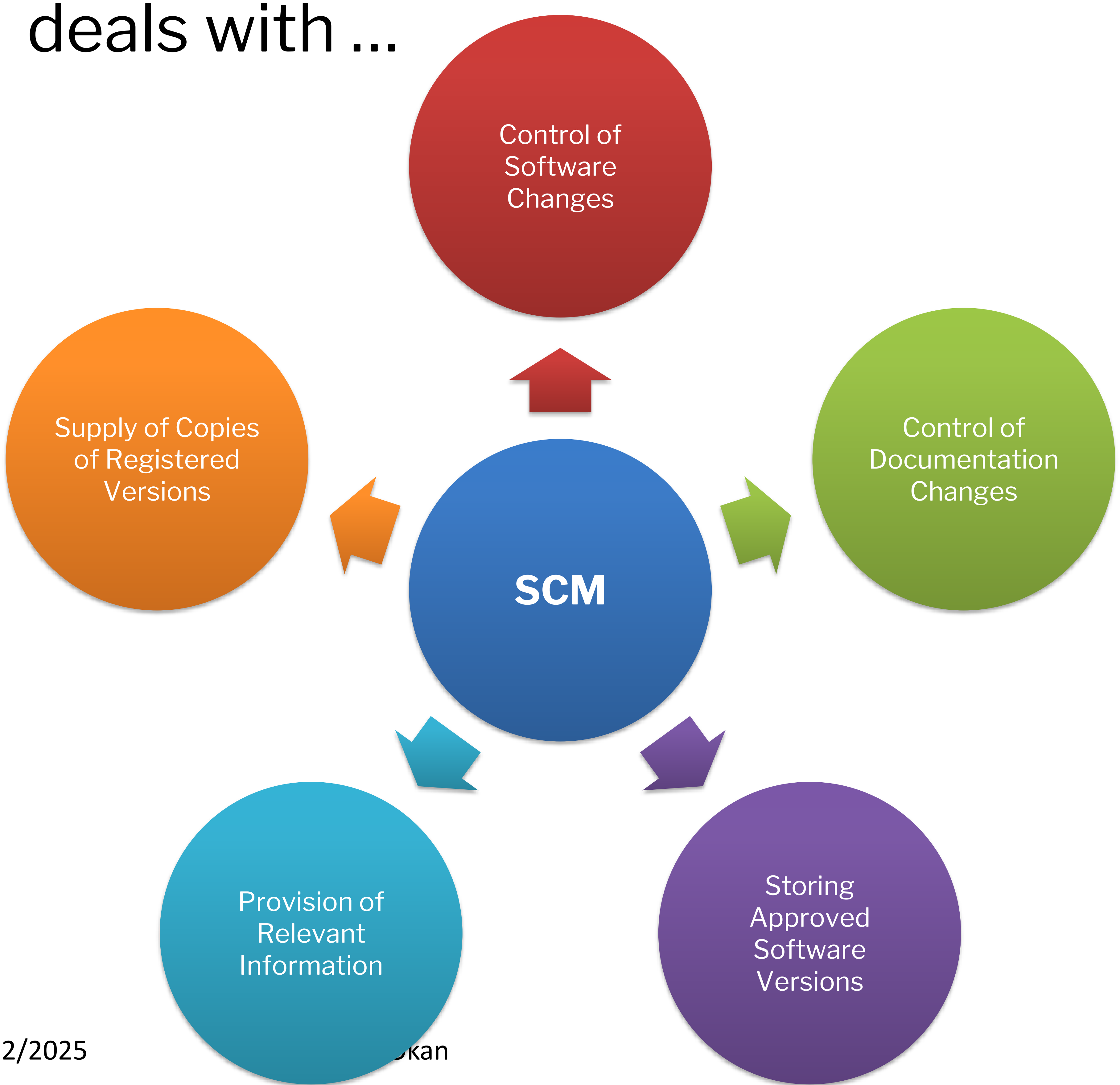
Change is inevitable or not?

What is there to change?

The need to cope with software changes throughout the software life cycle is one of the more important and difficult tasks of the software development and maintenance teams.

Performing changes usually under time constraints is one of the processes more susceptible to software errors.

Software Configuration Management (SCM) deals with ...


Control of Software Changes
Supply of Copies of Registered Versions
SCM
Control of Documentation Changes
Provision of Relevant Information
Storing Approved Software Versions

Aims to:

- Identify Change
- Control Change
- Ensure Change is properly implemented
- Report Changes to all the people who need to know

Software Configuration Management (SCM) Definitions

Software Configuration Item (SCI)

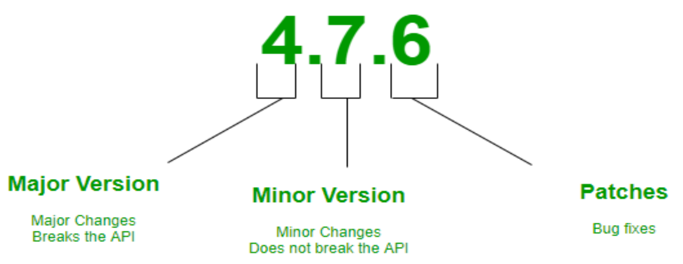
An approved unit of software code, a document or piece of hardware that is subjected to configuration management and treated as a distinct entity in the software configuration management process.

Software Configuration Item Version (SCI version)

The approved state of an SCI at any given point of time during the development or maintenance process.

Software Configuration Version

An approved selected set of documented SCI versions, that constitute a software system or document at a given point of time, where the activities to be performed are controlled by software configuration management procedures.

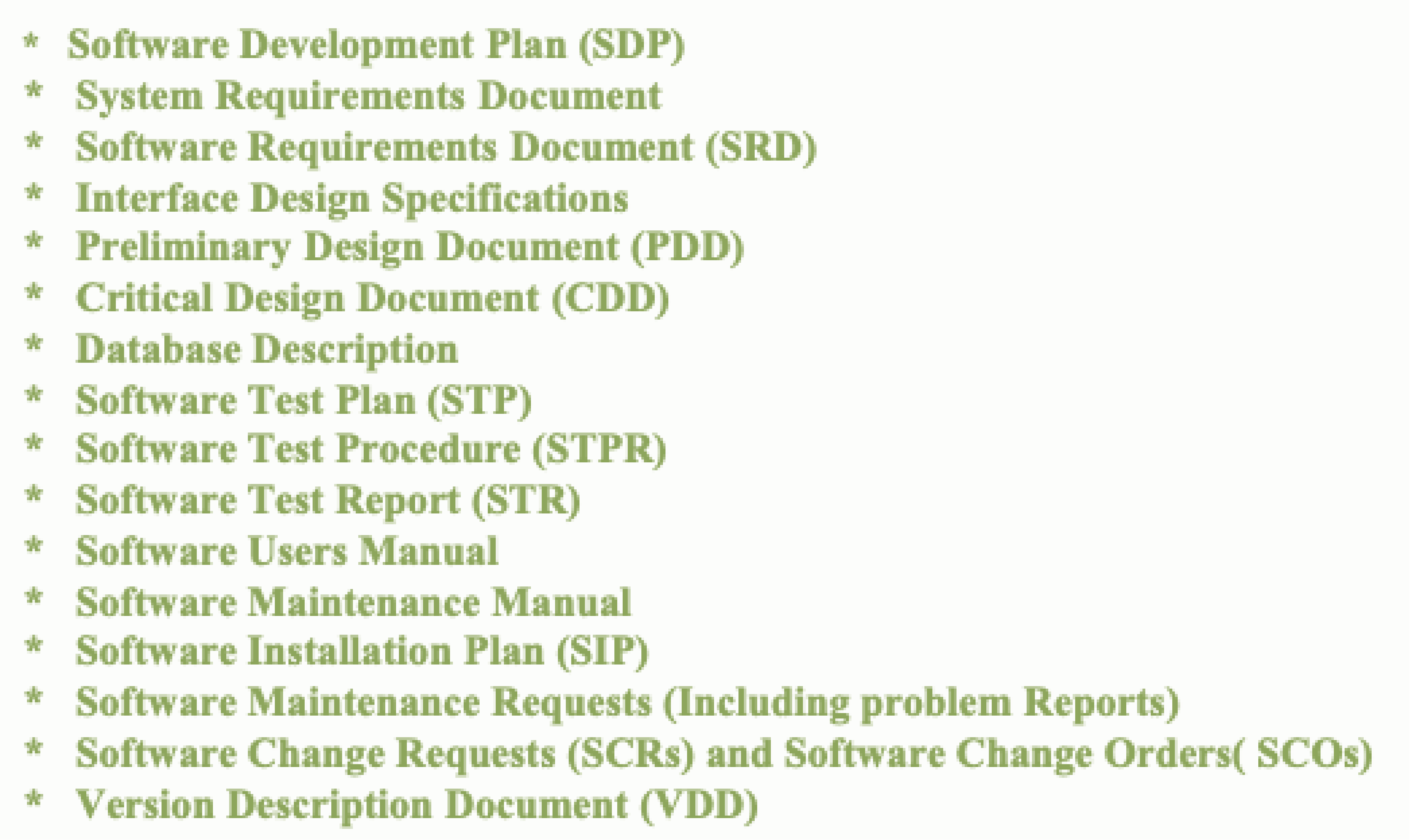


Baseline

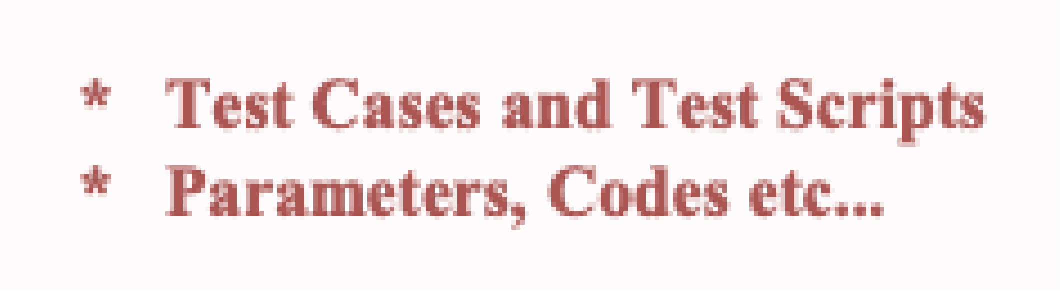
“A specification or product that has been formally reviewed and agreed upon, that there-after serves as the basis for further development, and that can be changed only through formal change control procedures."

IEEE Std 729 Standard Glossary of Software Engineering Terminology

Software Configuration Management (SCM)
Example Configuration Items

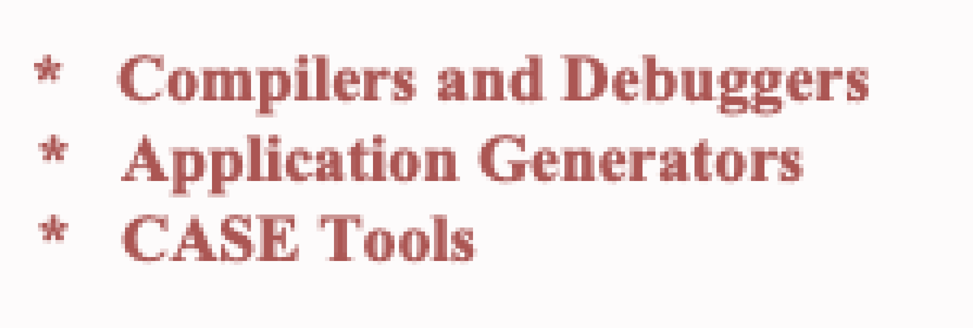


* Software Development Plan (SDP)
* System Requirements Document
* Software Requirements Document (SRD)
* Interface Design Specifications
* Preliminary Design Document (PDD)
* Critical Design Document (CDD)
* Database Description
* Software Test Plan (STP)
* Software Test Procedure (STPR)
* Software Test Report (STR)
* Software Users Manual
* Software Maintenance Manual
* Software Installation Plan (SIP)
* Software Maintenance Requests (Including problem Reports)
* Software Change Requests (SCRs) and Software Change Orders(SCOs)
* Version Description Document (VDD)



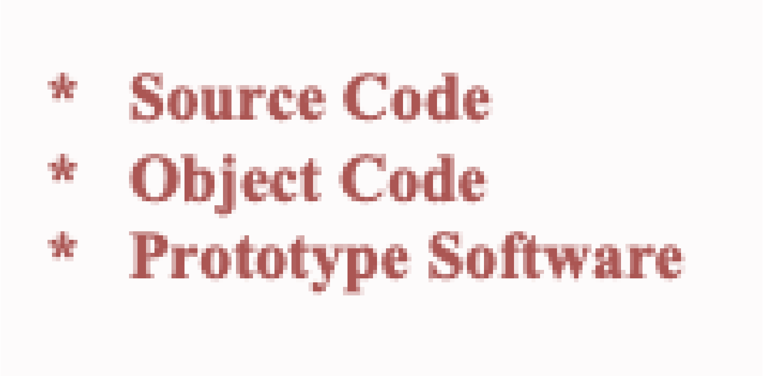
* Test Cases and Test Scripts
* Parameters, Codes etc...

Data Files



* Compilers and Debuggers
* Application Generators
* CASE Tools

Software Development Tools

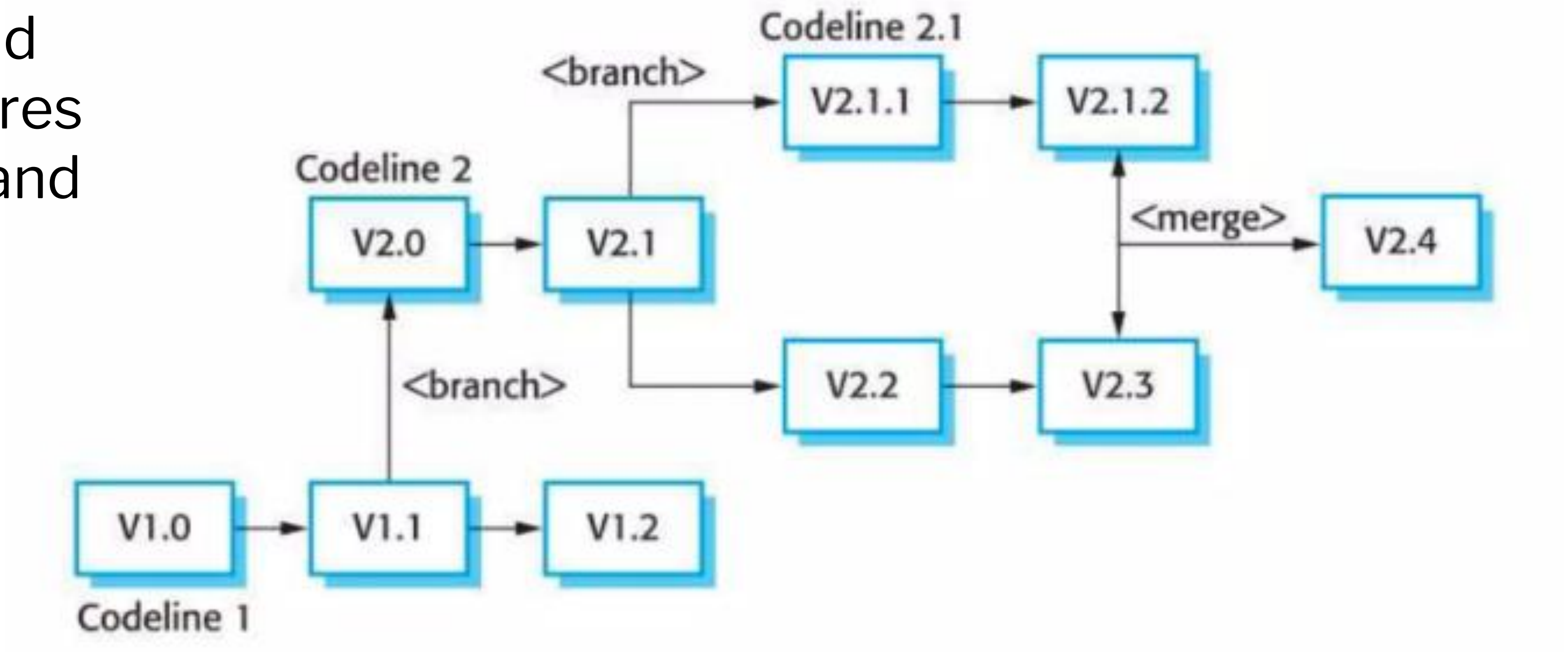


* Source Code
* Object Code
* Prototype Software

Software Code

What is SCM then?

An SQA component responsible for applying (computerised and non-computerised) technical tools and administrative procedures that enable completion of the tasks required to maintain SCIs and software configuration versions.



Software Change Control (SCC) process

1. Submission of Software Change Request (SCR) and examination of requested or suggested changes.
2. Approval of implementation of only those changes which are worthy and necessary, while all remaining are rejected.
3. Scheduling of the implementation of each of the approved changes.
4. Follow-up of the approved changes

THINK
CHECK
SUBMIT

1 - Submission of SCRs

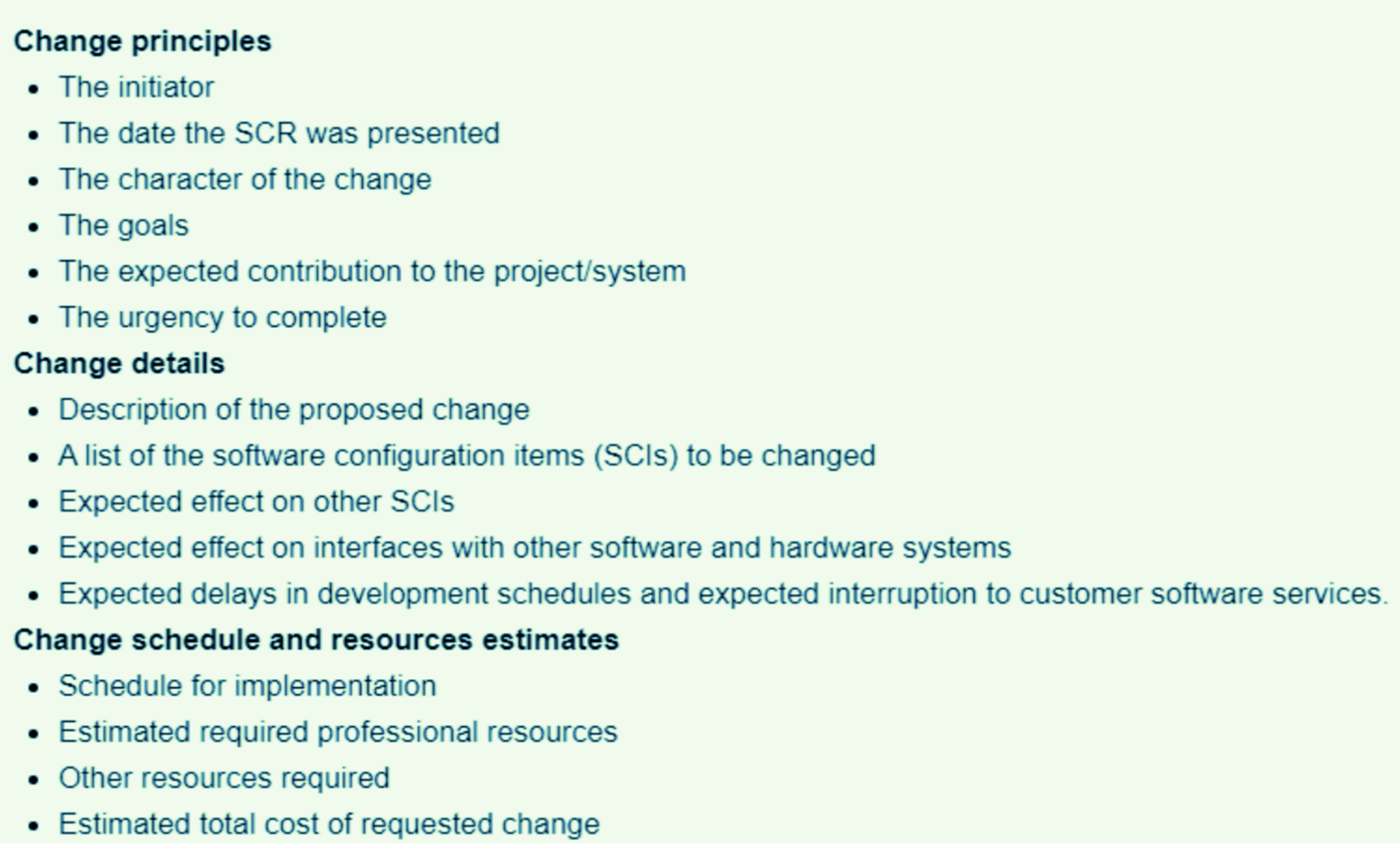
CR’s may relate to one or more of the following:

- A need to correct a software error.
- A need to adapt the software to changes in the operations of a customer's business or organisation missions.
- A need to adapt the software to a new customer's needs.
- A need to adapt the software to changes in general business and market changes.
- A proposal to update and improve the software product, to achieve higher customer satisfaction
- A need to adapt the software to budget and schedule constraints.

Initiatives to submit SCRs **in the software development stage** are mainly proposed by the customer and the developer.

In the software operational stage, it is the maintenance teams and customers (COTS software customers and custom-made software customers) who initiate.

SCR Document Template



Change principles

- The initiator
- The date the SCR was presented
- The character of the change
- The goals
- The expected contribution to the project/system
- The urgency to complete

Change details

- Description of the proposed change
- A list of the software configuration items (SCIs) to be changed
- Expected effect on other SCIs
- Expected effect on interfaces with other software and hardware systems
- Expected delays in development schedules and expected interruption to customer software services.

Change schedule and resources estimates

- Schedule for implementation
- Estimated required professional resources
- Other resources required
- Estimated total cost of requested change

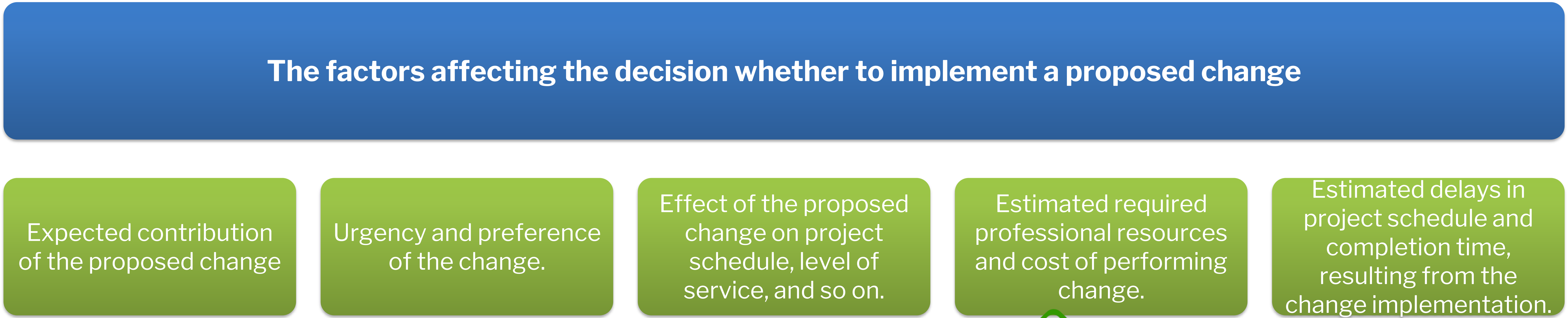
2 - Examination of the SCR by the SCC committee

The SCC committee performs an examination of each SCR by professionals.

The SCR evaluation report refers to the various aspect of the request, and serves as a basis for the SCC committee's decision

3 - Approval to carry out requested changes by the SCC committee

Decisions by SCC committee after examination of each SCR



The factors affecting the decision whether to implement a proposed change

- Expected contribution of the proposed change
- Urgency and preference of the change.
- Effect of the proposed change on project schedule, level of service, and so on.
- Estimated required professional resources and cost of performing change.
- Estimated delays in project schedule and completion time, resulting from the change implementation.

The SCC committee may approve, delay, or deny the request for immediate implementation.

APPROVED

For each SCR approved for immediate implementation, a **Software Change Order (SCO)** is issued (also called change orders (CO) or engineering change orders (ECO))

It provides the change details, its costs and schedule, which may differ from the original estimates

4 - Follow-up of software change processes by the SCC committee

The SCC committee follows-up implementation of the SCQs regarding

- conformance to the project schedule,
- successful completion of change activities, and
- the actual resources utilized for the SCO.

The function of SCC in the Organisation

- The responsibility for implementation of the SCC tasks in software development and maintenance organisations is usually assigned to professional committee appointed by the management.
- Additional ad hoc members include the project manager, the customer representative, and experts for specific subjects.

This committee is

- commonly called the software change control authority **(SCCA)** or the software change control board **(SCCB)**, and
- also frequently known as the change control authority **(CCA)** or change control board **(CCB)**

- In some cases, the SCC function is performed by the software configuration management board (SCMB), who appoints an SCC committee. The operation of SCC activities is supported by SCC procedures **defining the process of SCR submission, and the activities of the SCC committee.**

During the software development phase, when the development is carried out according to a contract, the **main considerations of the SCCA committee are the cost of changes and schedule delays.**

During the operational phase, the SCCA decisions, typical **to maintenance, are the continuity of regular service, in addition to cost of changes**. In organisations developing COTS software, it is expected that decisions regarding SCRs are strongly affected by merchantability considerations.

SQA Activities Related to SCC

- Review of change control procedures
- Initiation of changes and improvement revisions of SCC procedures
- Review of the SCC committee's performance: compliance with procedures, completeness of SCR examinations by the committee, and delays in decision making by the committee.
- Review of SCC decisions regarding approvals and rejections, approved additional resources, project completion delays and more
- Auditing of the implementation of SCOs; on schedule and implementation processes of the approved changes.

CM Audits

- % of unapproved changes
- % of Change Orders completed on schedule
- % of affected Configuration Items that were not checked
- % of properly documented Configuration Items
- number of CM Process Failures

How do we keep track of all these versions, dependencies among components, approval records, etc. etc. etc., and still assure quality?

- Have a Sound CM **Procedure**
- Use Good CM **Tools**

CM Plan - Format

According to IEEE Standard 828 - standard for Software Configuration Management Plans

1. Introduction

a) purpose b) scope c) definitions and acronyms d) references

2. Management

a) organisation

b) SCM responsibilities

c) interface control

d) SCMP implementation

e) policies, directives, procedures (naming conventions, version designations, problem report process)

3. SCM Activities

a) configuration identification

b) configuration control (change history, review authority, read/write control, member identification)

c) configuration status accounting (status of requests, status of approved changes, …)

d) audits and reviews

4. Tools, Techniques, and Methodologies

5. Supplier Control

6. Records Collection and Retention

IEEE 1042 Guide to Software Configuration Management

Defines **terms**, such as baseline and version

Discusses configuration management as a management discipline and its **role** in the engineering process

Includes **checklists** of issues for sections of the SCMP (IEEE Std 828)

Includes four complete **examples** of SCMPs

CM Tools - Necessary Features

- Dependency Tracking!!!
- Audit Trails!!!
- Reporting of Changes
- Supports the Change Rules
- Versioning
- Requirements Tracing
- Repository arranged as "basic objects" and "aggregate objects"
- Supports both Linear evolution and Trees

Discussion Case

BetaTech Software Solutions is a growing software development company that specialises in creating solutions for the healthcare industry. The company is currently working on a major project to develop a new patient management system designed to integrate with existing hospital information systems to improve patient care and administrative efficiency.

The project has entered its final year, but several significant issues have emerged that have led to delays and increased costs. To address these issues, BetaTech's management is considering a comprehensive review of their current software change control procedures. The project has experienced problems such as frequent unplanned changes, delays in approvals, and difficulties in implementing approved changes efficiently.

1. What are the potential benefits and drawbacks of conducting a thorough review of the existing change control procedures?
2. How might initiating changes and improvements to SCC procedures affect the project’s outcome?
3. What are the implications of reviewing the SCC committee's performance on the overall project management and change control process?
4. Evaluate the potential impacts of carefully reviewing past SCC decisions on future project management strategies.