What is software quality?

- The International Organization for Standardization (ISO) defines **quality** as "the degree to which a set of inherent characteristics fulfils requirements" (ISO9000:2000).
- Other experts define quality based on:
  - **Conformance to requirements**: The project's processes and products meet written specifications.
  - **Fitness for use**: A product can be used as it was intended.
- Quality, simplistically, means that a product should meet its specification.
- This is problematical for software systems
  - There is a tension between customer quality requirements (efficiency, reliability, etc.) and developer quality requirements (maintainability, reusability, etc.);
  - Some quality requirements are difficult to specify in an unambiguous way;
  - Software specifications are usually incomplete and often inconsistent.

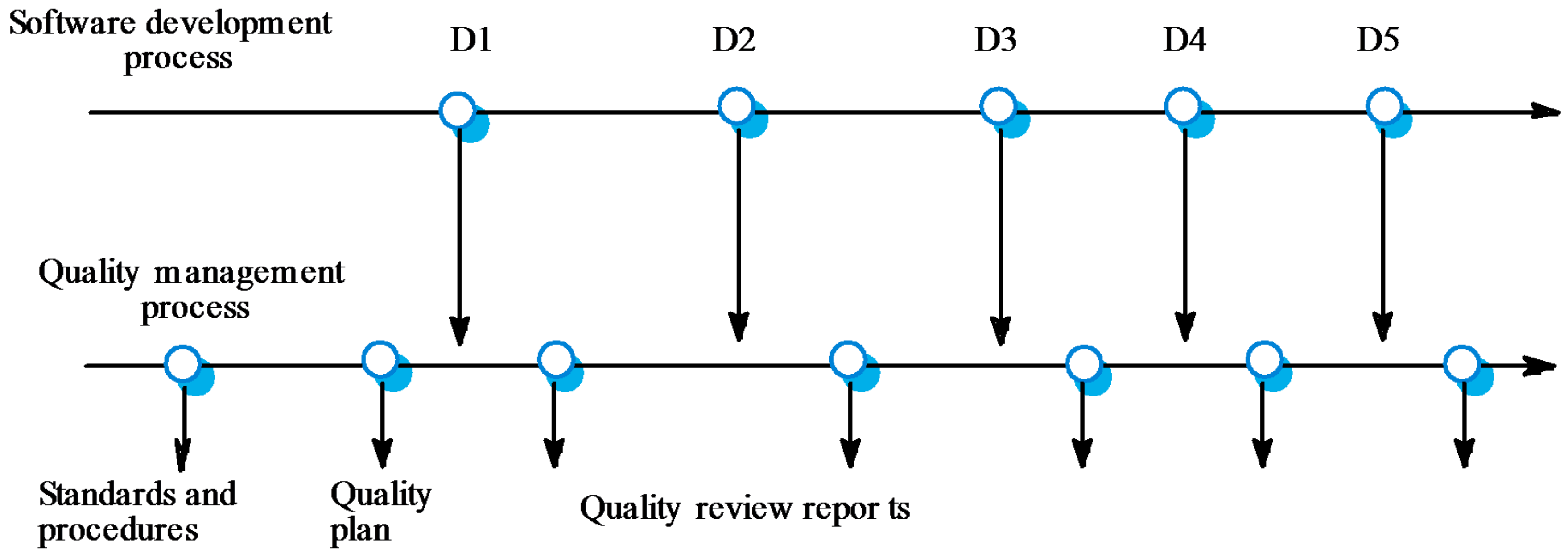
Software quality management

- Concerned with ensuring that the required level of quality is achieved in a software product.
- Involves defining appropriate quality standards and procedures and ensuring that these are followed.
- Should aim to develop a 'quality culture' where quality is seen as everyone's responsibility.
- Establish organisational procedures and standards for quality.
- Project quality management ensures that the project will satisfy the needs for which it was undertaken.

Quality management activities

- Processes include:
  - **Quality planning**: Identifying which quality standards are relevant to the project and how to satisfy them. Select applicable procedures and standards for a particular project and modify these as required.
  - **Quality assurance**: Periodically evaluating overall project performance to ensure the project will satisfy the relevant quality standards.
  - **Quality control**: Monitoring specific project results to ensure that they comply with the relevant quality standards. Ensure that procedures and standards are followed by the software development team.

Quality management and software development



Software Quality Requirement

- A condition or capacity that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
- A condition or capacity needed by a user to solve a problem or achieve a quality objective.
- Requirements constitute a complete statement of what the software will do, without referring to how it will do it.
- Requirements are separated into functional or behavioural requirements and non-functional requirements (NFRs).

Requirements and Software Requirements Specifications (SRS)

- Functional requirements are the services provided for the user and usually make up the bulk of the requirements specification document.
- Non functional requirements are the constraints imposed on the software, and the restrictions put on the software development project. They include reliability, security, performance, usability, availability and portability constraints.
- The SRS should specify the external behaviour of the software as well as the constraints on software development.

Measuring Software Quality

- Software Reliability
  - Mean time to failure
  - Probability of unavailability
  - Rate of failure occurrence
  - Availability
- Robustness and Portability
  - Time to restart failures
  - Percentage of events causing failures
  - Probability of data corruption on failure
  - Percentage of target-dependent statements
  - Number of target systems

Working environments that affect Software Quality

- The physical workplace provision has an important effect on individual productivity and satisfaction
  - Comfort;
  - Privacy;
  - Facilities.
- Health and safety considerations must be taken into account
  - Lighting;
  - Heating;
  - Furniture.

Factors influencing group working

- Group composition (task-oriented, self-oriented, interaction-oriented).
- Group cohesiveness.
- Group communications.
- Group organisation.
- People management involves:
  - Consistency
    - Team members should all be treated in a comparable way without favourites or discrimination.
  - Respect
    - Different team members have different skills and these differences should be respected.
  - Inclusion
    - Involve all team members and make sure that people's views are considered.
  - Honesty
    - You should always be honest about what is going well and what is going badly in a project.

Group cohesiveness

- In a cohesive group, members consider the group to be more important than any individual in it.
- The advantages of a cohesive group are:
  - Group quality standards can be developed;
  - Group members work closely together so inhibitions caused by ignorance are reduced;
  - Team members learn from each other and get to know each other's work;
  - Egoless programming where members strive to improve each other's programs can be practised.
- Cohesiveness is influenced by factors such as the organisational culture and the personalities in the group.

Group Dynamics

- Information must be exchanged on the status of work, design decisions and changes to previous decisions.
- Good communications also strengthens group cohesion as it promotes understanding.
- Software development groups should be small and cohesive. Leaders should be competent and should have administrative and technical support.
- Group communications are affected by status, group size, group organisation and personality composition of the group
- Working environments should include spaces for interaction and spaces for private working.

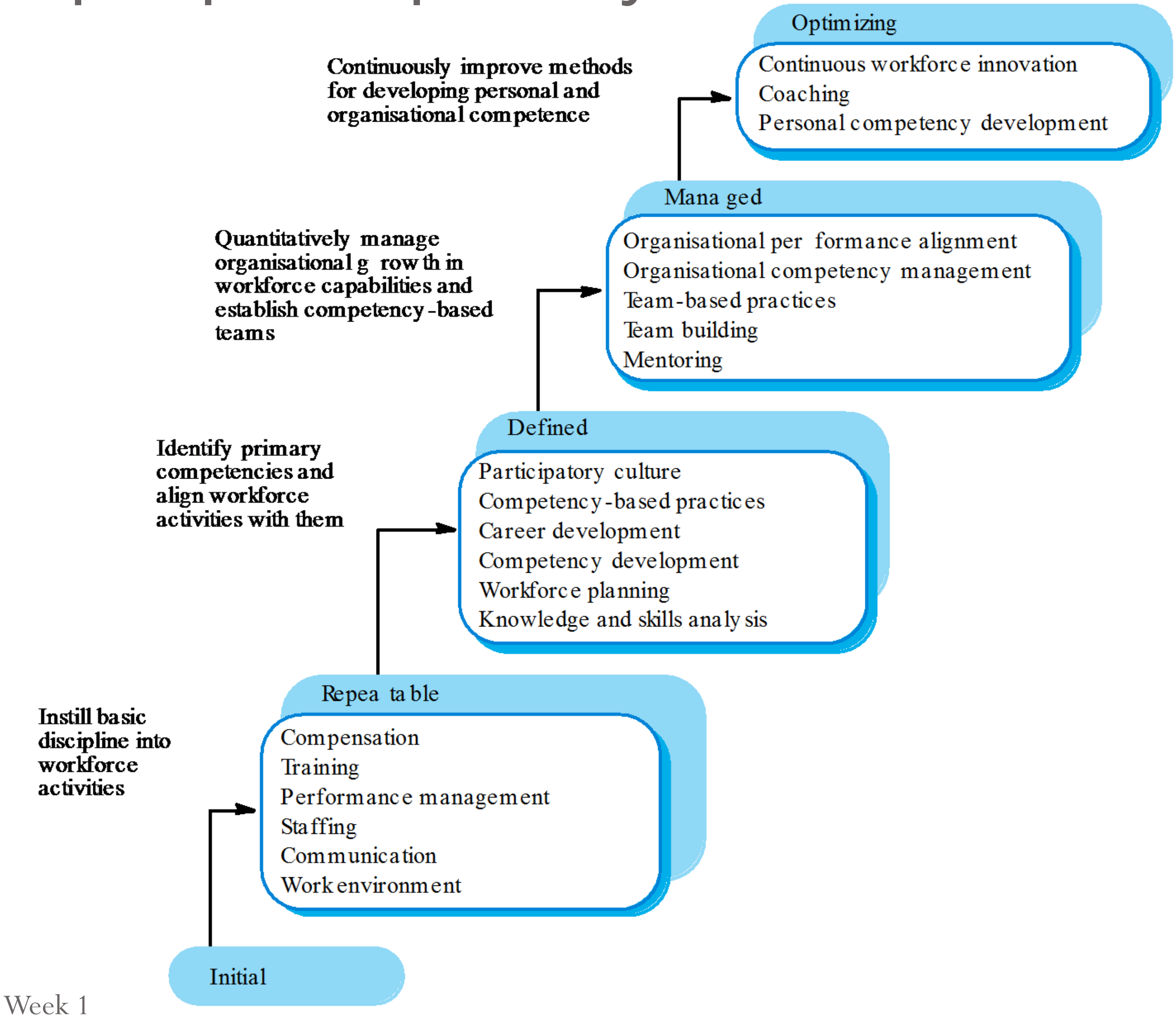
The People Capability Maturity Model

- Intended as a framework for managing the development of people involved in software development.
- P-CMM Objectives:
  - To improve organisational capability by improving workforce capability.
  - To ensure that software development capability is not reliant on a small number of individuals.
  - To align the motivation of individuals with that of the organisation.
  - To help retain people with critical knowledge and skills.

P-CMM levels

- Five stage model
  - Initial. Ad-hoc people management
  - Repeatable. Policies developed for capability improvement
  - Defined. Standardised people management across the organisation
  - Managed. Quantitative goals for people management in place
  - Optimizing. Continuous focus on improving individual competence and workforce motivation

The people capability model



Optimizing

Continuous workforce innovation
Coaching
Personal competency development

Continuously improve methods for developing personal and organisational competence

Managed

Organisational performance alignment
Organisational competency management
Team-based practices
Team building
Mentoring

Quantitatively manage organisational growth in workforce capabilities and establish competency-based teams

Defined

Participatory culture
Competency-based practices
Career development
Competency development
Workforce planning
Knowledge and skills analysis

Identify primary competencies and align workforce activities with them

Repeatable

Compensation
Training
Performance management
Staffing
Communication
Work environment

Instill basic discipline into workforce activities

Initial

Verifications vs Validation, The Cost of Quality

- Verification: "Are we building the product right”.
- The software should conform to its specification.
- Validation: "Are we building the right product”.
- The software should do what the user really requires
- The **cost of quality** is the cost of conformance plus the cost of nonconformance.
  - **Conformance** means delivering products that meet requirements and fitness for use.
  - **Cost of nonconformance** means taking responsibility for failures or not meeting quality expectations.

Five Cost Categories Related to Quality

- **Prevention cost**: Cost of planning and executing a project so it is error-free or within an acceptable error range.
- **Appraisal cost**: Cost of evaluating processes and their outputs to ensure quality.
- **Internal failure cost**: Cost incurred to correct an identified defect before the customer receives the product.
- **External failure cost**: Cost that relates to all errors not detected and corrected before delivery to the customer.
- **Measurement and test equipment costs**: Capital cost of equipment used to perform prevention and appraisal activities.