Chapter : Software Process

Chapter - Topic Covered

- Layered Technology
- Software Process Framework
- Generic Process Framework Activities
- Umbrella Activities
- CMMI Level

Overview

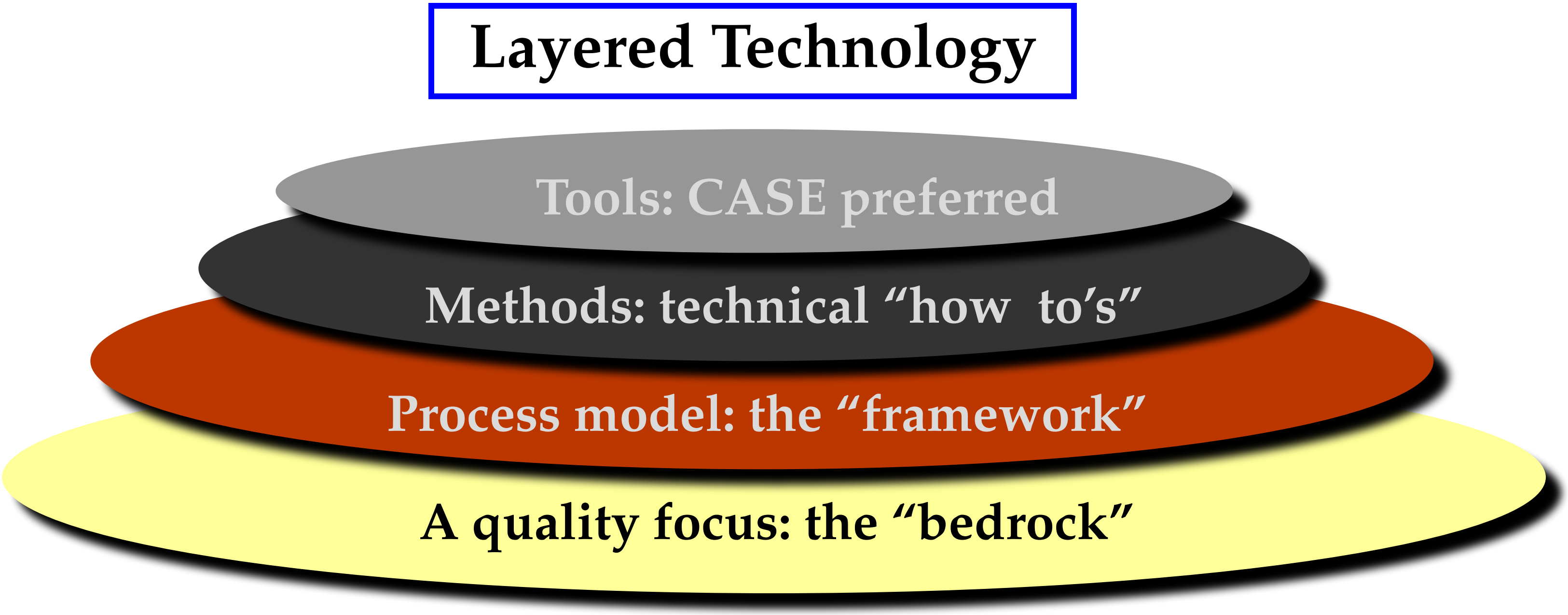
- **What? A software process** – as a framework for the tasks that are required to build high-quality software.
- **Who?** Managers, software engineers, and customers.
- **Why?** Provides stability, control, and organization to an otherwise chaotic activity.
- **Steps?** A handful of activities are common to all software processes, details vary.
- **Work product?** Programs, documents, and data.

What is software engineering?

- **Definition** :
  - (1) The application of systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.(2) The study of approaches as in (1) above
- Its a discipline that is concerned with all aspects of software production.

- Software engineers should adopt
  - Systematic and organized approach to their work
  - Use appropriate tools and techniques depending on the problem to be solved
  - The development constraints and the resources available
- Apply Engineering Concepts to developing Software
- Challenge for Software Engineers is to produce high quality software with finite amount of resources & within a predicted schedule

Software Engineering – Layered Technology



Layered Technology

Tools: CASE preferred

Methods: technical “how to’s”

Process model: the “framework”

A quality focus: the “bedrock”

Layered Technology

A quality Focus

- Every organization is rest on its commitment to quality.
- Total quality management, Six Sigma, or similar continuous improvement culture and it is this culture ultimately leads to development of increasingly more effective approaches to software engineering.
- The bedrock that supports software engineering is a quality focus.

Process:

- It’s a foundation layer for software engineering.
- It’s define **framework** for a set of *key process areas* (KRA) for effectively manage and deliver quality software in a cost effective manner
- The processes define the tasks to be performed and the order in which they are to be performed

Layered Technology

Methods:

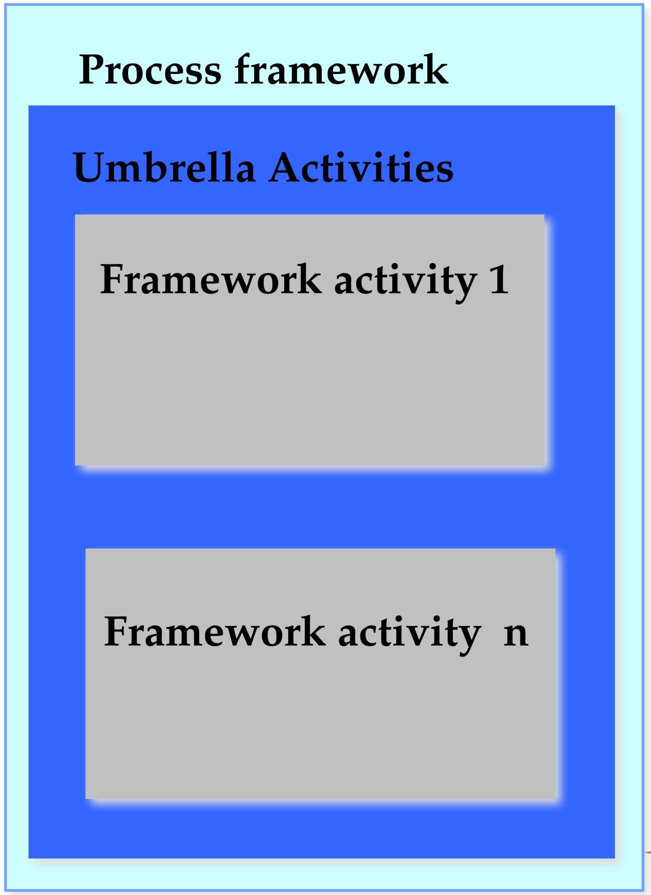
- It provide the technical **how-to's** for building software.
- Methods encompass a broad array of tasks that include requirements analysis, design, program construction, testing, and support.
- There could be more than one technique to perform a task and different techniques could be used in different situations.

Tools:

- Provide automated or semi-automated support for the process, methods and quality control.
- When tools are integrated so that information created by one tool can be used by another, a system for the support of software development, called ***computer-aided software engineering (CASE)***

Process Framework

Software Process



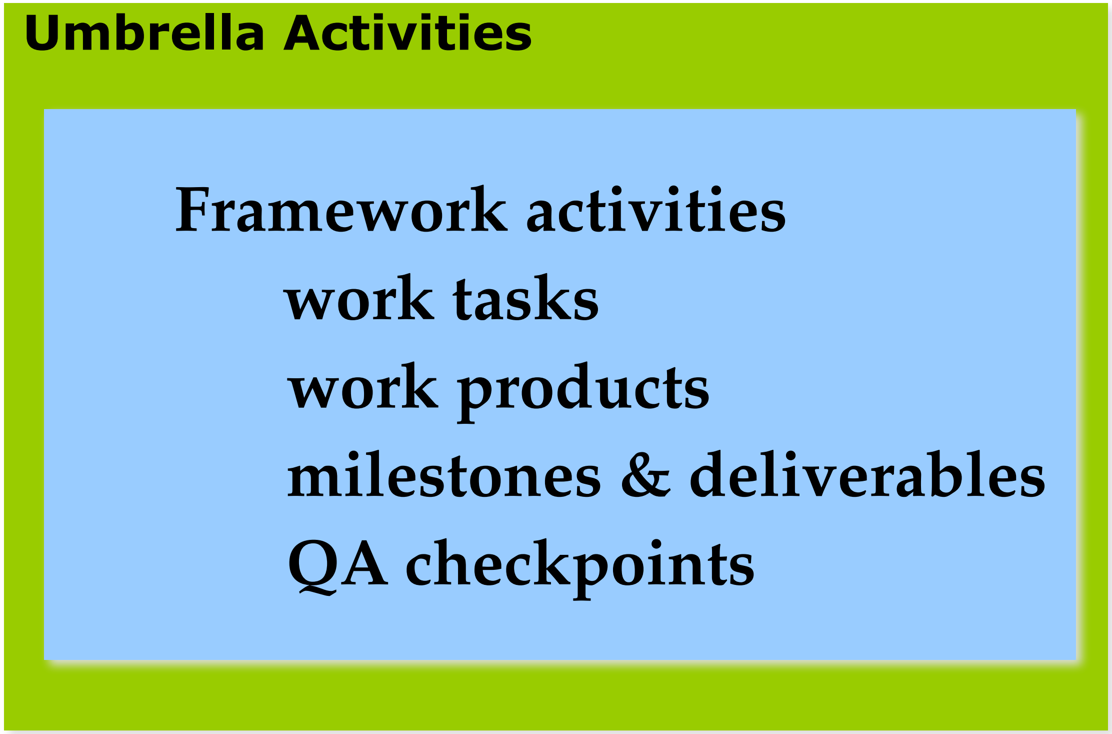
Process framework

Umbrella Activities

Framework activity 1

Framework activity n

Process Framework



Umbrella Activities

Framework activities

work tasks

work products

milestones & deliverables

QA checkpoints

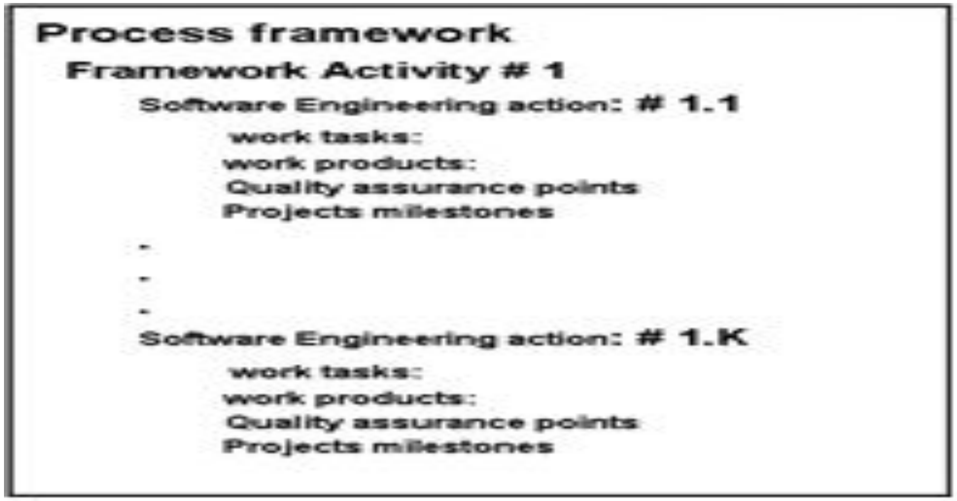
Process framework

Why process :

A process defines who is doing what, when and how to reach a certain goal.

- To build complete software process.
- Identified a small number of framework activities that are applicable to all software projects, regardless of their size or complexity.
- It encompasses a set of umbrella activities that are applicable across the entire software process.

Process Framework



Process framework

Framework Activity # 1

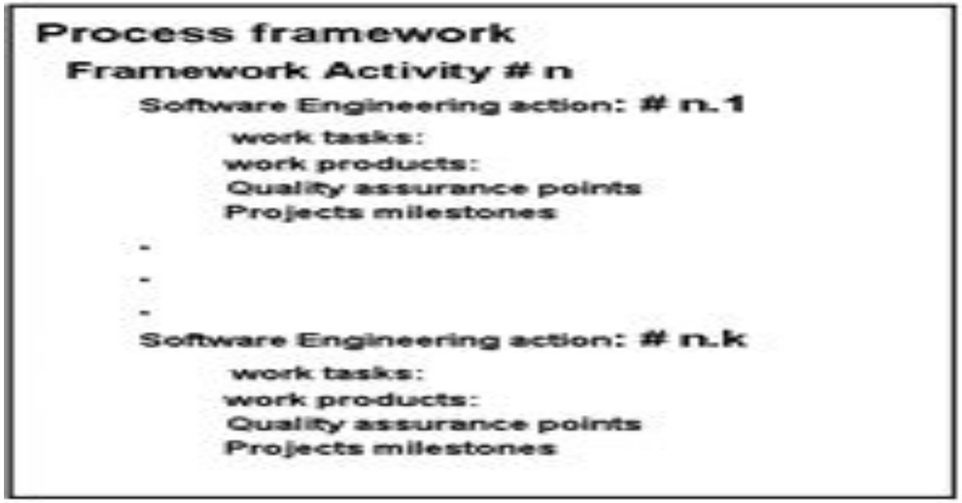
Software Engineering action: # 1.1

work tasks:
work products:
Quality assurance points
Projects milestones

-
-
-

Software Engineering action: # 1.K

work tasks:
work products:
Quality assurance points
Projects milestones



Process framework

Framework Activity # n

Software Engineering action: # n.1

work tasks:
work products:
Quality assurance points
Projects milestones

-
-
-

Software Engineering action: # n.k

work tasks:
work products:
Quality assurance points
Projects milestones

- Each framework activities is populated by a set for *software engineering actions* – a collection of related tasks.
- Each action has individual *work task*.

Generic Process Framework Activities

- Communication:
  - Heavy communication with customers, stakeholders, team
  - Encompasses requirements gathering and related activities
- Planning:
  - Workflow that is to follow
  - Describe technical task, likely risk, resources will require, work products to be produced and a work schedule.
- Modeling:
  - Help developer and customer to understand requirements (Analysis of requirements) & Design of software
- Construction
  - Code generation: either manual or automated or both
  - Testing – to uncover error in the code.
- Deployment:
  - Delivery to the customer for evaluation
  - Customer provide feedback

The Process Model: Adaptability

- The framework activities will always be applied on every project ... BUT
- The tasks for each activity will vary based on:
  - The type of project (an "entry point" to the model)
  - Characteristics of the project
  - Common sense judgment; concurrence of the project team

Umbrella Activities

- Software project tracking and control
  - Assessing progress against the project plan.
  - Take adequate action to maintain schedule.
- Formal technical reviews
  - Assessing software work products in an effort to uncover and remove errors before goes into next action or activity.
- Software quality assurance
  - Define and conducts the activities required to ensure software quality.
- Software configuration management
  - Manages the effects of change.
- Document preparation and production
  - Help to create work products such as models, documents, logs, form and list.
- Reusability management
  - Define criteria for work product reuse
  - Mechanisms to achieve reusable components.
- Measurement
  - Define and collects process, project, and product measures
  - Assist the team in delivering software that meets customer’s needs.
- Risk management
  - Assesses risks that may effect that outcome of project or quality of product (i.e. software)

Capability Maturity Model Integration (CMMI)

- The Software Engineering Institute (SEI) has developed process meta-model to measure organization different level of process capability and maturity.
- CMMI – developed by SEI
- The CMMI defines each process area in terms of “specific goals” and the “specific practices” required to achieve these goals.
- ***Specific goals*** establish the characteristics that must exist if the activities implied by a process area are to be effective.
- ***Specific practices*** refine a goal into a set of process-related activities.

CMMI Level

Level 0 (Incomplete) –

- Process are not perform or not achieve all the goals and objectives defined by the CMMI for Level I capability.

Level 1 (Performed) – All specific goals are performed as per defined by CMMI

Level 2 (Managed) –

- All level 1 criteria have been satisfied
- In addition to Level I;
  - People doing work have access to adequate resources to get job done,
  - Stakeholders are actively involved,
  - Work tasks and products are monitored, controlled, reviewed, and evaluated for conformance to process description.

Level 3 (Defined) –

- All level 2 criteria have been achieved.
- In addition;
  - management and engineering processes documented
  - standardized and integrated into organization-wide software process

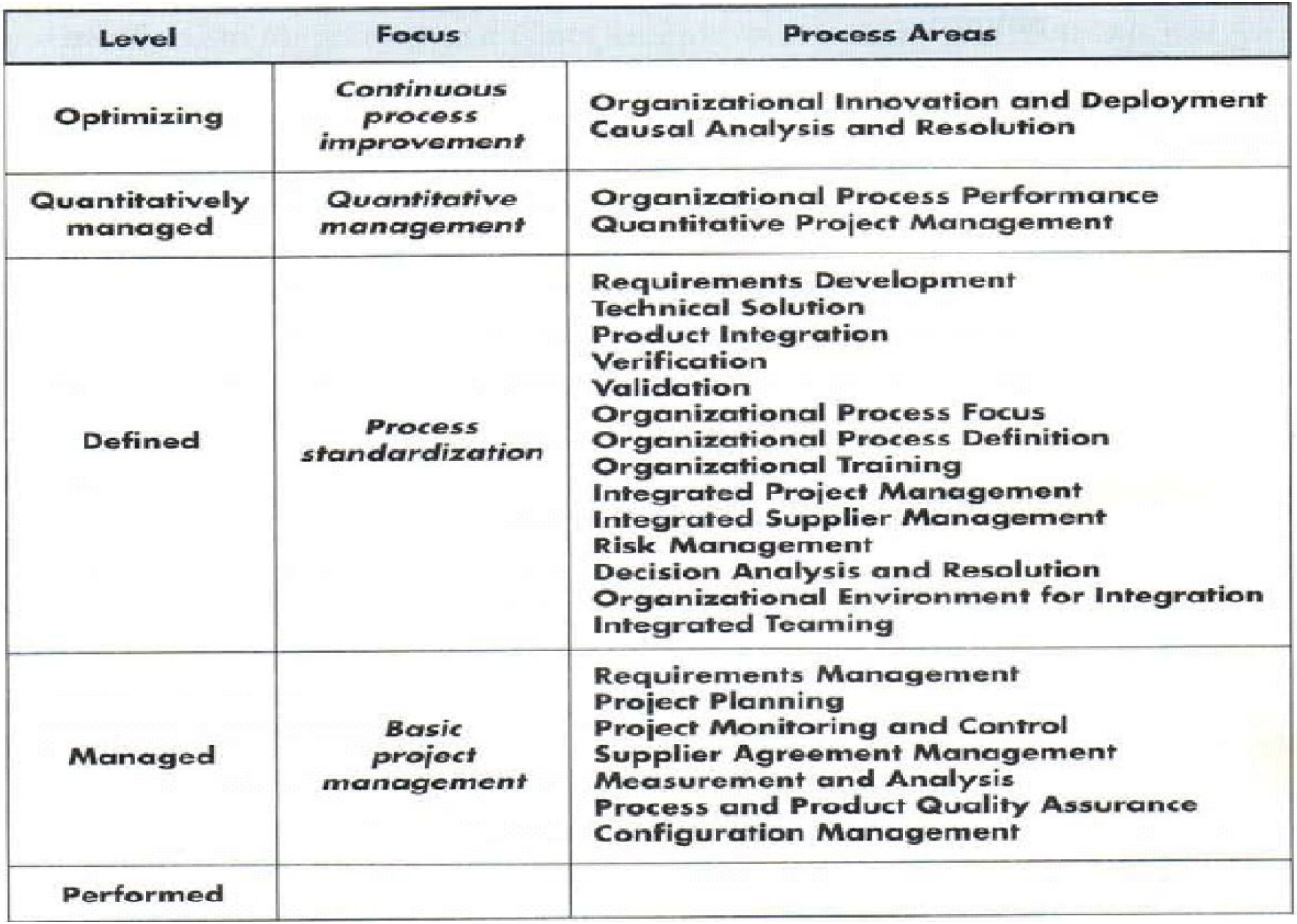
CMMI Level (cont.)

Level 4 (Quantitatively Managed) -

- All level 3 criteria have been satisfied.
- Software process and products are quantitatively understood
- Controlled using detailed measures and assessment.

Level 5 (Optimized) –

- Continuous process improvement is enabled by quantitative feedback from the process and testing innovative ideas.



Level	Focus	Process Areas
Optimizing	*Continuous process improvement*	Organizational Innovation and Deployment Causal Analysis and Resolution
Quantitatively managed	*Quantitative management*	Organizational Process Performance Quantitative Project Management
Defined	*Process standardization*	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Project Management Integrated Supplier Management Risk Management Decision Analysis and Resolution Organizational Environment for Integration Integrated Teaming
Managed	*Basic project management*	Requirements Management Project Planning Project Monitoring and Control Supplier Agreement Management Measurement and Analysis Process and Product Quality Assurance Configuration Management
Performed		