

Software process frameworks: ISO/IEC 12207, CMMI, and related

Jaak Tepandi

Tallinn University of Technology

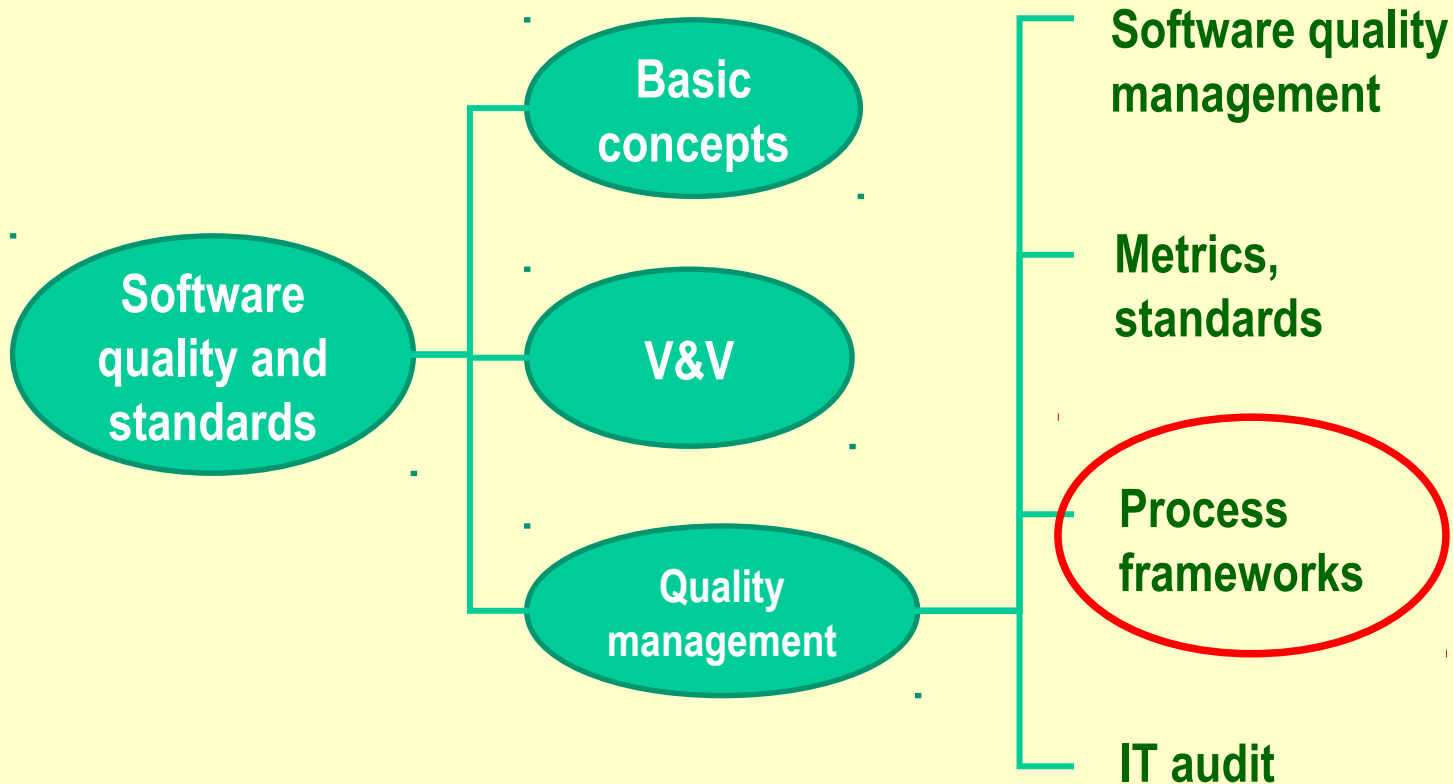
Institute of Informatics

Moodle: „Software Quality (Tarkvara kvaliteet)”

Alternate download: tepandi.ee

Version 14.12.2016

Part 3: Context and content



Topics for today (and more)



Is VTV sufficient? TPI? ISO 9001?

Today...

- ISO/IEC 12207 and related
- CMMI / ISO/IEC 15504 family / ISO/IEC 33000 family

... and there are more

- ISKE / IT Baseline Protection / ISO/IEC 27000 family
- ITIL / ISO/IEC 20000 family
- COBIT
- ...

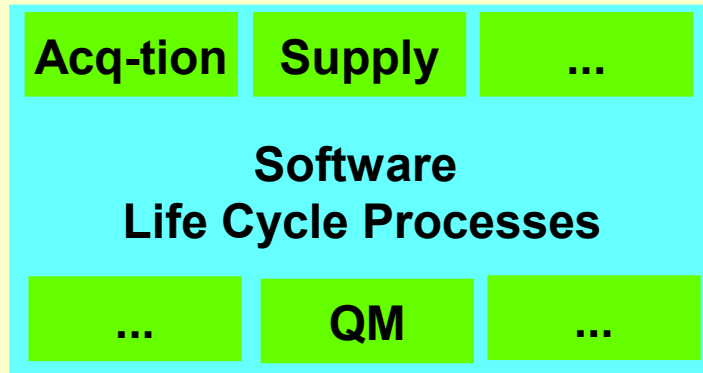
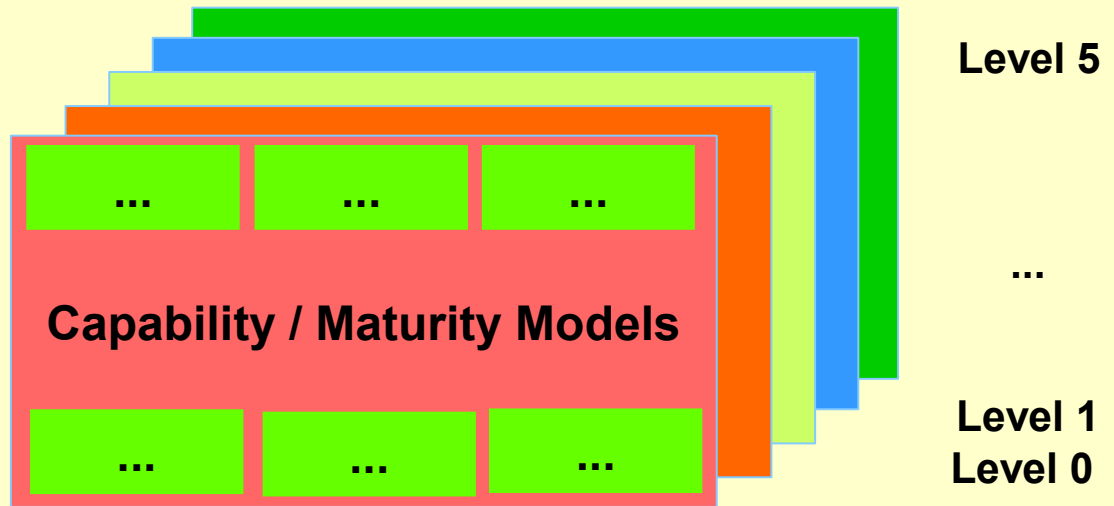
Outcomes

- **Understanding of the purpose and application areas of ISO 9000 family, ISO/IEC 12207, and capability / maturity models**
- **Understanding of the basic concepts and structure of these frameworks**
- **Given an organisation and a system - ability to choose, and to provide a rudimentary justification for the choice of a process model**
- **Understanding of the main activities needed to start selection and application of these frameworks and standards**
- **Ability to find needed information in these process models**

Useful to IT manager, programmer, maintainer, and other stakeholders involved in various software related processes

Broad view

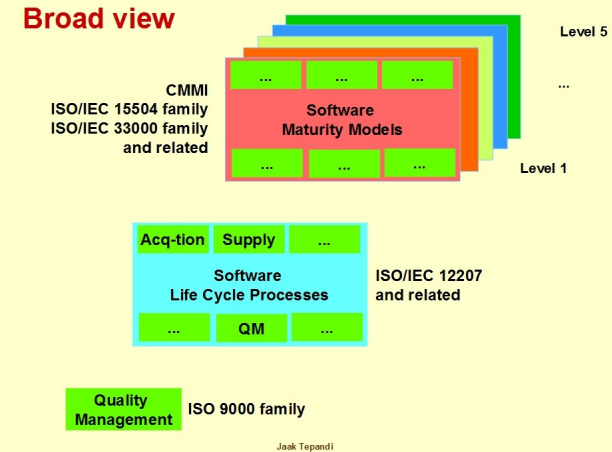
CMMI
ISO/IEC 15504 family
ISO/IEC 33000 family
TPI Next, TMMI
and related



ISO/IEC 12207
and related

Quality Management **ISO 9000 family**

ISO/IEC 12207



- Standards, structure. Software and system
- Processes
- Case studies
- Life cycle models and ISO/IEC 12207
- Implementation

ISO/IEC 12207 and related standards

**ISO/IEC 12207:2008 Systems and software engineering –
Software life cycle processes**

**ISO/IEC TR 24748-3:2011 Systems and software engineering
-- Life cycle management -- Part 3: Guide to the application
of ISO/IEC 12207**

System:

- **business processes, computer based processes**
- **hardware, software, facilities**
- **manual operations**

Purpose of ISO/IEC 12207

- To provide a defined set of processes to facilitate communication among acquirers, suppliers and other stakeholders in the life cycle of a software product
- For acquirers of systems and software products and services and for suppliers, developers, operators, maintainers, managers, quality assurance managers, and users of software products
- For use in a two-party situation (may be from the same organization)
- May be used by a single party through a self-imposed set of processes

ISO/IEC 12207:2008 - Categories of Life Cycle Processes

System context processes

- a) Agreement Processes — 2
- b) Organizational Project-Enabling Processes — 5
- c) Project Processes — 7
- d) Technical Processes — 11

Software specific processes

- e) Software Implementation Processes — 7
- f) Software Support Processes — 8
- g) Software Reuse Processes — 3

Categories...

Processes...

Activities...

Tasks...

Notes...

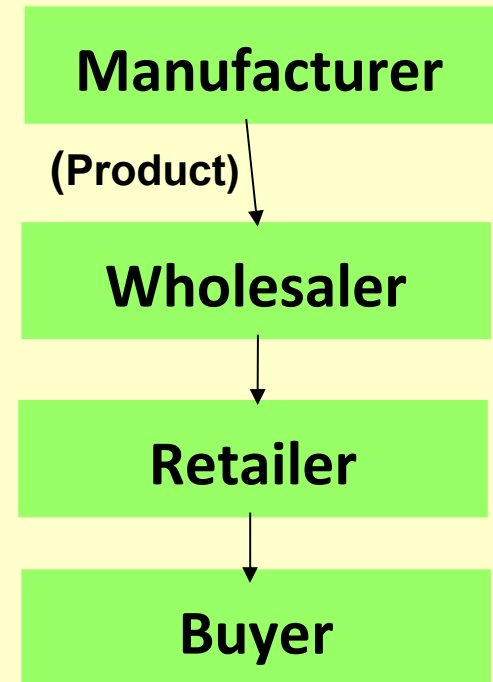
How to benefit from ISO/IEC 12207?

Case studies and examples

- Acquisition: Traceability of products
- Stakeholder requirements=> Software requirements:
Logistics
- Supply: problems with users' acceptance
- Maintenance: Destruction of confidential data
- Training: Selection of a training company

A case study in system acquisition: traceability of products

- The manufacturer (major supplier) requires from the wholesaler that the individual products must be traceable until the retailer
- This requirement was in the initial problem statement for the wholesaler information system, not tested, nobody has used
- Traceability does not work, changes are needed very fast, warranty has expired
- Some of the acquisition activities were missing?



ISO/IEC 12207 example: Acquisition process activities

- Acquisition preparation
- Acquisition advertisement
- Supplier selection
- Contract agreement
- Agreement monitoring
- Acquirer acceptance
- Closure

ISO/IEC 12207 example: traceability case study cont-d

[Acquisition process activities: Acquisition preparation. Acquisition advertisement. Supplier selection. Contract agreement. Agreement monitoring. Acquirer acceptance. Closure.]

■ From Acquirer acceptance:

- The acquirer should prepare for acceptance based on the defined acceptance strategy and criteria. The preparation of test cases, test data, test procedures, and test environment should be included.
- The acquirer shall conduct acceptance review and acceptance testing of the deliverable software product or service and shall accept it from the supplier when all acceptance conditions are satisfied.
- After acceptance the acquirer should manage the software configuration of the delivered software product

Life cycle models and ISO/IEC 12207

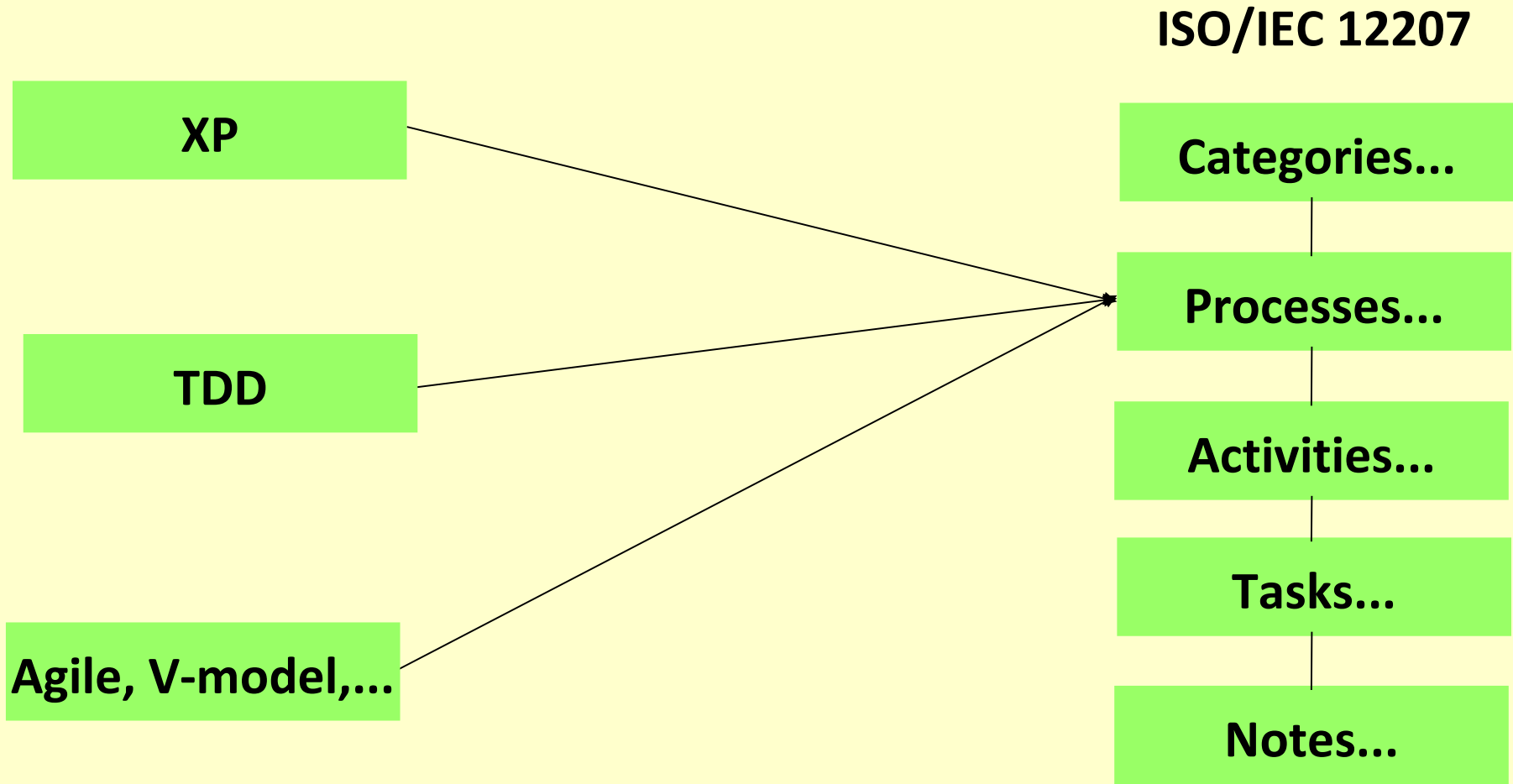
ISO/IEC 12207:

- does not detail documentation
- does not prescribe a specific system or software life cycle model, development methodology, method, model or technique
- the stakeholders are responsible for selecting a life cycle model for the software project and mapping the processes, activities, and tasks in this International Standard onto that model

Can be used with ISO/IEC 12207:

- Agile methods, eg XP, TDD, Scrum,...
- Different life cycle models: evolutionary, incremental, waterfall
- Each of these life cycle models can combined

Some possible combinations



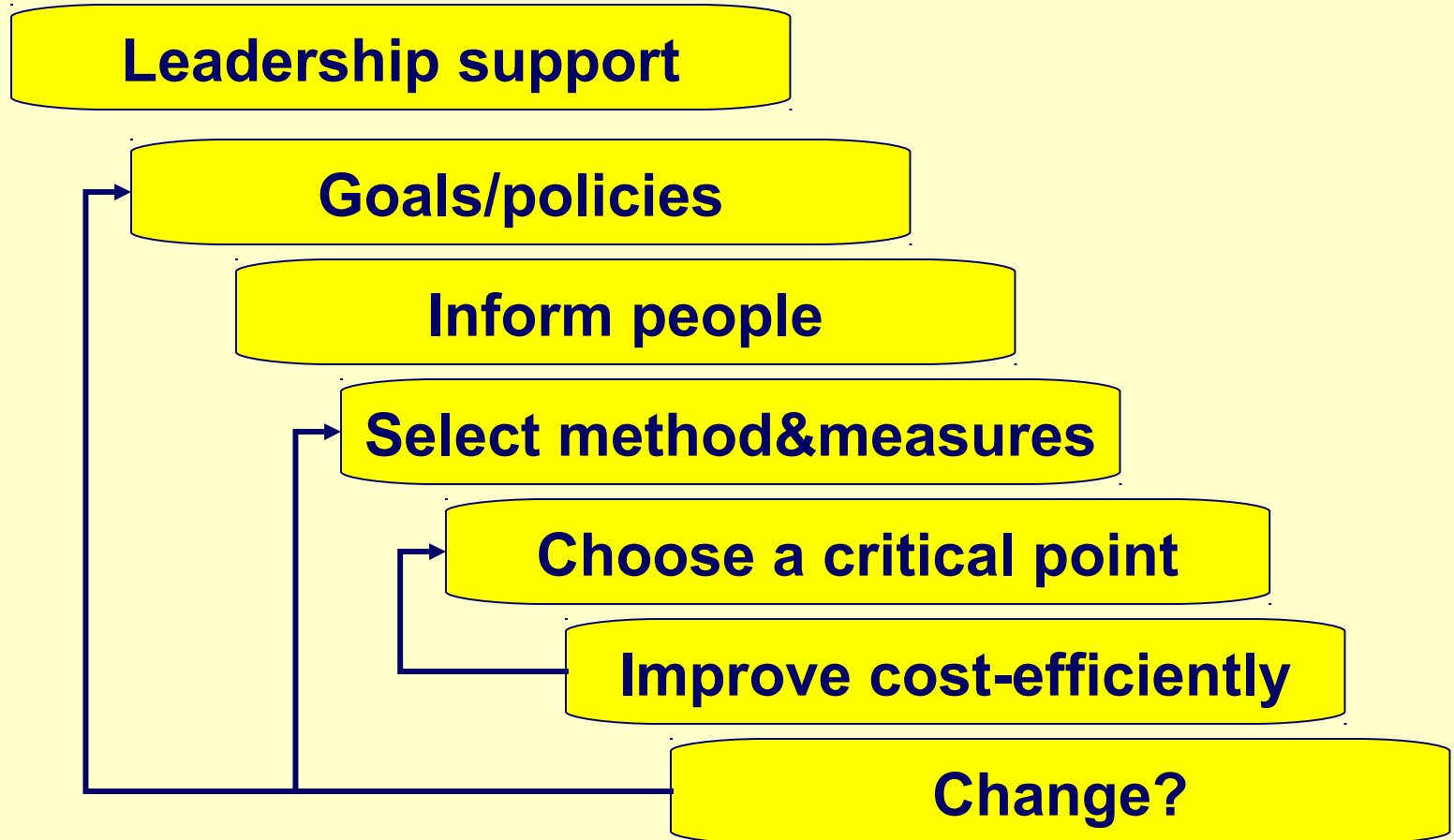
Implementation of ISO/IEC 12207

- Plan the implementation;
- Tailoring ISO/IEC 12207;
- Conduct pilot project(s);
- Formalize the approach;
- Institutionalize the approach.

Tailoring ISO/IEC 12207 - Select processes, activities and tasks

- Tailoring may be done: on organisation level (eg, establishing standard life cycle models); on project level applying the organisation level processes; or directly in individual projects.
- It is usually preferable to start with those processes that will achieve the most significant returns, rather than attempting to implement all of ISO/IEC 12207 at once.
- ISO/IEC 12207 does not define the sequencing of processes, activities and tasks and it does not prescribe any particular software life cycle model.
- Mapping the current processes, practices and/or methods to the processes, activities and tasks of ISO/IEC 12207 is useful at this stage.

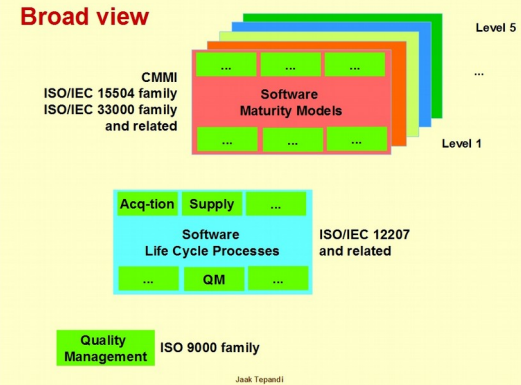
Implementing standards



ISO/IEC 12207 Information technology – Software life cycle processes

- **Why? Concept, idea**
- **What? Content**
- **Who? Organisation behind. Who might implement?**
- **When? When to use it, when not? Advantages, disadvantages?**
- **Where? Relationship to other methods**
- **How? How can my organisation benefit? How to implement?**

Capability / Maturity models



- Examples
- Capability levels in ISO/IEC 12207
- SEI CMM=>CMMI
- Test process maturity models

Capability / Maturity model examples

- Capability levels in ISO/IEC 12207, based on ISO/IEC 15504-2
- ISO/IEC 15504 family (SPICE*) revised by ISO/IEC 33000 family of standards
- CMMI and related
- Test Maturity Model integration (the TMMi Model) and TPI Next use maturity levels

***SPICE - Software Process Improvement and Capability Determination**

Capability levels in ISO/IEC 12207:2008

Adopted capability levels from ISO/IEC TR 15504-2:

0 - Incomplete Process: few or no easily identifiable work products or outputs of the process

1 - Performed Process: the purpose of the process is generally achieved

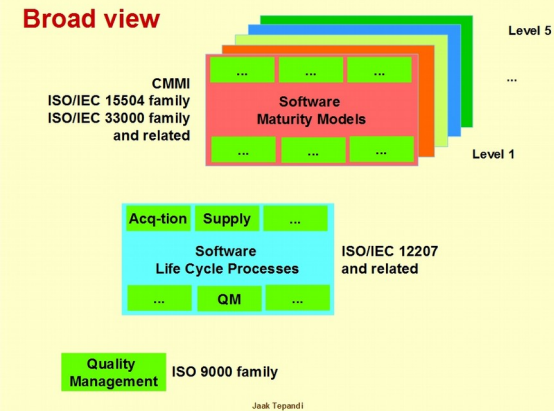
2 - Managed Process: the process delivers work products according to specified procedures and is planned and tracked

3 - Established Process: the process is performed and managed using a defined process based upon good software engineering principles

4 - Predictable Process: the defined process is performed consistently in practice within defined control limits, to achieve its defined process goals. Detailed measures of performance are collected and analyzed

5 - Optimizing Process: performance of the process is optimized to meet current and future business needs

Capability Maturity Model Integration (CMMI)

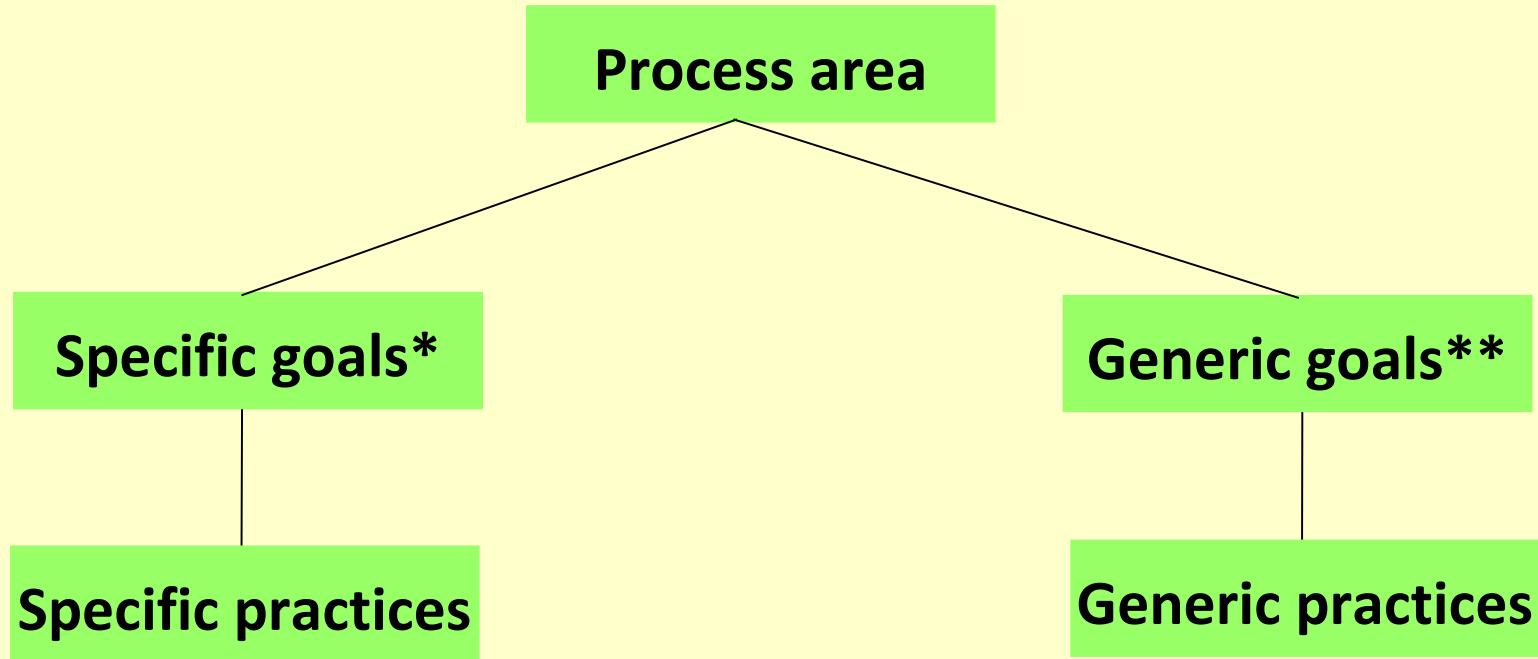


- Developed at Carnegie Mellon University
- CMMI for Development / Services / Acquisition / People
- Continuous and staged view
- Capability levels (Continuous): Incomplete (0), Performed (1), Managed (2), Defined (3)
- Maturity levels (Staged): Initial (1), Managed (2), Defined (3), Quantitatively managed (4), Optimizing (5)
- CMMI for Development (CMMI-DEV); Services (CMMI-SVC); Acquisition (CMMI-ACQ)

CMMI-DEV

http://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15287.pdf

CMMI components



* Determine content of the process area.
Apply to the current process area

** Determine capability level.
Apply to multiple process areas

CMMI ideas

- **Process areas: What to do**
- **Capability levels: How to do**
- **Maturity levels. How much (which process areas) + how (capability level) to do**
- **Capability levels apply to individual process areas**
- **Maturity levels apply to the organisation**
- **CMMI enables evaluating the current level and setting target level**
- **CMMI enables prioritizing the activities toward target profiles**
- **CMMI provides an example followed by several other frameworks**

The 22 process areas (CMMI-DEV, V1.3)

- Causal Analysis and Resolution (CAR)
- Configuration Management (CM)
- Decision Analysis and Resolution (DAR)
- Integrated Project Management (IPM)
- Measurement and Analysis (MA)
- Organizational Process Definition (OPD)
- Organizational Process Focus (OPF)
- Organizational Performance Management (OPM)
- Organizational Process Performance (OPP)
- Organizational Training (OT)
- Product Integration (PI)
- Project Monitoring and Control (PMC)
- Project Planning (PP)
- Process and Product Quality Assurance (PPQA)
- Quantitative Project Management (QPM)
- Requirements Development (RD)
- Requirements Management (REQM)
- Risk Management (RSKM)
- Supplier Agreement Management (SAM)
- Technical Solution (TS)
- Validation (VAL)
- Verification (VER)

(where is implementation?)

Summary of CMMI target profiles for maturity levels

Process Area	ML	CL1	CL2	CL3
Configuration Management	2	Target Profile 2		
Measurement and Analysis	2			
Project Monitoring and Control	2			
Project Planning	2			
Process and Product Quality Assurance	2			
Requirements Management	2			
Supplier Agreement Management	2			
Decision Analysis and Resolution	3	Target Profile 3		
Integrated Project Management	3			
Organizational Process Definition	3			
Organizational Process Focus	3			
Organizational Training	3			
Product Integration	3			
Requirements Development	3			
Risk Management	3			
Technical Solution	3			
Validation	3			
Verification	3			
Organizational Process Performance	4		Target Profile 4	
Quantitative Project Management	4			
Causal Analysis and Resolution	5	Target Profile 5		
Organizational Performance Management	5			

ML - Maturity levels (Staged):
 Initial (1),
 Managed (2),
 Defined (3),
 Quantitatively managed (4),
 Optimizing (5)

CL - Capability levels (Continuous):
 Incomplete (0),
 Performed (1),
 Managed (2),
 Defined (3)

Source: CMMI-DEV, V 1.3, Fig 3.4

Maturity level 2

- The projects of the organization have ensured that requirements are managed and that processes are planned, performed, measured, and controlled
- Existing practices are retained during times of stress. When these practices are in place, projects are performed and managed according to their documented plans
- Requirements, processes, work products, and services are managed. The status of the work products and the delivery of services are visible to management at defined points
- Commitments are established among relevant stakeholders and are revised as needed. Work products are reviewed with stakeholders and are controlled. The work products and services satisfy their specified requirements, standards, and objectives.

Process areas on maturity level 2

- Configuration Management
- Measurement and Analysis
- Project Monitoring and Control
- Project Planning
- Process and Product Quality Assurance
- Requirements Management
- Supplier Agreement Management

... To achieve maturity level 2, these process areas should be a least on capability level 2

Specific goal and practice summary example: Technical solution

Specific Goal and Practice Summary

SG 1 Select Product Component Solutions

SP 1.1 Develop Alternative Solutions and Selection Criteria

SP 1.2 Select Product Component Solutions

SG 2 Develop the Design

SP 2.1 Design the Product or Product Component

SP 2.2 Establish a Technical Data Package

SP 2.3 Design Interfaces Using Criteria

SP 2.4 Perform Make, Buy, or Reuse Analyses

SG 3 Implement the Product Design

SP 3.1 Implement the Design

SP 3.2 Develop Product Support Documentation

Generic goal /practices example (Level 1)

Generic Goal 1 Achieve Specific Goals

Generic Practice 1.1 Perform Specific Practices

Perform the specific practices of the process area to develop work products and provide services to achieve the specific goals of the process area.

The purpose of this generic practice is to produce the work products and deliver the services that are expected by performing (i.e., executing) the process. These practices can be done informally without following a documented process description or plan. The rigor with which these practices are performed depends on the individuals managing and performing the work and can vary considerably.

Generic goal /practices example (Level 2)

Generic Goal (GG) 2 - Managed process

Generic Practice (GP) 2.3 Provide Resources. Provide adequate resources for performing the process, developing the work products, and providing the services of the process

Technical Solution Elaboration. Special facilities may be required for developing, designing, and implementing solutions to requirements. When necessary, these facilities... are developed or purchased

Examples of resources provided include the following:

- **Design specification tools**
- **Simulators and modeling tools**
- **Prototyping tools**
- **Scenario definition and management tools**
- **Requirements tracking tools**
- **Interactive documentation tools**

Generic goal /practices example (Level 3)

GG 3 Institutionalize a Defined Process

GP 3.2 Collect Process Related Experiences

Collect process related experiences derived from planning and performing the process to support the future use and improvement of the organization's processes and process assets.

- Examples of process related experiences include work products, measures, measurement results, lessons learned, and process improvement suggestions. The information and artifacts are collected so that they can be included in the organizational process assets and made available to those who are (or who will be) planning and performing the same or similar processes.

TS Elaboration

- Examples of process related experiences include the following:
- Results of the make, buy, or reuse analysis
- Design defect density
- Results of applying new methods and tools

CMMI-DEV in an organization

- **CMMI-DEV does not specify that a project or organization must follow a particular process flow or that a certain number of products be developed per day or specific performance targets be achieved.**
- **The model does specify that a project or organization should have processes that address development related practices.**
- **To determine whether these processes are in place, a project or organization maps its processes to the process areas in this model.**
- **Do not expect that every CMMI-DEV process area will map one to one with your organization's or project's processes.**

Using CMMI models

Process areas help improve processes (no processes are given in CMMI)

Three selections are needed to apply CMMI to your organization for process improvement:

1. Select a part of the organization
2. Select a model
3. Select a representation

To help those who use Agile methods, notes have been added to selected process areas

Using CMMI only for appraisal may be counter-productive (similar to certification based on ISO 9001)

Capability Maturity Model Integration (CMMI)

- **Why? Concept, idea**
- **What? Content**
- **Who? Organisation behind. Who might implement?**
- **When? When to use it, when not? Advantages, disadvantages?**
- **Where? Relationship to other methods**
- **How? How can my organisation benefit? How to implement?**

Summary

- **Understanding of the purpose and application areas of ISO 9000 family, ISO/IEC 12207, and maturity models**
- **Understanding of the basic concepts and structure of these frameworks**
- **Given an organisation and a system - ability to choose, and to provide a rudimentary justification for the choice of a process model**
- **Understanding of the main activities needed to start selection and application of these frameworks and standards**
- **Ability to find needed information in these process models**

Useful to IT manager, programmer, maintainer, and other stakeholders involved in various software related processes

Additional reading (examples)

ISO/IEC 12207:2008, <https://www.evs.ee/tooted/iso-iec-12207-2008>

Capability maturity model for Software,
<https://www.sei.cmu.edu/reports/93tr024.pdf>

CMMI, <http://cmmiinstitute.com/>, <http://www.cmmifaq.info/>

CMMI-DEV,
http://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15287.pdf

Test Maturity Model integration (the TMMi Model), <https://www.tmmi.org/>

TPI Next, <http://www.tmap.net/tpi-downloads>

Daniel Galin, Software Quality assurance from theory to implementation,
Pearson - Addison-Wesley. Chapter 23, 24

Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE.
Chapter 8, Section 3