Software quality, processes, and standards
Basic concepts: processes (first iteration)

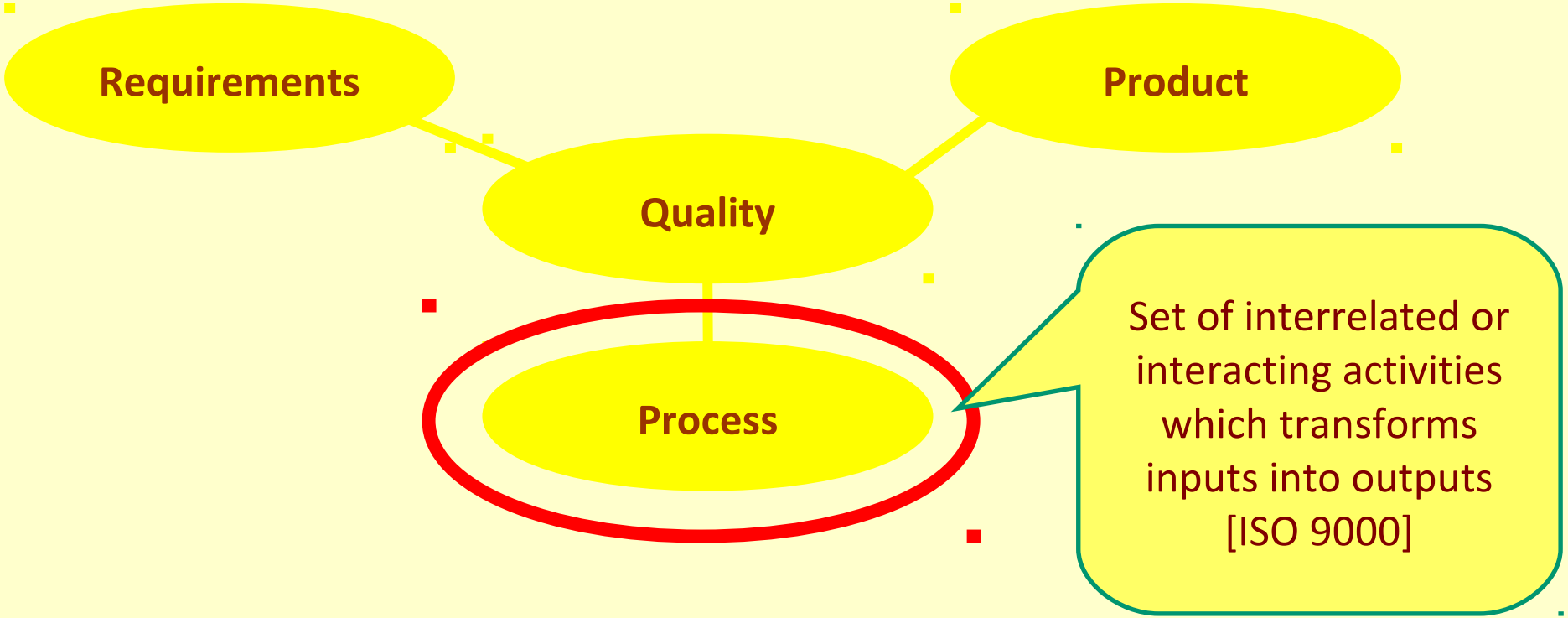
Jaak Tepandi, Jekaterina Tšukrejeva, Stanislav Vassiljev, Pille Haug
Tallinn University of Technology
Department of Software Science
Moodle: „Software Quality (Tarkvara kvaliteet)”
Alternate download: tepandi.ee
Version 27.09.2017

Today and the course

Software quality, processes, and standards

- Quality
  - Why? Who? What?
  - Testing example
  - Quality
  - Course
  - Product
  - Require-rements
  - Processes
    - Uses
    - Processes
    - Life cycles
    - Acquisition
    - SLAs
- V&V
- Quality and processes
- Moodle tepandi.ee

How to improve processes for better software quality? (first iteration)



Software processes and development life cycles

Process uses

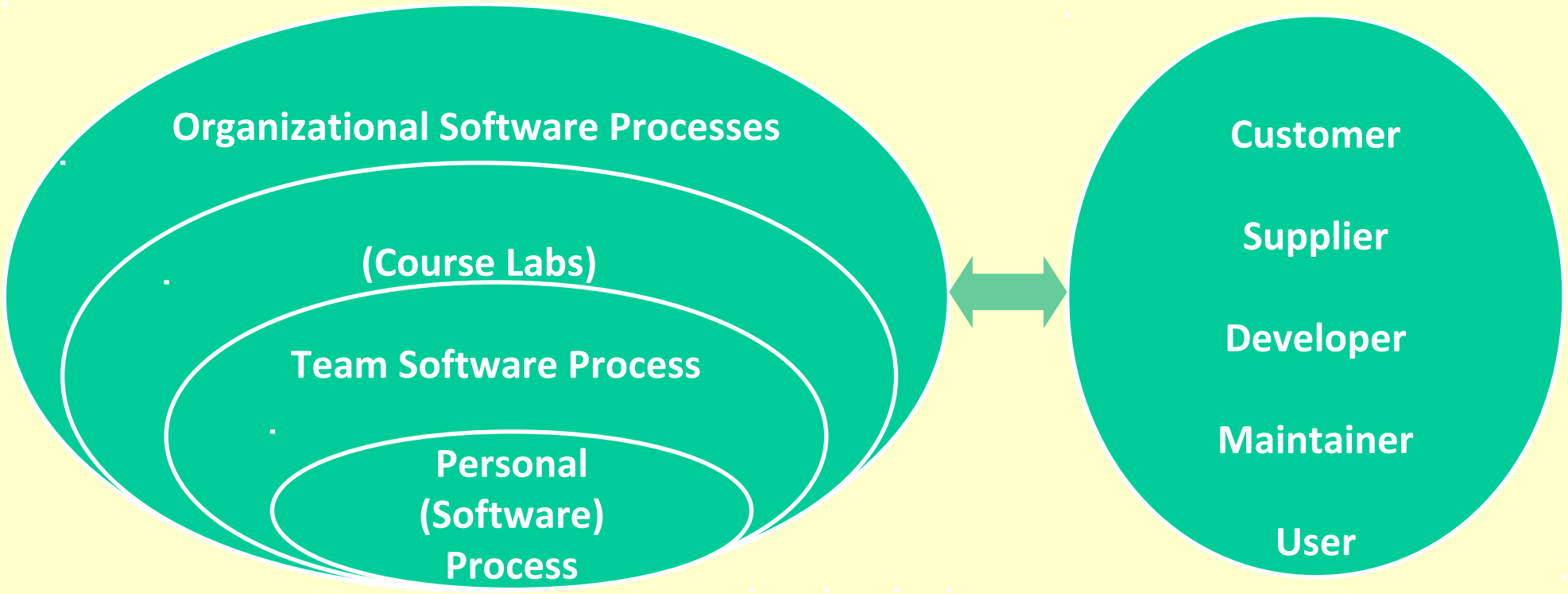
Software processes and development life cycles

- Relationship: process frameworks and development life cycles
- Process frameworks – ISO/IEC 12207, Capability maturity models, ...
- Development life cycle examples

Acquisition

Service Level Agreement

Process uses



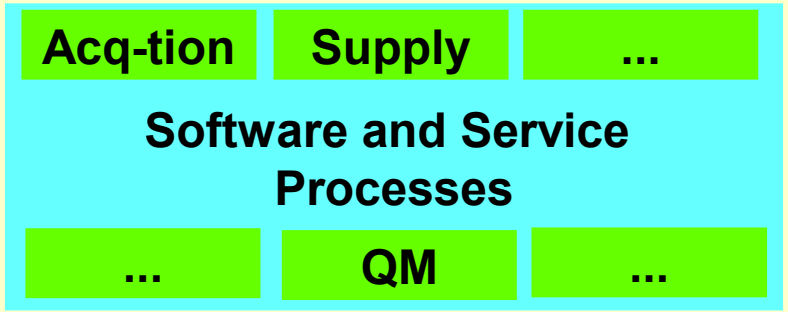
Broad view

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

Level N

...

Level 1
Level 0



ISO/IEC 12207
ISO 20000 series (ITIL)
and related

Quality Management

ISO 9000 family

SDLC-s

... and more

Process frameworks and life cycle models

- Process frameworks present many processes (eg ISO/IEC 12207, CMMI, COBIT, ITIL), therefore may be large
- Process frameworks may provide levels
- Life cycle models mostly concern development
- No contradiction between "process frameworks" and "agile"
- Process (lad processus, movement)
- Use as little as possible, but not less

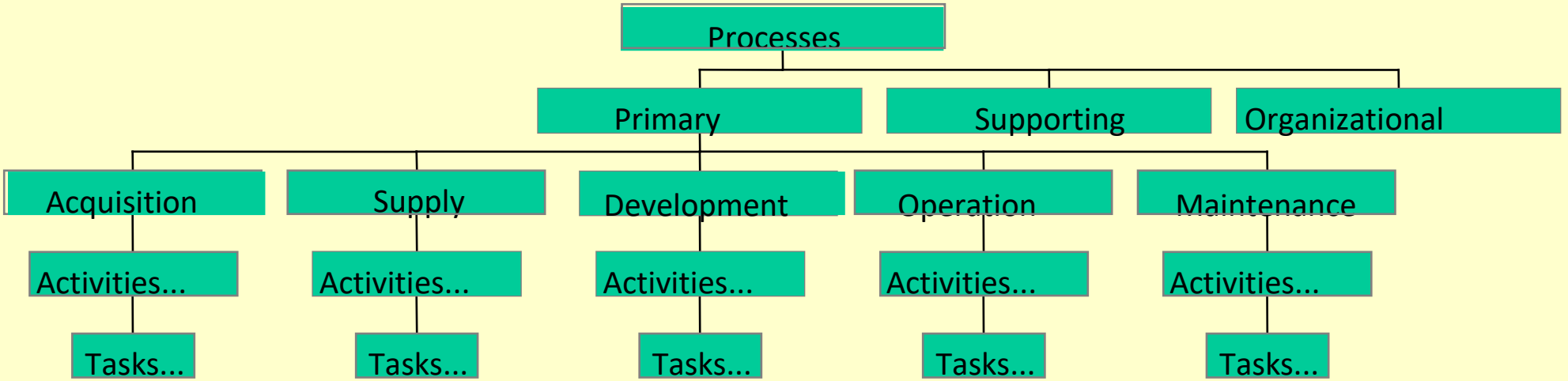
ISO/IEC 12207 Information technology-software life cycle processes (earlier version)

- Primary processes
  - Acquisition - Supply - Development - Operation - Maintenance
- Supporting processes
  - Documentation – Configuration management – Quality assurance - Verification – Validation – Joint review – Audit – Problem resolution
- Organizational processes
  - Management - Infrastructure - Improvement - Training

Available, good for first understanding

Many more processes in ISO/IEC 12207:2008

ISO/IEC 12207 Information technology – Software life cycle processes (earlier version)



ISO/IEC 12207 Information technology-software life cycle processes: versions

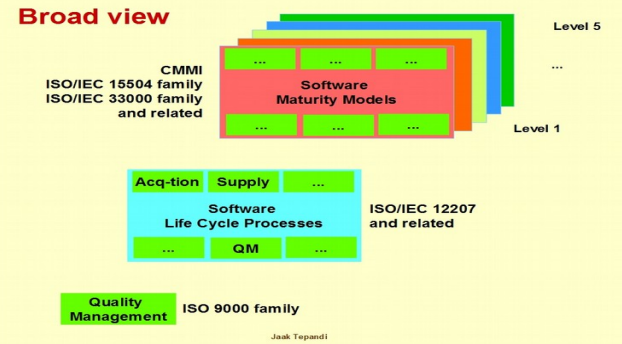
First version 1995, only software processes => amendments in 2002, 2004

Current version ISO/IEC 12207:2008, 43 software and system processes

Version under development - ISO/IEC 12207:2017

- 2 agreement processes: Acquisition, Supply
- 6 organisational project-enabling processes: Life Cycle Model Management, Infrastructure Management, Quality Management,...
- 8 technical management processes: Project Planning, Risk Management, Configuration Management, Quality Assurance, …
- 14 technical processes: Systems/Software Requirements Definition, Architecture Definition, Design Definition, Implementation, Verification, Validation, Maintenance,...

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 ideas

- Process areas: What to do
- Capability levels: How to do
- Maturity levels. How much to do (which process areas) + how to do (capability level)
- 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

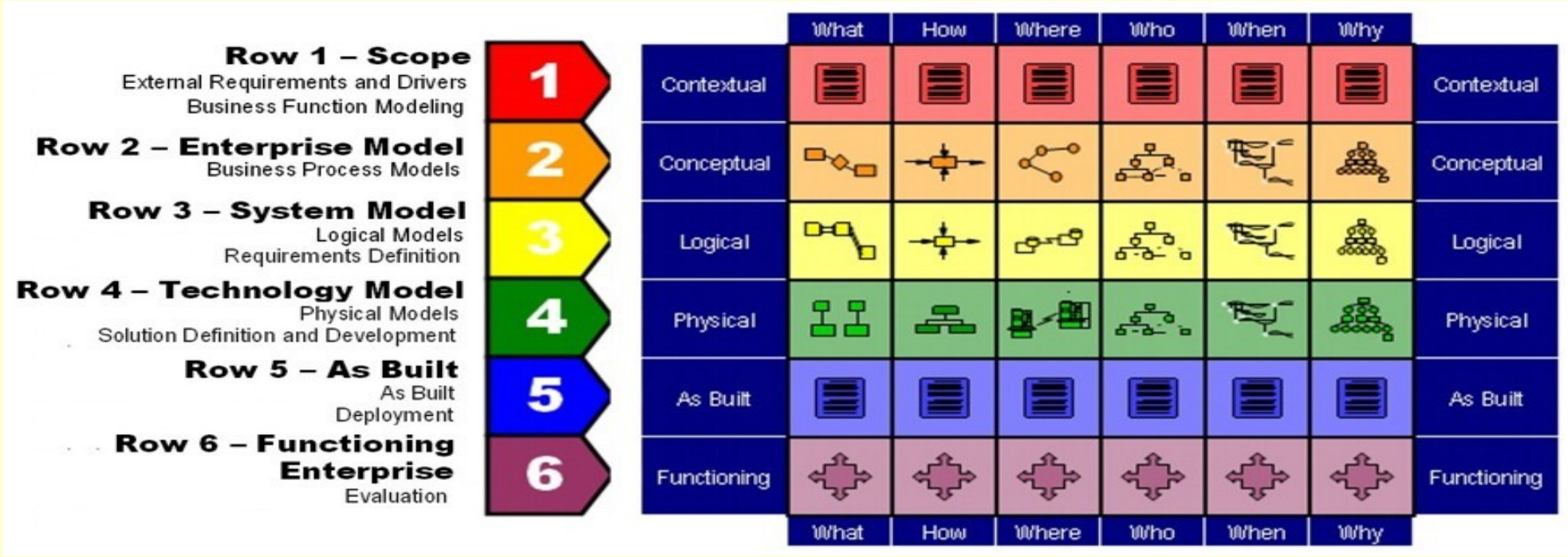
The 22 process areas (CMMI-DEV, V1.3)

- 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)

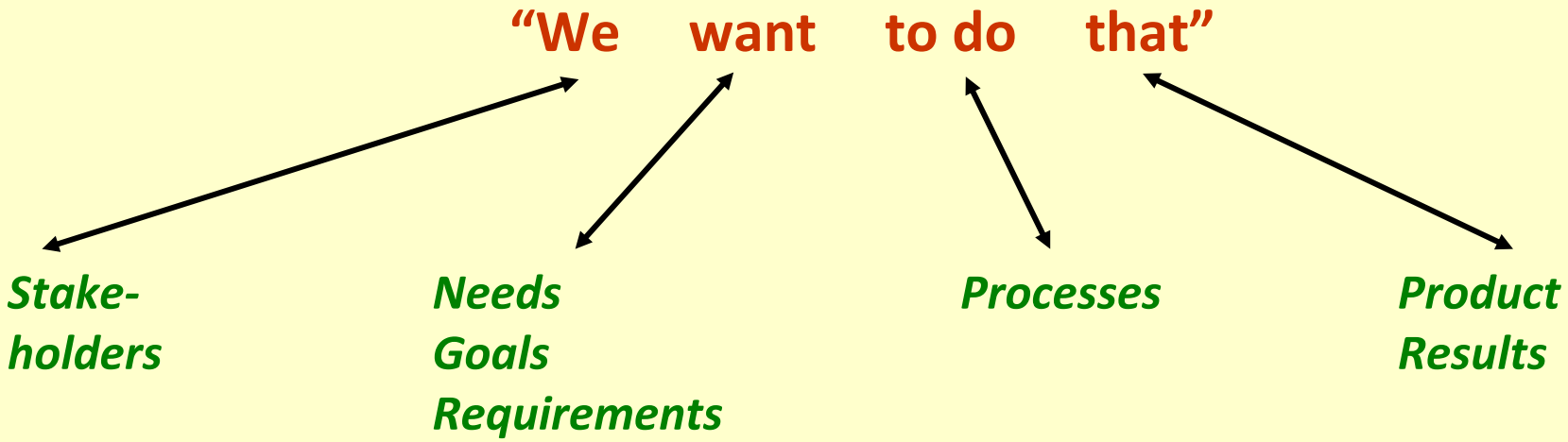
(where is implementation?)

- 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)

Zachman Framework (example)



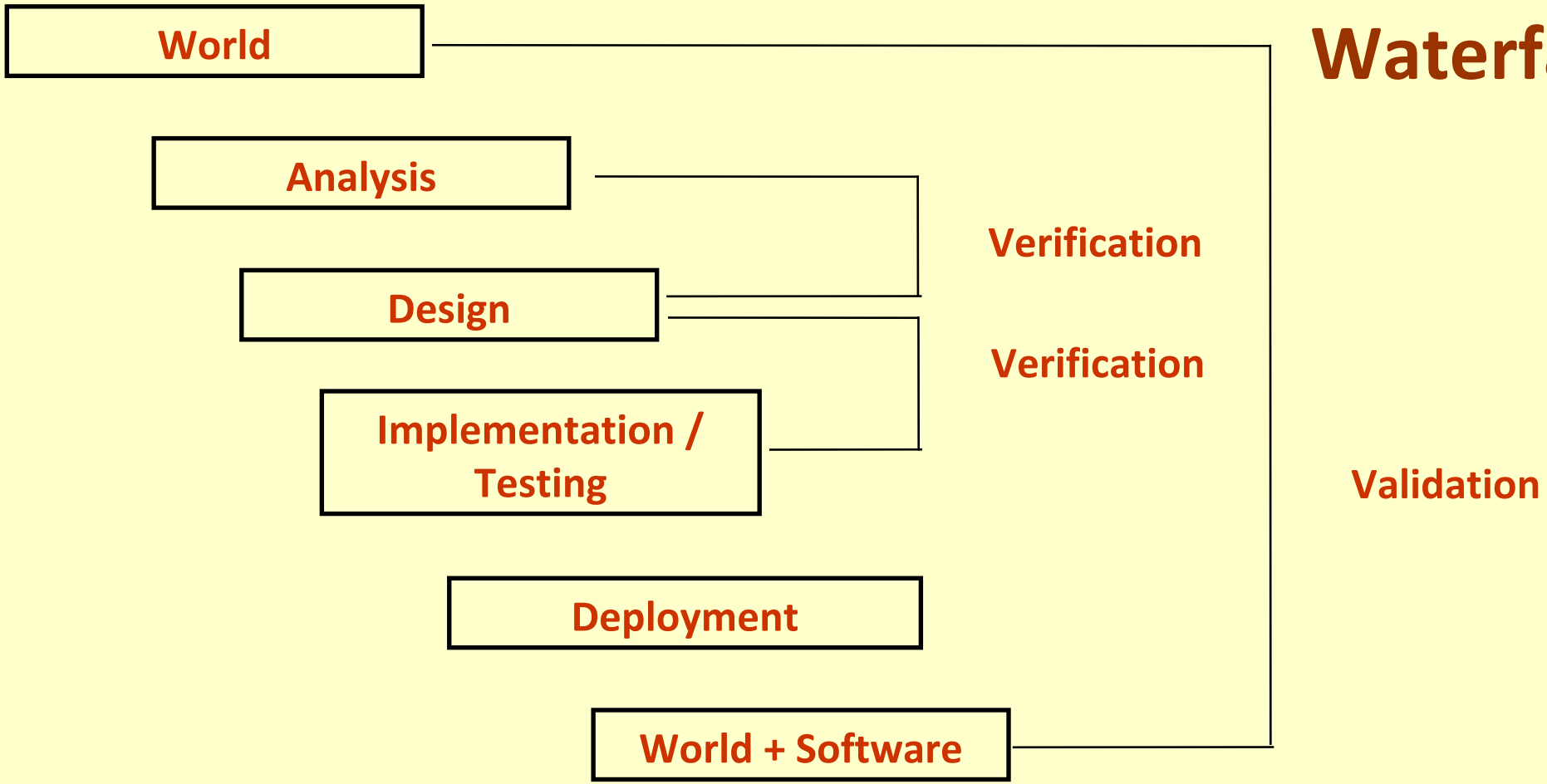
"Simplification Zachman Enterprise Framework" by Al Zuech, Director, Enterprise Architecture Service at the US Department of Veterans Affairs. - VA's Enterprise Architecture. Licensed under Public domain via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Simplification_Zachman_Enterprise_Framework.jpg#mediaviewer/File:Simplification_Zachman_Enterprise_Framework.jpg



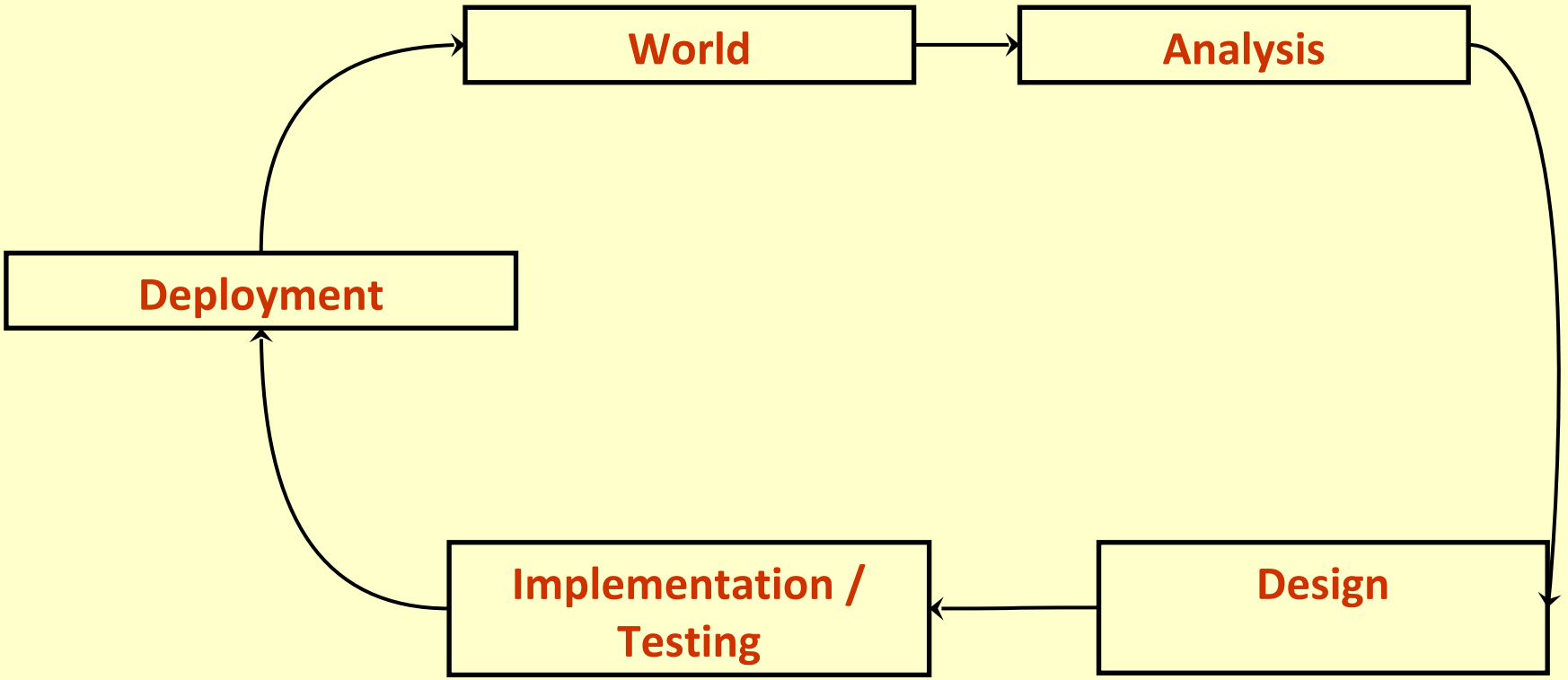
Software life cycle model / reference examples

- Waterfall - http://www.nouveautech.co.uk/images/developmentprocess.gif
- V-model - http://en.wikipedia.org/wiki/V-Model_(software_development)
- Waterfall – iterative - http://kanemar.files.wordpress.com/2005/12/Staggered-Iterative-Waterfall.jpg
- Spiral - http://en.wikipedia.org/wiki/File:Software_Development_Spiral.svg
- Extreme Programming - http://www.extremeprogramming.org/map/project.html
- Scrum - https://en.wikipedia.org/wiki/File:Scrum_process.svg
- TDD - http://en.wikipedia.org/wiki/Test-driven_development
- Also: http://agilemanifesto.org/
- Also: http://en.wikipedia.org/wiki/Software_development_methodology
- Also: https://youtu.be/xtpyjPrpyX8

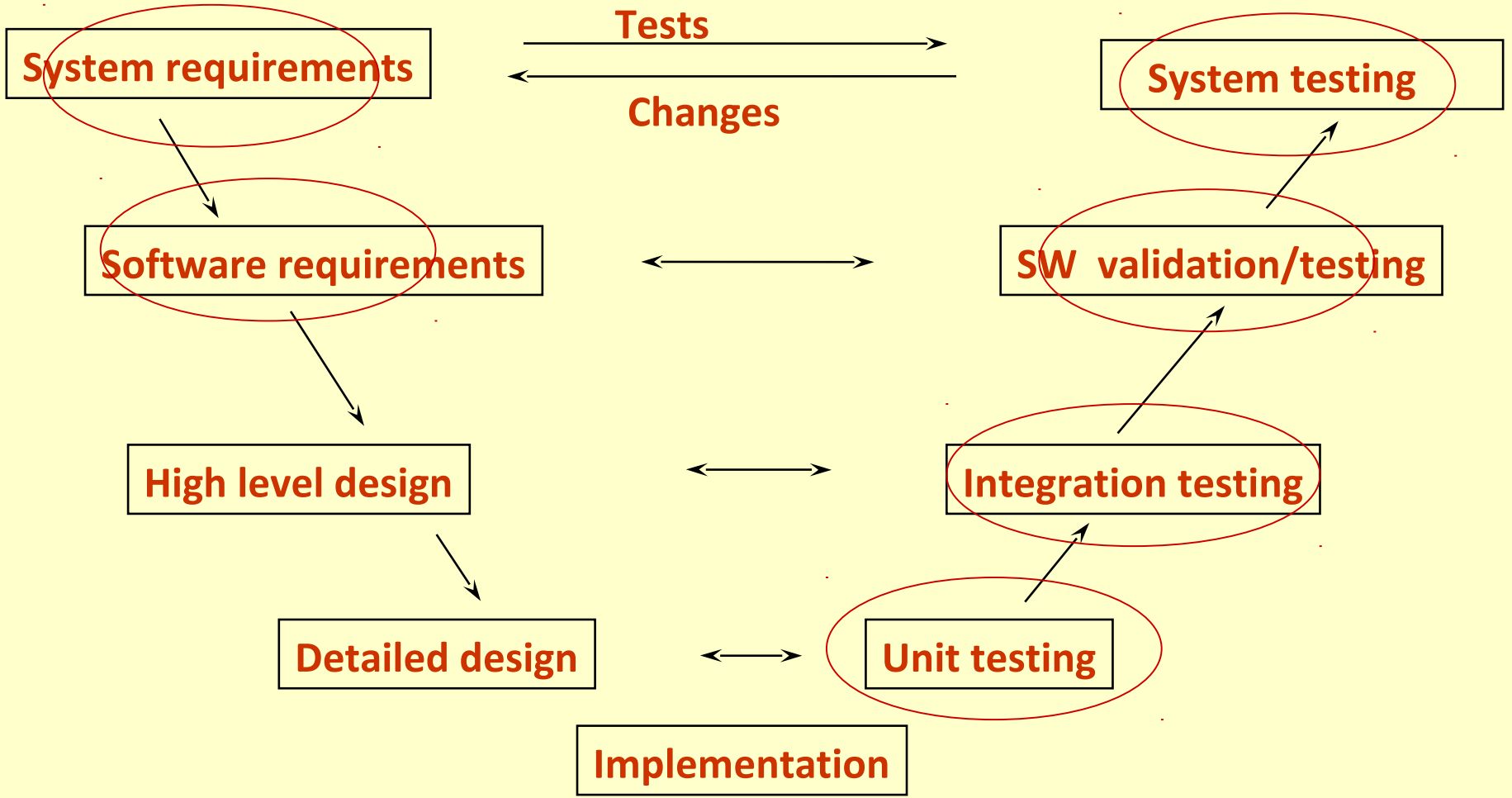
Waterfall


World
Analysis
Design
Implementation / Testing
Deployment
World + Software
Verification
Verification
Validation

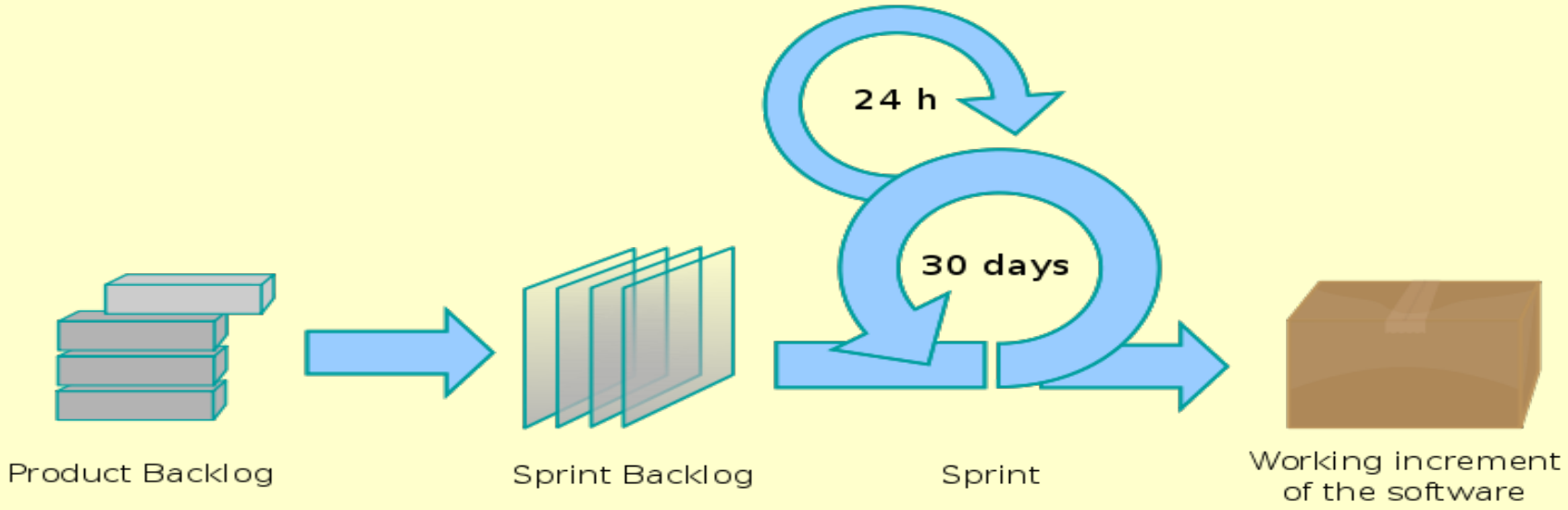
Waterfall, modified



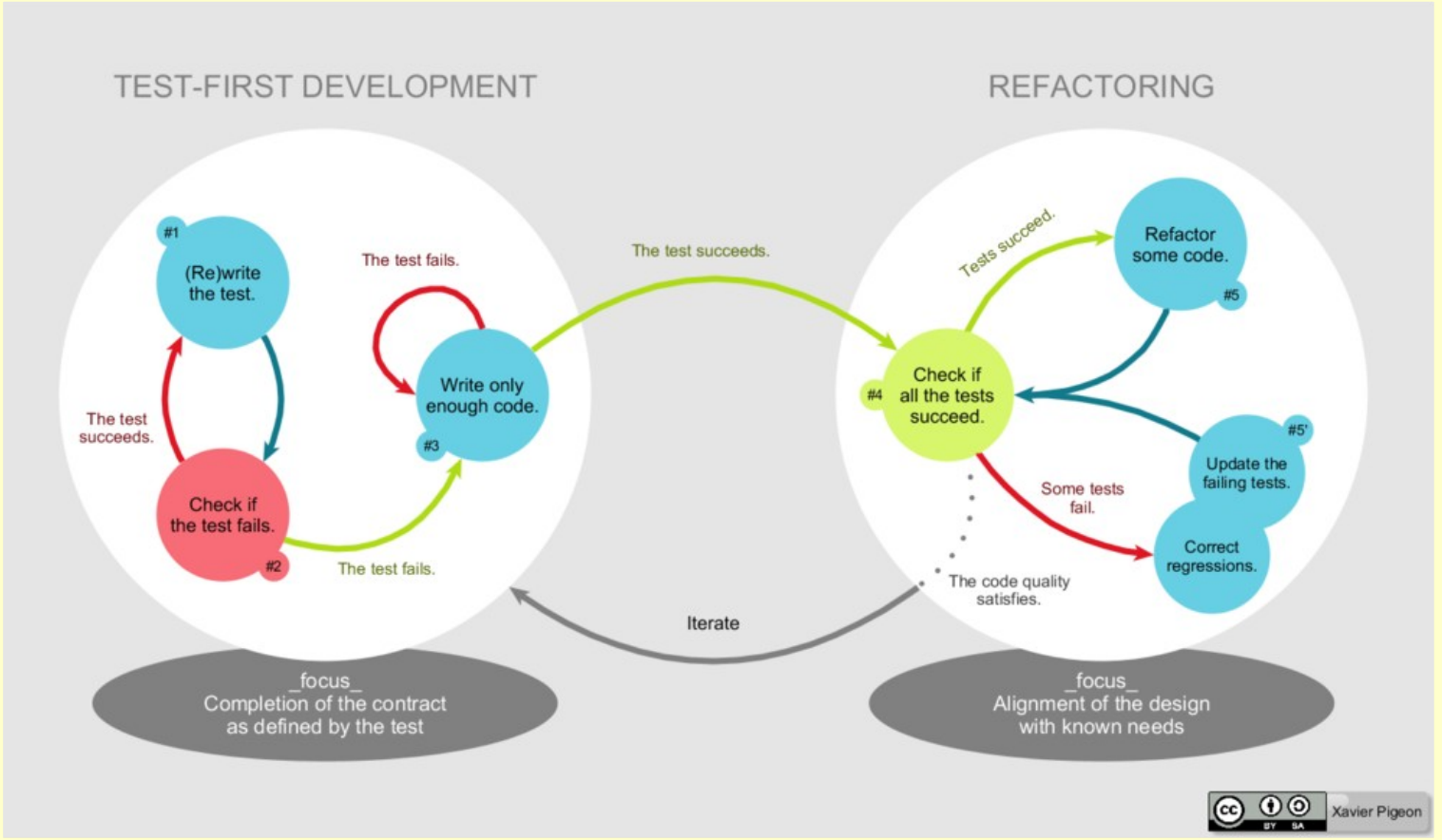
V-model (example)



The Scrum process



http://en.wikipedia.org/wiki/File:Scrum_process.svg



https://en.wikipedia.org/wiki/File:TDD_Global_Lifecycle.png

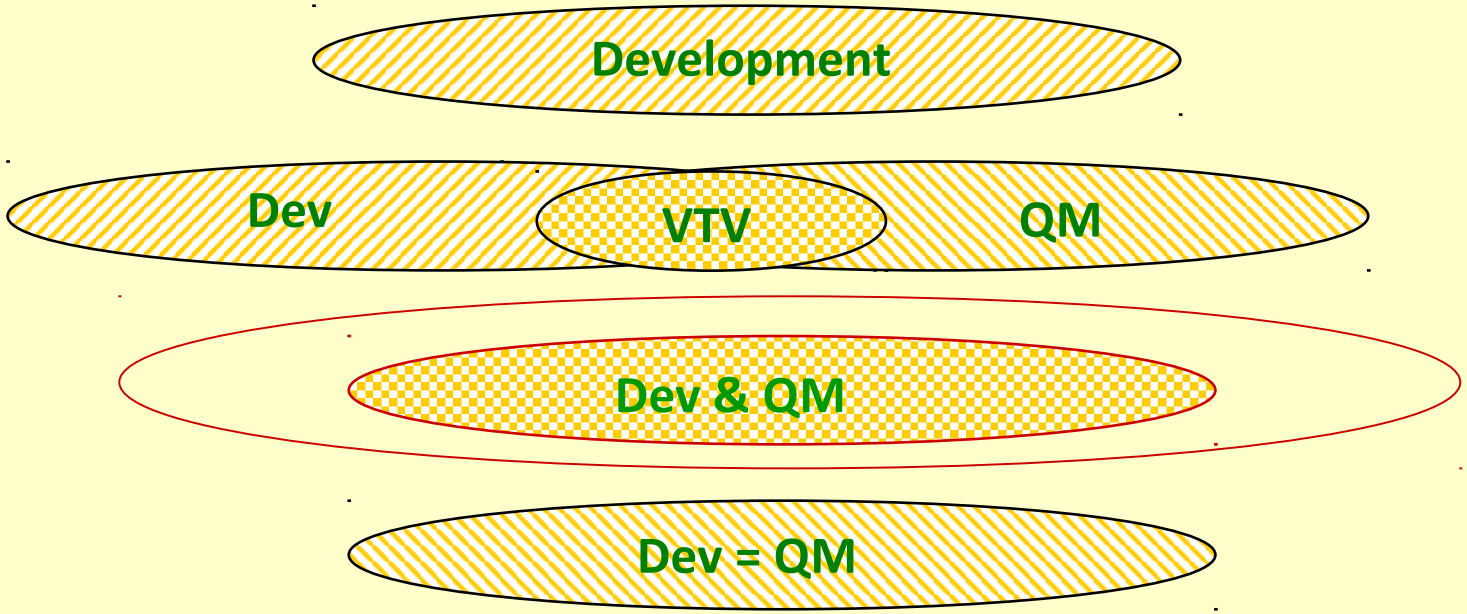
Life cycle models and ISO/IEC 12207

Can be used with ISO/IEC 12207:

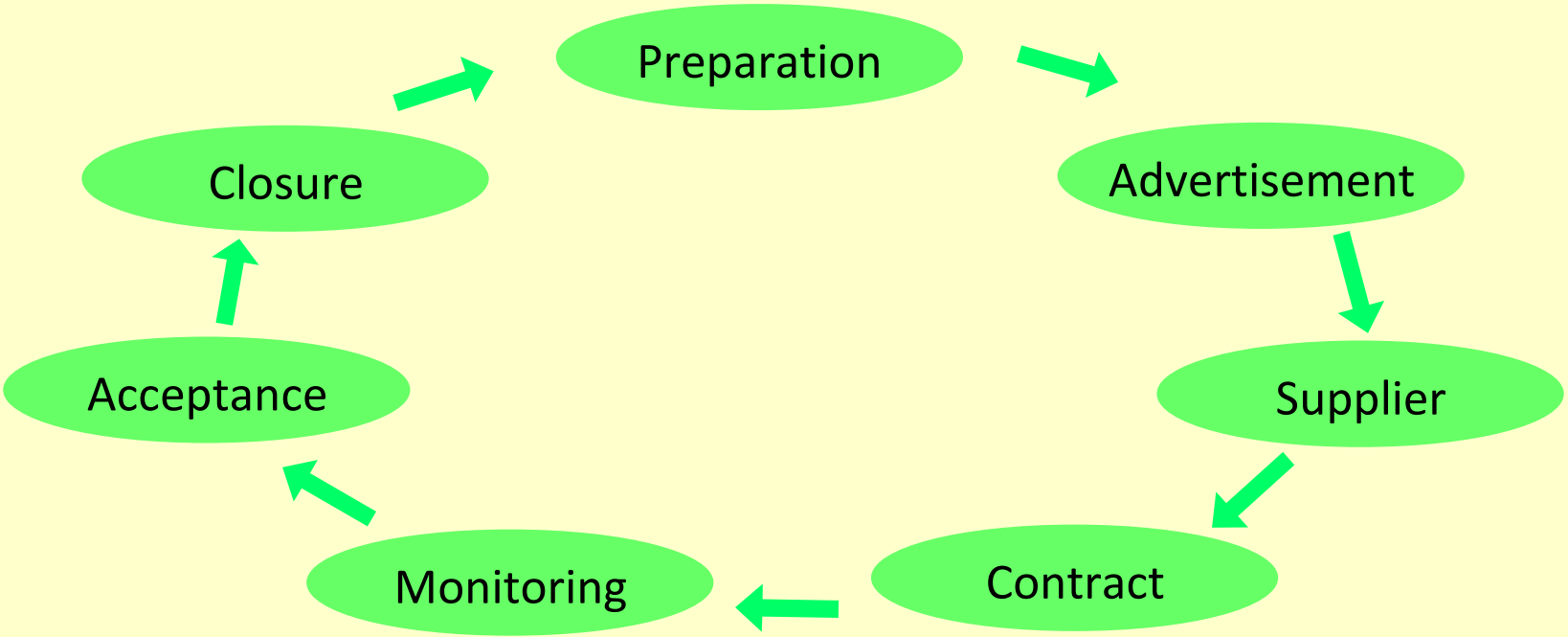
- Agile /Evolutionary
- Incremental
- Waterfall
- Each of these life cycle models can be used as is, or they can be combined to create another hybrid life cycle model

(ISO/IEC TR 15271 – guidelines, revised by ISO/IEC TR 24748-3:2011)

Development and quality management - views



Acquisition - process of obtaining a system, product or service



(ISO/IEC 12207)

Acquisition activities

Acquisition - process of obtaining a system, product or service

- Acquisition preparation (incl describing a concept or a need to acquire, system requirements, software requirements, acquisition plan, processes, options)
- Advertisement (communicating the request for supply to selected suppliers)
- Supplier selection (based on criteria identified in the request for supply)
- Contract agreement (incl rights, change control)
- Agreement monitoring (using other processes)
- Acquirer acceptance (incl acceptance strategy and criteria, test cases,acceptance testing, configuration management)
- Closure (incl payments and conclusion of the agreement)

(ISO/IEC 12207)

Manage Requirements Changes

Requirements management, specific practice (SP) 1.3. Subpractices

- Document all requirements and requirements changes that are given to or generated by the project.
- Maintain a requirements change history, including the rationale for changes.
- Evaluate the impact of requirement changes from the standpoint of relevant stakeholders.
- Make requirements and change data available to the project.

(CMMI-ACQ)

Acquisition validation

The purpose of Acquisition Validation is to demonstrate that an acquired product or service fulfills its intended use when placed in its intended environment.

SG 1 Prepare for Validation

SP 1.1 Select Products for Validation

SP 1.2 Establish the Validation Environment

SP 1.3 Establish Validation Procedures and Criteria

SG 2 Validate Selected Products and Product Components

SP 2.1 Perform Validation

SP 2.2 Analyze Validation Results

CMMI-ACQ

Acquisition verification

The purpose of Acquisition Verification is to ensure that selected work products meet their specified requirements.

Specific Goal (SG) 1 Prepare for Verification

- SP 1.1 Select Work Products for Verification
- SP 1.2 Establish the Verification Environment
- SP 1.3 Establish Verification Procedures and Criteria

SG 2 Perform Peer Reviews

- SP 2.1 Prepare for Peer Reviews
- SP 2.2 Conduct Peer Reviews
- SP 2.3 Analyze Peer Review Data

SG 3 Verify Selected Work Products

- SP 3.1 Perform Verification
- SP 3.2 Analyze Verification Results

(CMMI-ACQ)

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 extent of supplier involvement should be defined.

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 take the responsibility for the configuration management of the delivered software product.

(ISO-IEC 12207)

Process /life cycle model usage

- Need depends on the situation (organisation, stage, complexity, volumes etc)
- Insufficient processes: inefficient work, stress etc
- Too strict processes suppress creativity and efficiency
- Use as little as possible, but not less
- Usage may be less or more formal depending on situation

Process /life cycle model usage (2)

- +/- of the models?
- What is the best model?
- When are such models needed? When not?
- Different needs for different processes in a:
  - large telecom, bank
  - startup
  - software development company
  - chemistry manufacturer
  - public sector organisation
  - port, power plant, prison, ...

Services and SLAs

- **Service -** means of delivering value for the customer by facilitating results the customer wants to achieve
- **Service level agreement (SLA) -** documented agreement between the service provider and customer that identifies services and service targets...
- ... incorporates non-functional requirements from a business perspective
- Delivering service may require software, processes, people, infrastructure, etc
- SLAs - customer-level requirements => software, process, requirements
- Examples

  http://www.slatemplate.com/

  http://www.itsm.info/Sample%20SLA%20Templates.pdf

SLA content (examples, ISO/IEC 20000-2:2012)

- Contract related items (eg, SLA change control)
- Service quality related items, eg:
  - service hours, date exceptions, critical business periods and out-of-hours coverage
  - scheduled and agreed interruptions to services, including notice to be given
  - customer responsibilities, e.g. correct use of systems, adherence to the information security policy
  - service provider liability and obligations, e.g. security
  - escalation and notification process
  - complaints procedure
  - service targets
  - upper and lower workload limits, e.g. the ability of the service to support the agreed number of users/volume of work, system throughput
  - actions to be taken in the event of a service interruption
- SLAs: only most important aspects of the service for the business and the customer

Takeaway 1: processes

- Projects succeed better if the customer, supplier, and others have agreed on how the work is done ...

  … => have a common understanding of what processes are involved and how they are organised

- There is no sense to reinvent the processes for each organisation and project - use **process frameworks** (ISO/IEC 12207, ITIL, CMMI,...)...

  ...=> **select** useful processes (eg, acquisition, development, maintenance) and **customize** them as needed

- Software development is one of the many processes in process frameworks

  ...=> process frameworks allow for using different types of software development life-cycle models

Takeaway 2: processes

- For **software development process**, a number of software development life-cycle models are available....

  ...=> such as TDD, XP, Scrum, Agile, V-model, Spiral, Waterfall, etc. Apply the one most useful for your task

- **Acquisition**: preparation, CFP, supplier selection, contract agreement, agreement monitoring, acquirer acceptance, closure

- **Service level agreement** (SLA): documented agreement between the service provider and customer that identifies services and service targets

Key points to know

- Software process framework examples
- Software development life cycle model examples
- Relationship between development life cycle models and process frameworks.
- Acquisition process
- Service level agreements

Additional reading - processes (examples)

Ian Sommerville. Software Engineering. Ninth Edition. Addison-Wesley, Ch 2,3.

Daniel Galin, Software Quality assurance from theory to implementation, Pearson - Addison-Wesley. Chapter 7.

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

Manifesto for Agile Software Development, http://agilemanifesto.org/

Extreme Programming: A gentle introduction, http://www.extremeprogramming.org/

Scrum: http://www.scrum.org/Scrum-Guides

Lean: Mary Poppendieck, Implementing Lean Software Development: From Concept to Cash. Addison-Wesley.

SLA and non-functional requirements:
https://www.itpedia.nl/2017/07/14/devops-plan-slas-and-non-functional-requirements/