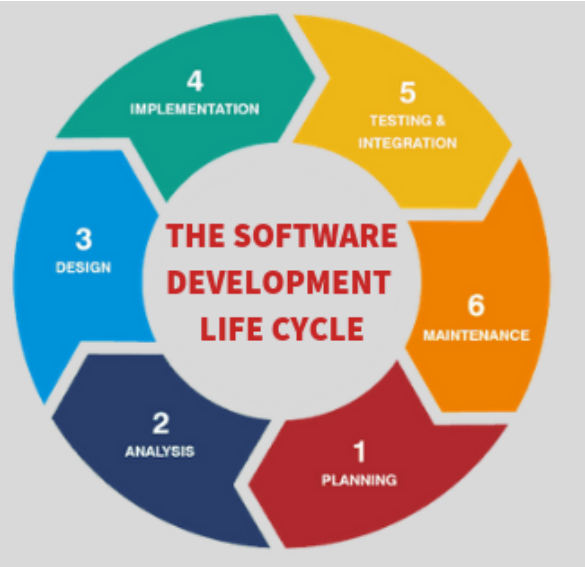
SDLC and Standards

1 Software Development Lifecycle



- The SDLC framework is used in industry to design, develop and test high quality software
- Focussing on creating software that meets the needs and expectations of the client
- Within the client's timeline and budget

2 Phases

2.1 Planning and RE(requirements) analysis

- Planning is the first and most fundamental of the phases
- Get this wrong and nothing else will work
- Inc. quality assurance and risk analysis

Stakeholders/Roles within this phase

- Client
- Customer
- Approver (managerial oversight)
- Assessor (some degree of technical expertise to advise)

Outcome

- An approved proposal
- A functional requirements doc

2.2 Design

- The goal here is to create the s/w design document(s) based on the inputs from the previous phase (planning and analysis)
- Perform design discussions, examine design patterns, consider requirements
- Output: systems design docs for DB, API, application, infrastructure, testing, training, maintenance, user ...

Roles:

- **End user** - final users of system
- **Business analyst** - provide requirements to the design team, review solution design and artefacts
- **Project Manager** - Finalize data conversion strategy and test strategy, review solution design and artefacts
- **Technical-Architect, Tech-Designer, Design-Team** - Design system architecture, software components, etc; design walk-through
- **Developer/Construction Team** - Assist with identifying and finalizing testing strategy; review of the architecture and software components
- **Database team** - Assist with architecture design and data conversion strategy

2.3 Implementation

- Get your hands dirty with coding
- Your teams may use any approach that works for you
- Within a business you will have to abide by their coding guidelines

Roles:

- **Customer**, sponsor and signs off team effort; review progress with the developers and the PM.
- **Project Manager**, resolve resource, scheduling, budget issues; review and report progress.
- **Developer**, construct a working solution from the approved design; produce artifacts and put them under configuration control and perform change control; employ tools, systems and conform to prescribed standards (platforms, coding practices, programming languages, etc) that are in line with the organization's objectives.
- **Database Administration Group**, assist with implementing the solution design and data conversion strategy.
- **Implementation supervisor/manager**, assist with identifying the requirements for implementation of the solution (which includes system readiness, resources, time-lines).
- **Integration supervisor/manager**, identify how integration of the solution in a new hardware/software environment would be achieved; what tests are required to evaluate integration.

2.4 Testing & Integration/Deployment

- These phases are sometimes separated
- The goal of testing is to check that the development is functional and meets requirements
- Complexity arises from integration of a novel system with existing (sub-)systems
- Test for various functional attributes
  - Security, conformance, accessibility, performance, stress

Roles:

- **Project Manager**, resolve resource, scheduling, budget issues; review and report progress.
- **Developer**, assist with building tests and analysis of test results.
- **Database Administration Group**, assist with integration of the solution design and data conversion tests.
- **Implementation manager**, assist with analysis of test results.
- **Integration manager**, assist with analysis of test results.

Sign off is completed when the functional requirements specification has been met

2.5 Implementation

- Some versions of the SDLC have an additional phase here
- The focus is to install the system in the production environment and to bring it into **operation**; and then to ensure that the system:
  - Satisfies the functional requirements
  - Satisfies the business needs
  - Adheres to all mandates, physical constraints and service level agreements
  - Operates as described in the User and Operator Manuals

2.6 Maintenance

- On successful operational transfer of the project, development group hands over to the maintenance group
- Documentation must be ready for transfer at this time
- Roles:
  - **Solution Delivery Team** – Prepares all solution documentation and manuals for the maintenance group. The solution delivery team will supply any requested training to help maintenance team technicians learn the solution's behaviours
  - **Solution Maintenance Team** – Reviews all solution documentation and supports the solution until the terms of the maintenance agreement expire

3 Alternatives

There are many correct models depending on the people and project. The SDLC just outlines the phases

4 Standards

Why standards?

- Quality
- Shared communication
- Shared understanding
- Influence, from understanding to creation/development
- Profit
- Collaboration
- Reputation
- Regulation (assurance)
- Flexibility

Importance

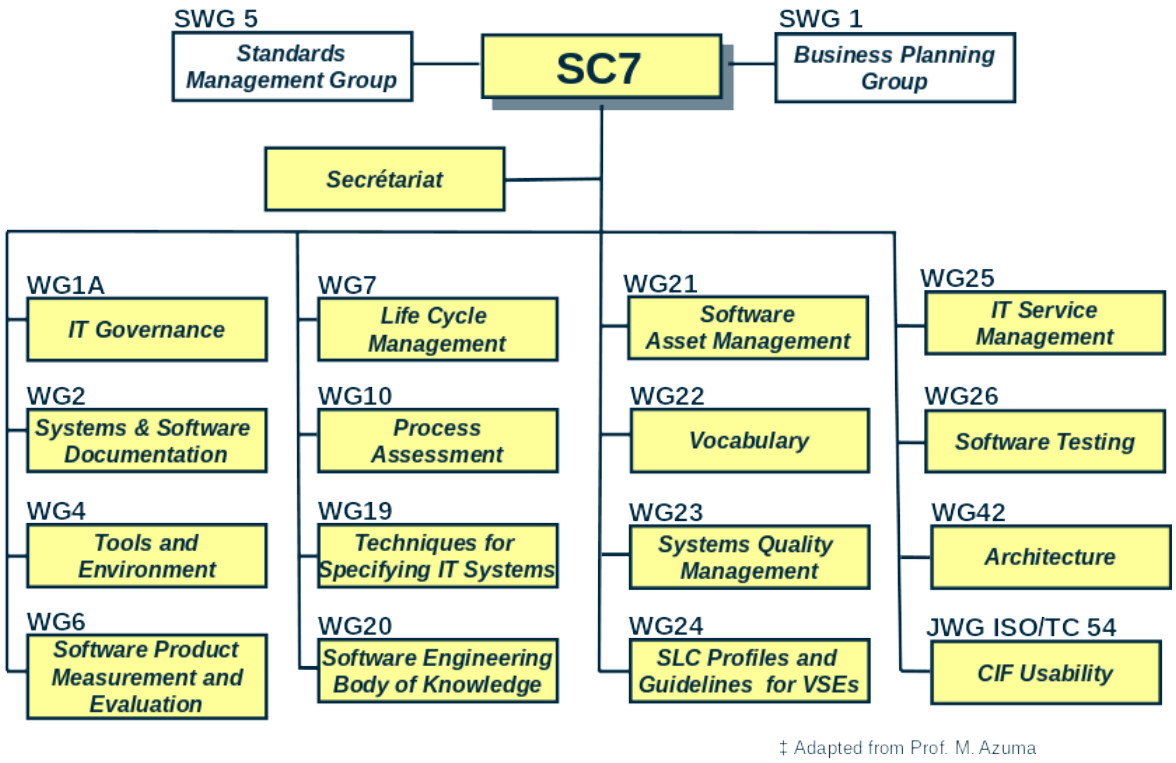
- They encapsulate best practice (normally)
- Framework for QA
- Provide continuity
  - Record of decision making process
  - Organisational memory
  - New staff save time

Issues:

- Standards are considered too large, unwieldy and difficult to adopt for SMEs
- Focus is on large organisations
- Concerns over cost and documentation
- Difficult to justify

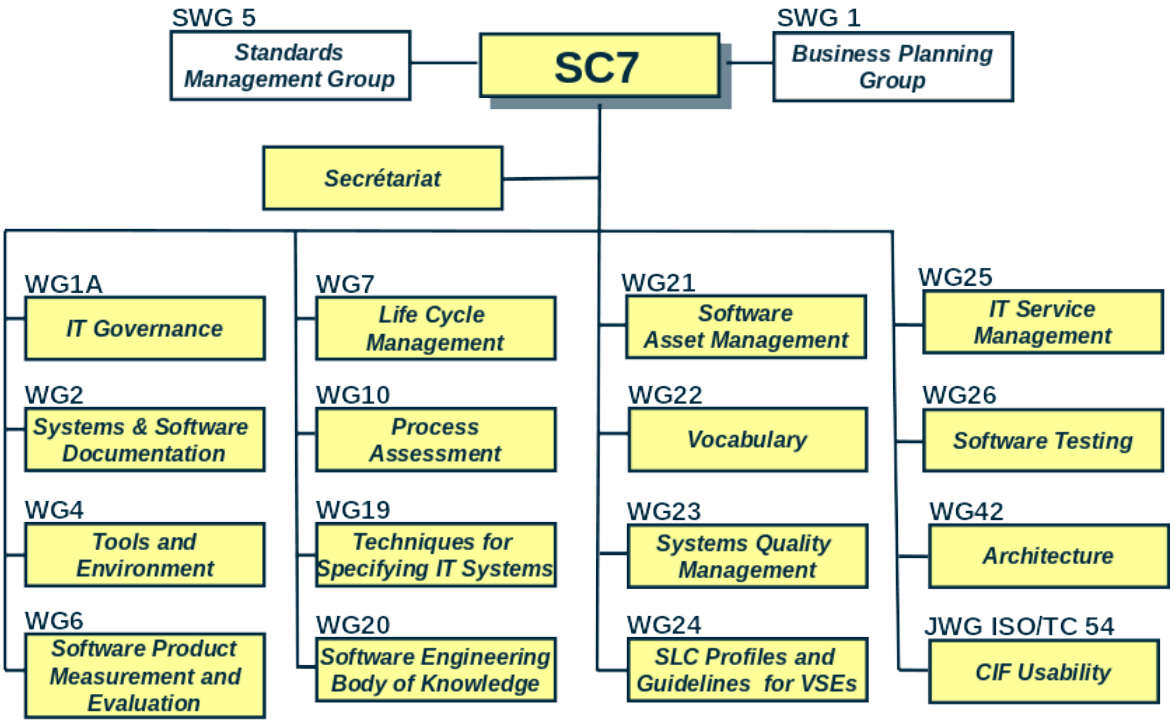
5 ISO SC7

5.1 Structure



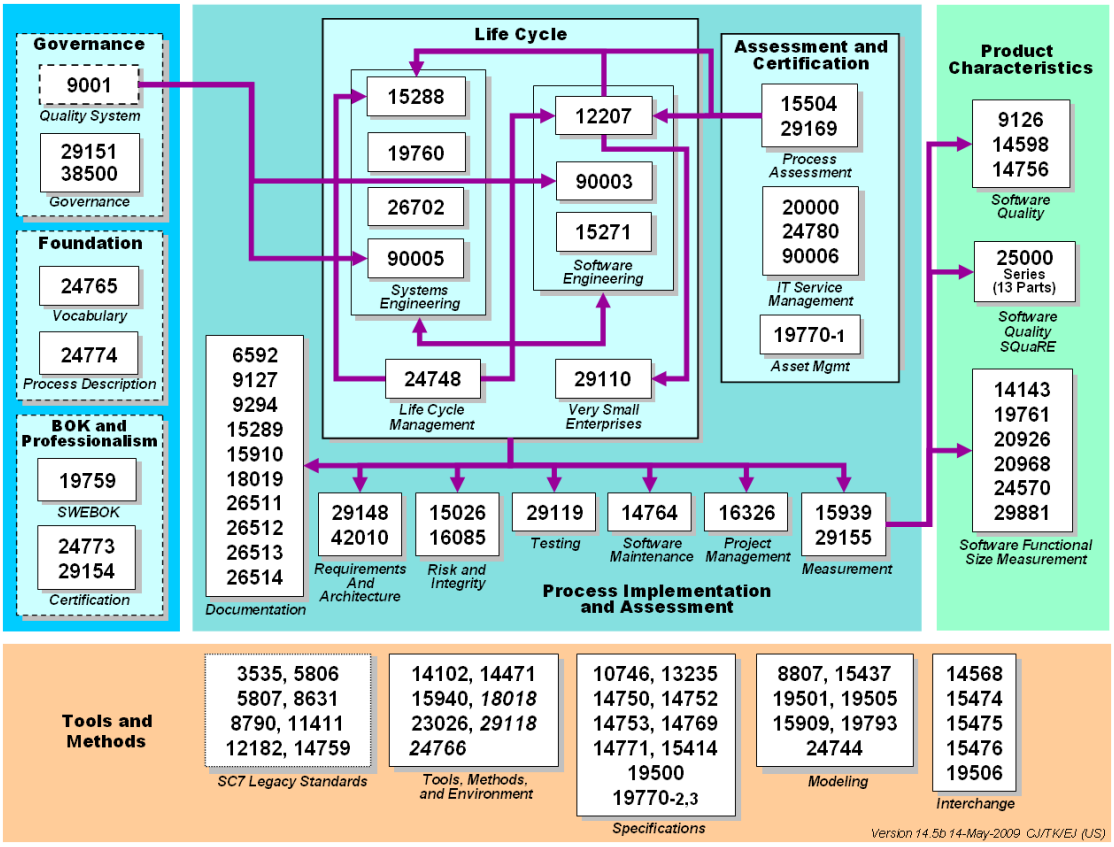
‡ Adapted from Prof. M. Azuma

5.2 Domains



‡ Adapted from Prof. M. Azuma

5.3 Standards



Standards of particular interest

- ISO 9000, family of standards for quality management systems
- ISO 12207, defines the software engineering process, activity, and tasks that are associated with a software life cycle process from conception through retirement
- ISO 15504, also known as SPICE (Software Process Improvement and Capability Determination), is a framework for the assessment of processes

6 ISO 9000

QSM:

- ISO9001 – QSM for Quality Assurance in design, development, production, installation and service
- ISO9002 – QSM for Quality Assurance in production, installation, and servicing
- ISO9003 – QSM for Quality Assurance in final inspection and test

Quality: refers to all features of a product (such as software) which are required by a customer

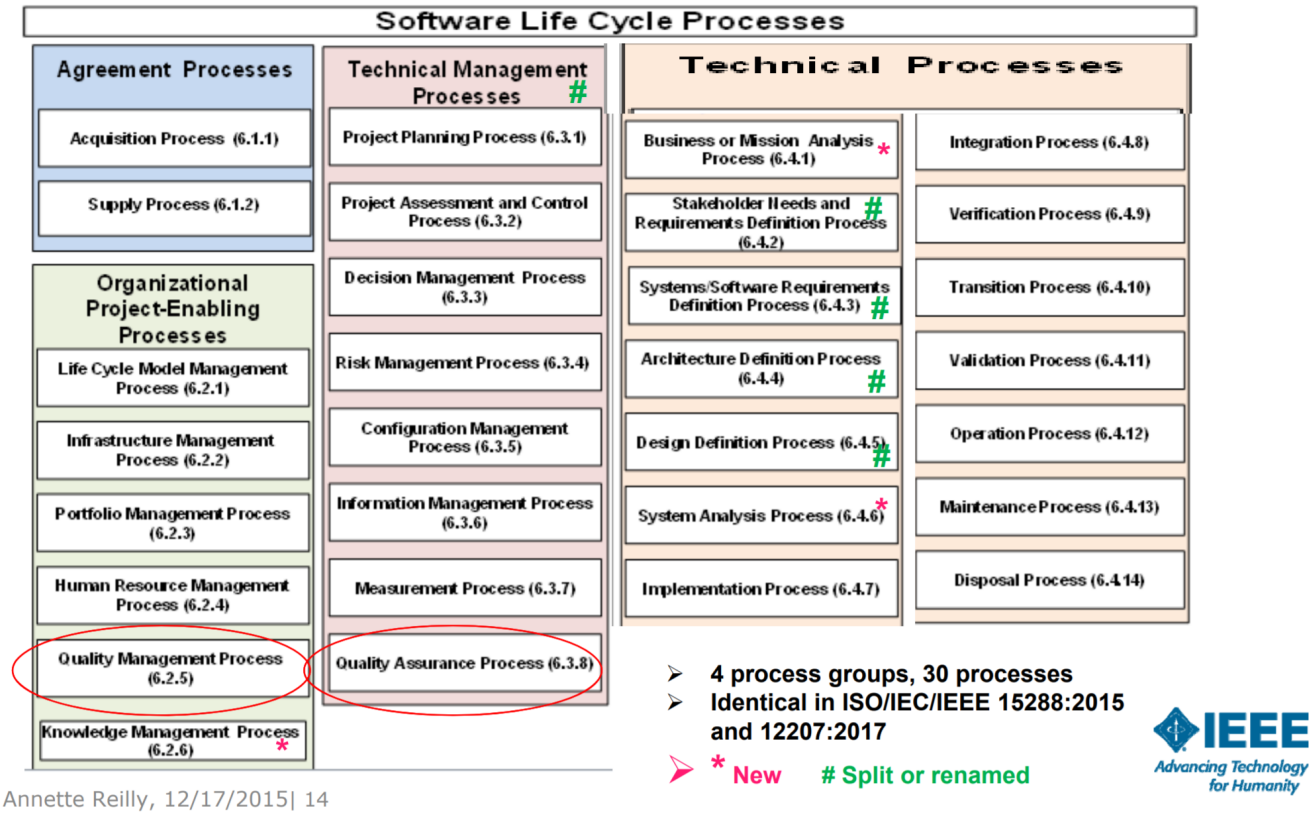
Quality management: covers the organisations approach to ensuring that it produces quality products and complies with the appropriate regulations

7 ISO 12207

- Created to supply a common structure so that the buyers, suppliers, developers, maintainers, operators, managers and technicians involved with the software development use a common language
- It is the standard that defines all the tasks required for developing and maintaining software
- Created in '95, last updated in '17 (ISO 12207:2017)
- Covers the process in the life cycle of software:

  - High level process architecture
  - Activities and tasks
  - Tailored for any organization or project (inc. SME et al)
  - An 'inventory' of processes from which to choose
- This standard does not create a standardised way to create a product
- It is not prescriptive
- Nor does it advocate or enforce a standardised methodology

7.1 ISO 12207:17



8 Process Implementation

- Define or select software life cycle model appropriate to the scope, magnitude, and complexity of the project;
- Select, tailor, and use standards, methods, tools, and programming languages (if not stipulated in contract);
- Develop plans for conducting the activities of the Development process.

9 ISO 15504

Process assessment: What is it?

- A disciplined examination of the processes by an organisation against a set of criteria to determine capability of those processes to perform within quality, cost and schedule goals
- Focus here is on continual, self-improvement

Why bother?

- Identify strengths and weaknesses in current utilisation of processes
- Ongoing development of systems, maturity and growth
- Feeds into the future