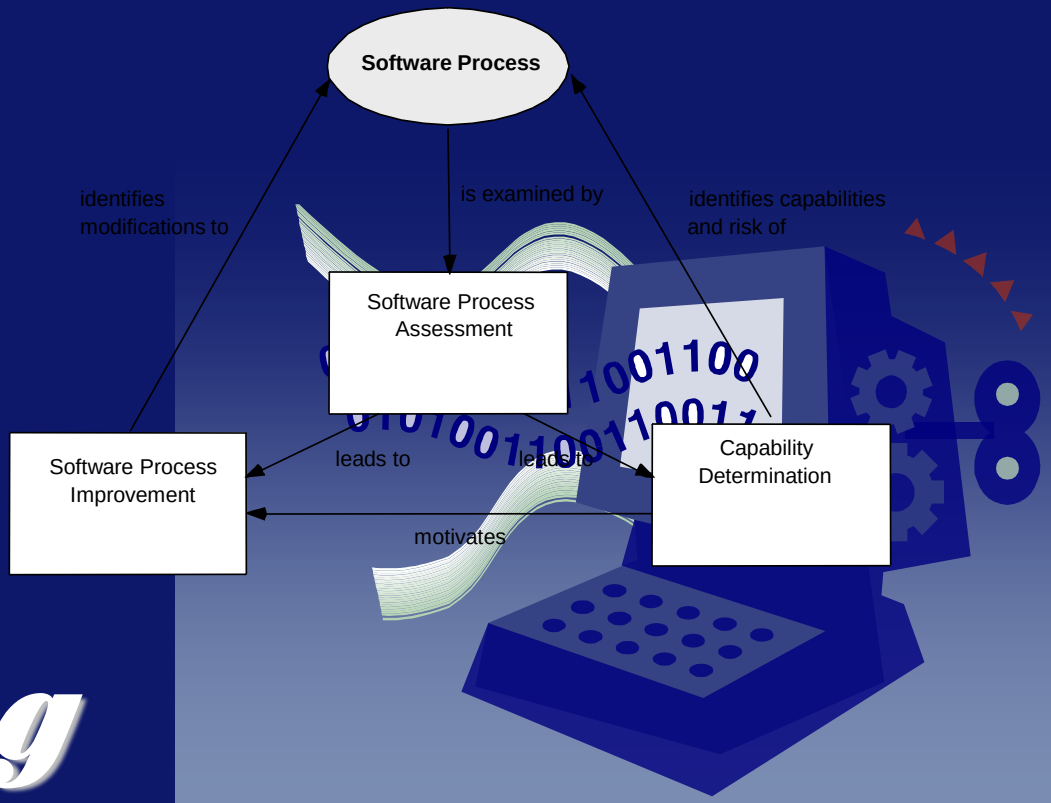
Software Engineering

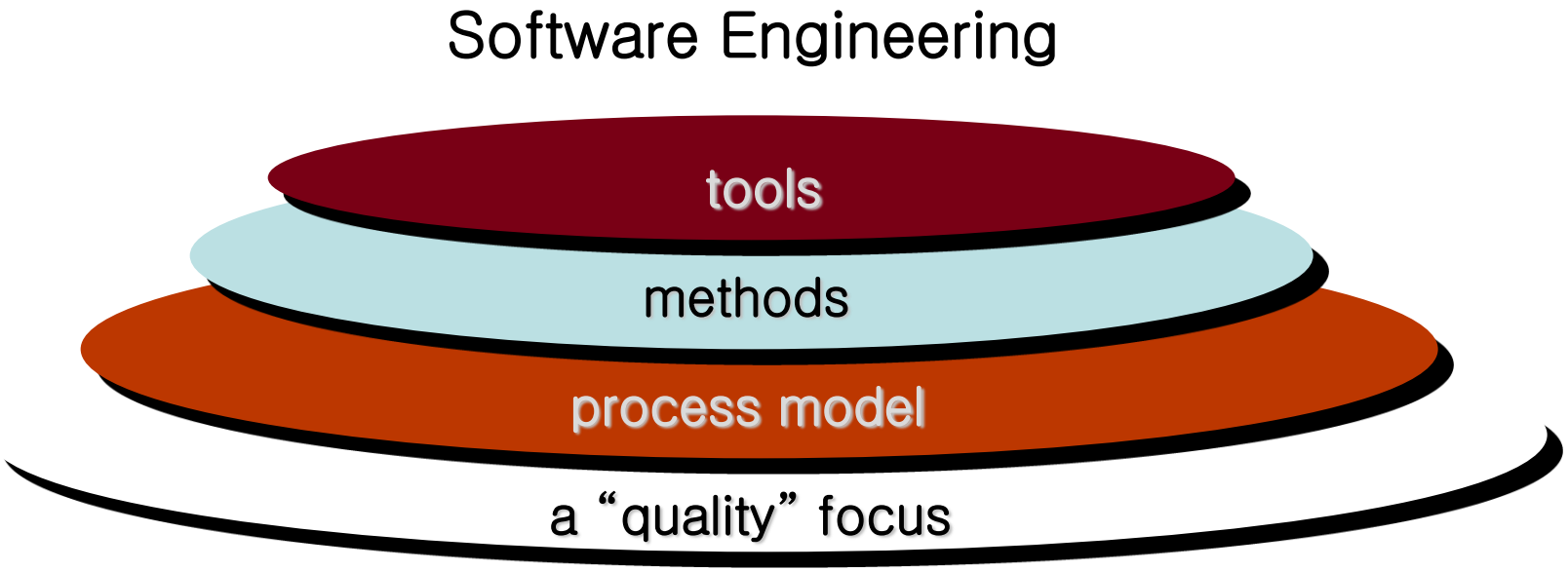


Chapter 2

Process: A Generic View

Moon kun Lee
Division of Electronics & Information Engineering
Chonbuk National University

A Layered Technology

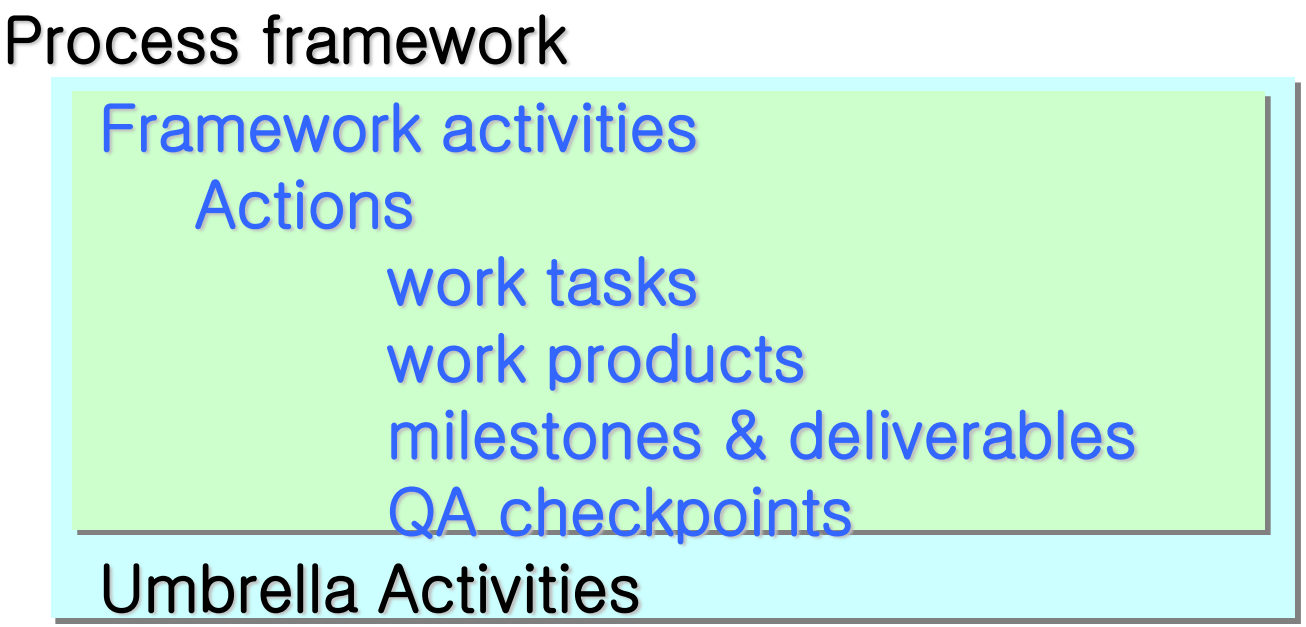


- Definition [IEEE]
  - The application of systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, that is, the application of engineering to software.
  - The study of approaches in the above application

과학 vs. 공학

- **과학**:
  - **자연, 사회 등에 대한 체계적인 지식, 또는 그것을 밝히는 학문.**
  - The observation, identification, description, experimental investigation, and theoretical explanation of phenomena.
- **공학**
  - **공업에 관한 이론과 기술 등을 연구하는 학문.**
  - **공업: 원료를 가공하여 인간 생활에 필요한 여러 가지 재화를 생산하는 산업.**
  - The application of scientific and mathematical principles to practical ends such as the design, manufacture, and operation of efficient and economical structures, machines, processes, and systems

A Process Framework



- *Process framework*: The foundation for a complete software process by identifying a small number of framework activities that are applicable to all software projects.
  - *Software process*: a framework for the tasks that are required to build high-quality software.
  - *Software Engineering Actions*: a collection of related tasks that produced a major software engineering work product.

Framework Activities

- Communication
- Planning
- [Activity] Modeling
  - [Action] Analysis of requirements
    - [work task] requirement gathering
    - [work task] elaboration
    - [work task] negotiation
    - [work task] specification
    - [work task] validation
  - [Action] Design
    - [work task] data design
    - [work task] architecture design
    - [work task] interface design
    - [work task] component-level design
- Construction
  - Code generation
  - Testing
- Deployment

e.g.: Comm.,Requirements gathering

- Small project
  1. Make a list of stakeholders for the project.
  2. Invite all stakeholders to an informal meeting.
  3. Ask each stakeholders to make a list of features and functions required.
  4. Discuss requirements and build a final list.
  5. Prioritize requirements.
  6. Note area of uncertainty.

- Larger, complex project
  1. Make a list of stakeholders for the project.
  2. Interview each stakeholder separately to determine overall wants and needs.
  3. Build a preliminary list of functions, base on stakeholder input
  4. Schedule a series of facilitated requirements gathering meetings.
  5. Conduct meetings.
  6. Produce informal user scenarios as part of each meeting.
  7. Refine use scenarios based on stakeholder feedback.
  8. Build a revised list of stakeholder requirements.
  9. Use quality function deployment techniques to prioritize requirements.
  10. Package requirements so that they can be delivered incrementally.
  11. Note constraints and restrictions that will be placed on the system.
  12. Discuss methods for validating the system.

Umbrella Activities

- Software project management
- Formal technical reviews
- Software quality assurance
- Software configuration management
- Work product preparation and production
- Reusability management
- Measurement
- Risk management

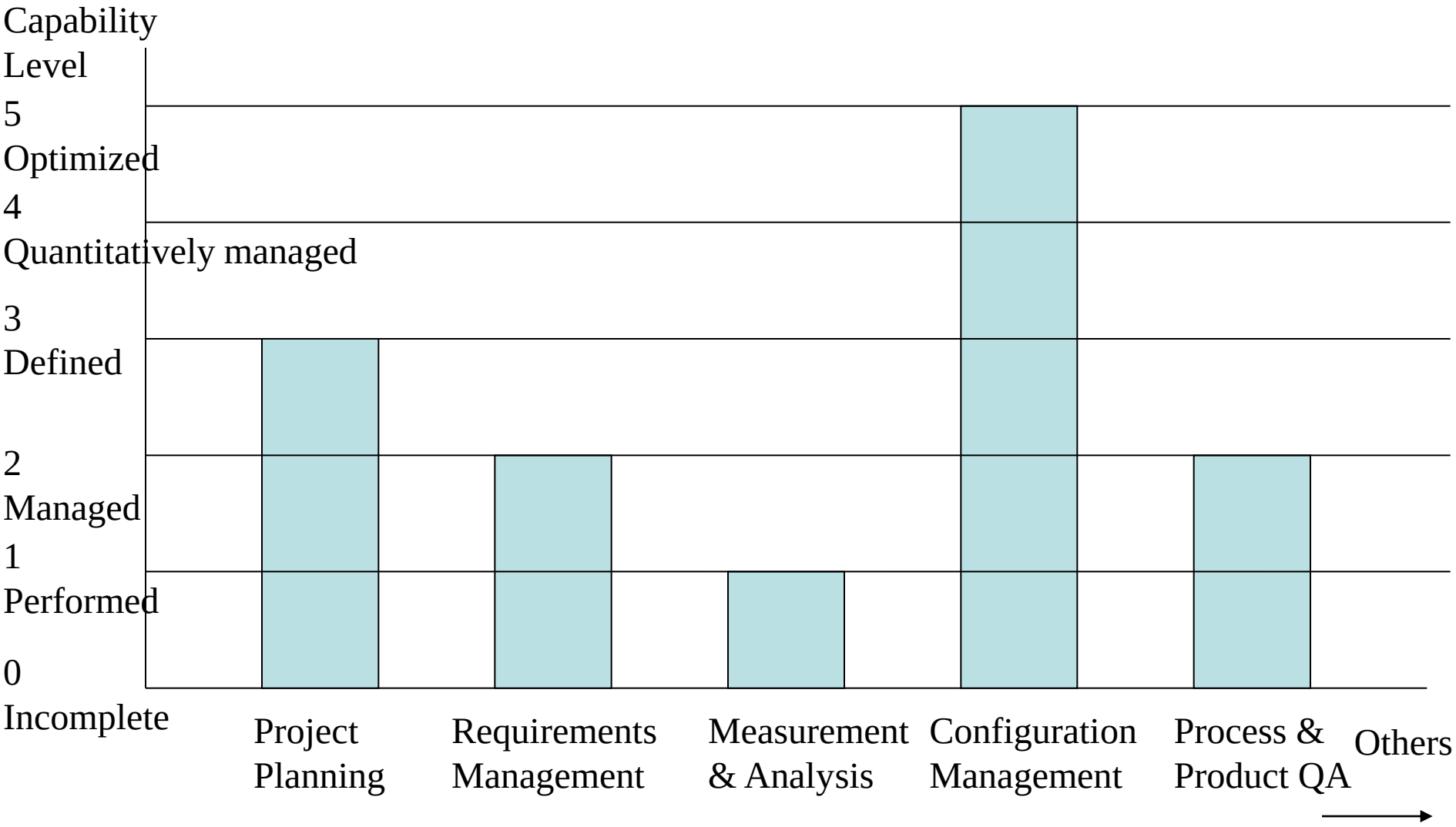
The Process Model: Adaptability

- the framework activities will always be applied on every project ... BUT
- the tasks (and degree of rigor) for each activity will vary based on:
  - the type of project
  - characteristics of the project
  - common sense judgment; concurrence of the project team

The CMMI

- CMU, SEI, *Capability Maturity Model Integration*
- 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.
- Two models:
  - Continuous model (next page)
  - Staged model (Fig. 2.4 on page 63, 6th edition)

CMMI Profile (Continuous Model)



e.g.: Project planning (1 of 8 Process Areas)

- SG (Specific Goal) 1 Establish estimates
  - SP (Specific Practice) 1.1-1 Estimate the scope of the project
  - SP 1.2-1 Establish estimates of work product and task attributes
  - SP 1.3-1 Define project life cycle
  - SP 1.4-1 Determine estimate of efforts and cost
- SG 2 Develop a Project Plan
  - SP 2.1-1 Establish the budge and schedule
  - SP 2.2-1 Identify project risks
  - SP 2.3-1 Plan for data management
  - SP 2.4-1 Plan for project resources
  - SP 2,5-1 Plan for needed knowledge and skills
  - SP 2.6-1 Plan stakeholder involvement
  - SP 2.7-1 Establish the project plan
- SG 3 Obtain commitment to the plan
  - SP 3.1-1 Review plans that affect the project
  - SP 3.2-1 Reconcile work and resource levels
  - SP 3.3-1 Obtain plan commitment

Generic Goals and Practices

- GG1 Achieve specific goals
  - GP 1.1 Perform base practices
- GG2 Institutionalize a managed process
  - GP 2.1 Establish an organizational policy
  - GP 2.2 Plan the process
  - GP 2.3 Provide resources
  - GP 2.4 Assign responsibility
  - GP 2.5 Train people
  - GP 2.6 Manage configurations
  - GP 2.7 Identify and involve relevant stakeholders
  - GP 2.8 Monitor and control the process
  - GP 2.9 Objectively evaluate adherence
  - GP 2.10 Review status with higher level management
- GG 3 Institutionalize a defined process
  - GP 3.1 Establish a defined process
  - GP 3.2 Collect improvement information
- GG4 Institutionalize a quantitatively managed process
  - GP 4.1 Establish quantitative objectives for the process
  - GP 4.2 Stabilize subprocess performance
- GG5 Institutionalize an optimized process
  - GP 5.1 Ensure continuous process improvement
  - GP 5.2 Correct root causes of problems

Process Patterns

- Process patterns define a set of activities, actions, work tasks, work products and/or related behaviors
- A template is used to define a pattern
  - A consistent method for describing an important characteristics of the software process (, activity, action or task)
  - By combining patterns, a software team construct a process that must meet the needs of a project.
- Typical examples:
  - Customer communication (a process activity)
  - Analysis (an action)
  - Requirements gathering (a process task)
  - Reviewing a work product (a process task)
  - Design model (a work product)

Some template for describing process pattern

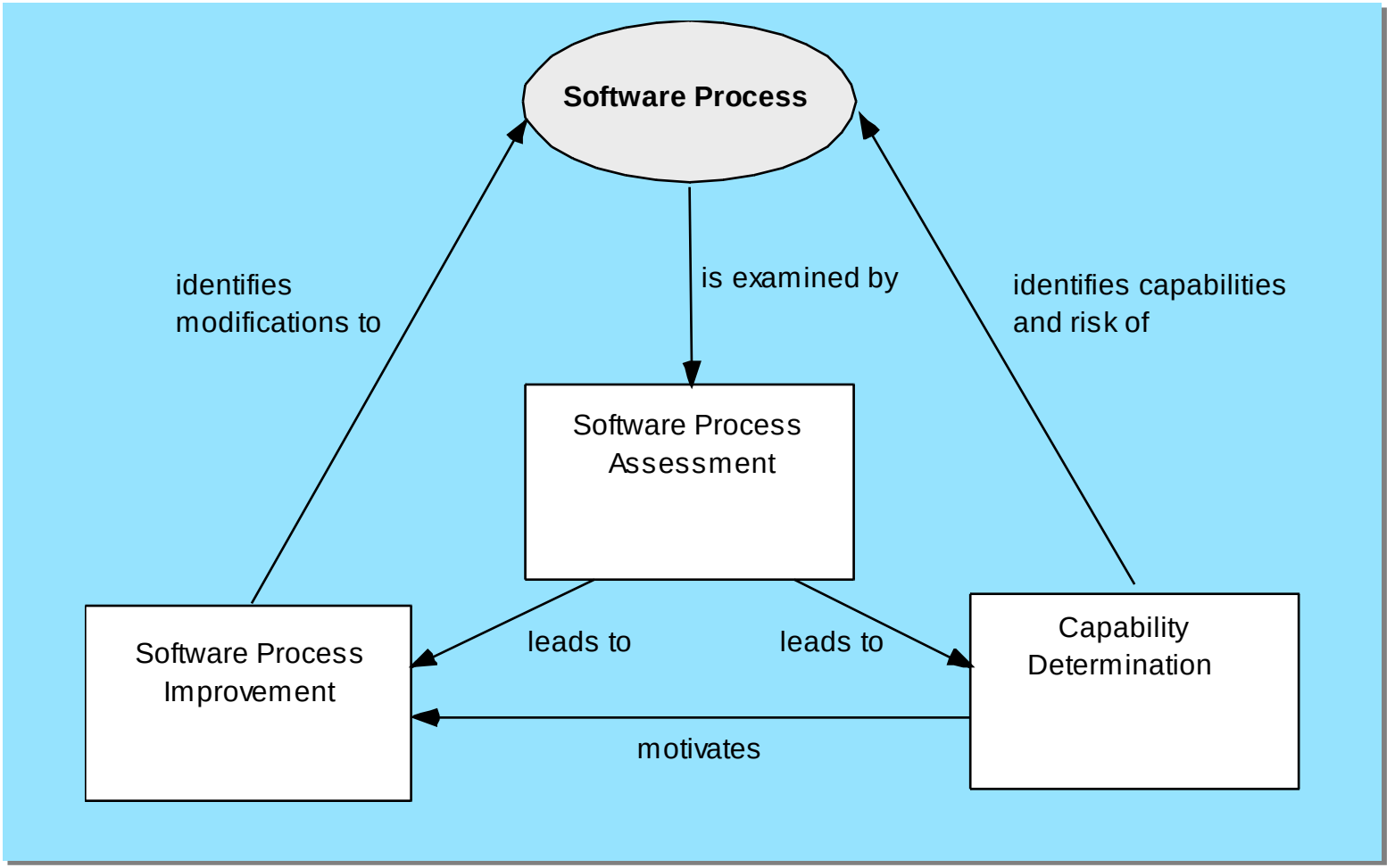
- Pattern name
- Intent
- Type
  - Task pattern: SW action or work task; requirements gathering
  - Stage pattern: a framework activity; communication
  - Phase pattern: a sequence of framework activities; prototyping
- Initial context
- Problem
- Solution
- Resulting context
- Related patterns
- Know uses/examples

e.g.: prototyping

- Pattern name: prototyping
- Intent:
  - to build a model that can be assessed iteratively by stakeholders in an effort to identify or solidify software requirements.
- Type: Phase pattern
- Initial context
  - Stakeholders have been identified
  - A mode of communication team has been established
  - The overriding problem to be solved has been identified by stakeholders
  - An initial understanding of project scope, basic business requirements, and project constraints has been developed.
- Problem:
  - Requirements are hazy and nonexistent, yet there is clear recognition that there is a problem, and the problem must be addressed with a software solution.
  - Stakeholders are unsure of what they want; that is, they cannot describe software requirements in any detail.
- Solution
  - A description of the prototyping process (Chapter 3)
- Resulting context
  - A software prototype that identifies basic requirements (i.e., modes of interaction, computational features, processing functions) is approved by stakeholders.
    - The prototype may evolve through a series of increments to become the production software or
    - The prototype may be discarded and the production software built using some other process pattern.
- Related patterns
  - Customer-communication; iterative design; iterative development, customer assessment; requirement extraction
- Know uses/examples
  - Prototyping is recommended when requirements are uncertain.

Process Assessment

- The process should be assessed to ensure that it meets a set of basic process criteria that have been shown to be essential for a successful software engineering.
- Many different assessment options are available:
  - SCAMPI (Standard CMMI for Process Improvement)
  - CBA IPI (CMM-Based Appraisal for Internal Process Improvement)
  - SPICE
  - ISO 9001:2000
    - Plan
    - Do
    - Check
    - Act


Software Process
identifies
modifications to
is examined by
identifies capabilities
and risk of
Software Process
Assessment
Software Process
Improvement
leads to
leads to
Capability
Determination
motivates

Personal Software Process (PSP)

- Recommends five framework activities:
  - Planning
  - High-level design
  - High-level design review
  - Development
  - Postmortem
- Stresses the need for each software engineer to identify errors early and as important, to understand the types of errors

Team Software Process (TSP)

- Each project is “launched” using a “script” that defines the tasks to be accomplished
- Teams are self-directed
- Measurement is encouraged
- Measures are analyzed with the intent of improving the team process

Launch Script (Recommended)

- Review project objects with mgmt and agree on and document team goals
- Establish team roles
- Define the team's development process
- Make a quality plan and set quality targets
- Plan for the needed support facilities
- Produce an overall development strategy
- Make a development plan for the entire project
- Make detailed plans for each engineering for the next phase
- Merge the individual plans into a team plan
- Rebalance team workload to achieve a minimum overall schedule
- Access project risks and assign tracking responsibility for each key risk.

The Primary Goal of Any SW Process: *High Quality*

Remember:

High quality = project timeliness

Why?

Less rework!

Process Modeling Tools

- Igrafx Process Tool
  - Corel Corporation
  - www.igrafx.com/products/process
- Objexis Team Portal
  - Objexis Corporation
  - www.objexis.com