Introduction of Software Engineering

Chapter 2:

Software Process Structure

VŨ THỊ TRÀ

Software Process Structure

1. What is it?
   - ✓ go through a series of predictable steps – a road map that helps you create a timely, high-quality result.
2. Who does it?
   - ✓ Software engineer and their managers adapt the process to their needs and then follow it.
   - ✓ Those have requested the software have a role to play in the process of defining, building, and testing it.
3. Why is it important?
   - ✓ A modern software engineering approach must be "agile.“, so that the activities, controls, and work products must be appropriate for the project team and the product.

Software Process Structure

4. What are steps?
   - ✓ A process might be appropriate for creating software that you are building.
5. What are the work product?
   - ✓ the programs, documents, and data that are produced as a consequence of the activities and tasks defined by the process.
6. How do I ensure that I've done it right?
   - ✓ a number of software process assessment mechanisms that enable organizations to determine the "maturity" of their software process.
   - ✓ the quality, timeliness, and long-term viability of the product you build are the best indicators of the efficacy of the process.

The Definition of Software Process

- *a framework for the activities, actions, and tasks that are required to build high-quality software.*

CONTENTS

- A generic Process Model
- Defining Framework Activity
- Identifying Task Set
- Process Patterns
- Process Assessment and Improvement

A Generic Process Model

- a process was defined as a collection of work activities, actions, and tasks that are performed when some work product is to be created
  - ✓ An activity = a set of actions
  - ✓ An action = a set of tasks
- After a work product is produced, the quality assurance points that will be required, and the milestones will be used to indicate progress.

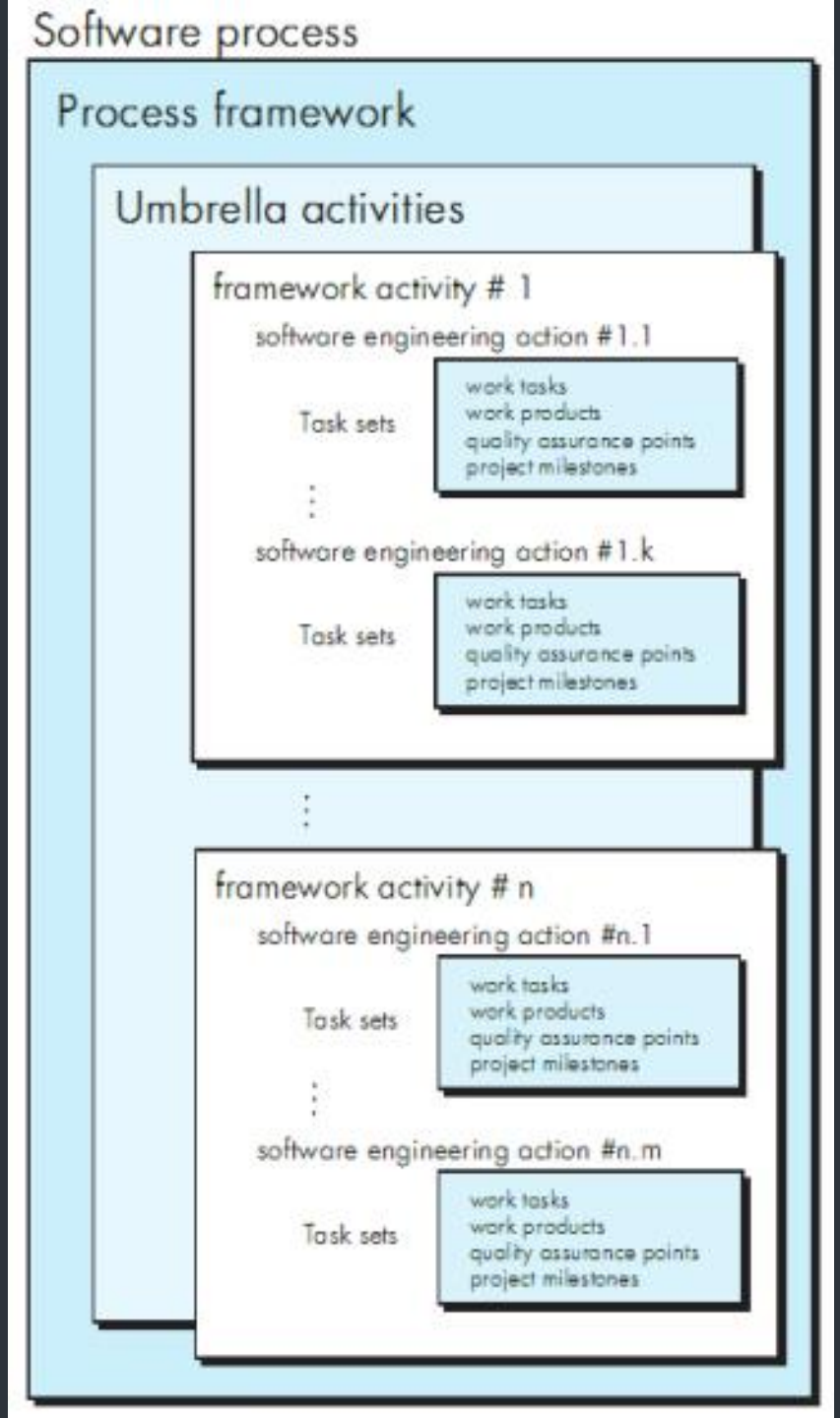
The Five Generic Process Activities

1. Communication
   - ✓ gather requirements that help define software
2. Planning
   - ✓ define scopes, the technical task, the risks, the resources, the work product, the work schedule
3. Modelling
   - ✓ analysis and design
4. Construction
   - ✓ design must be built, code and test
5. Deployment
   - ✓ complete and deliver product

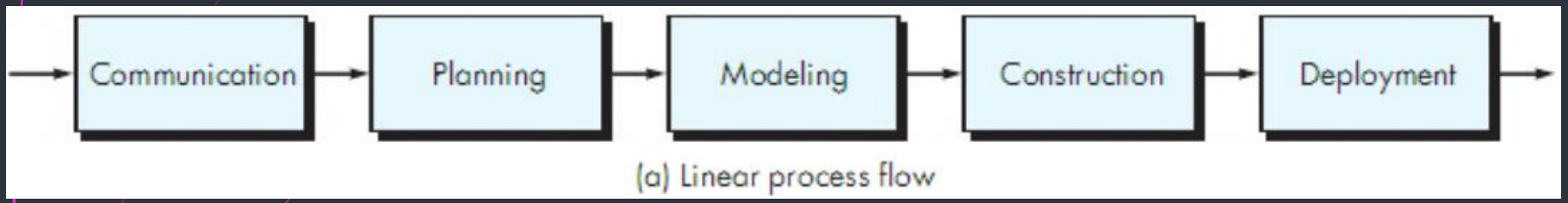
A Generic Process Framework: A set of umbrella activities

- Project tracking and control
- Risk management
- Quality assurance
- Configuration management
- Technical reviews
- *…and others are applied throughout the process*

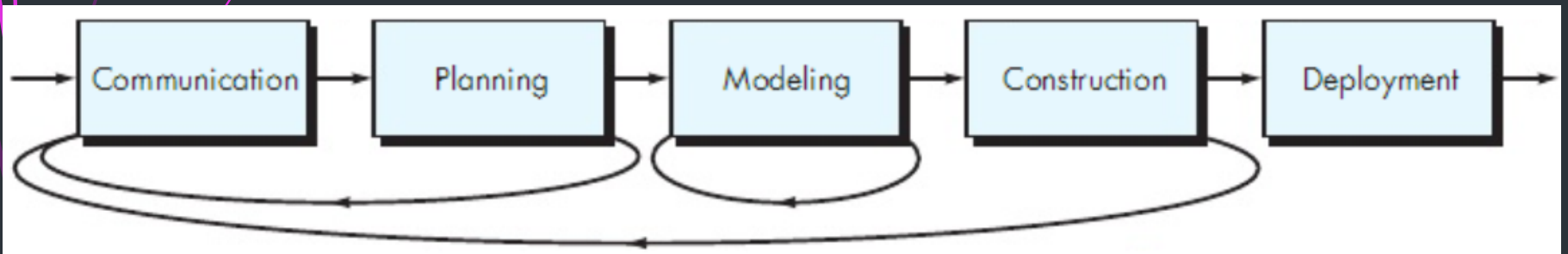
A Software Process Framework



Process Flow

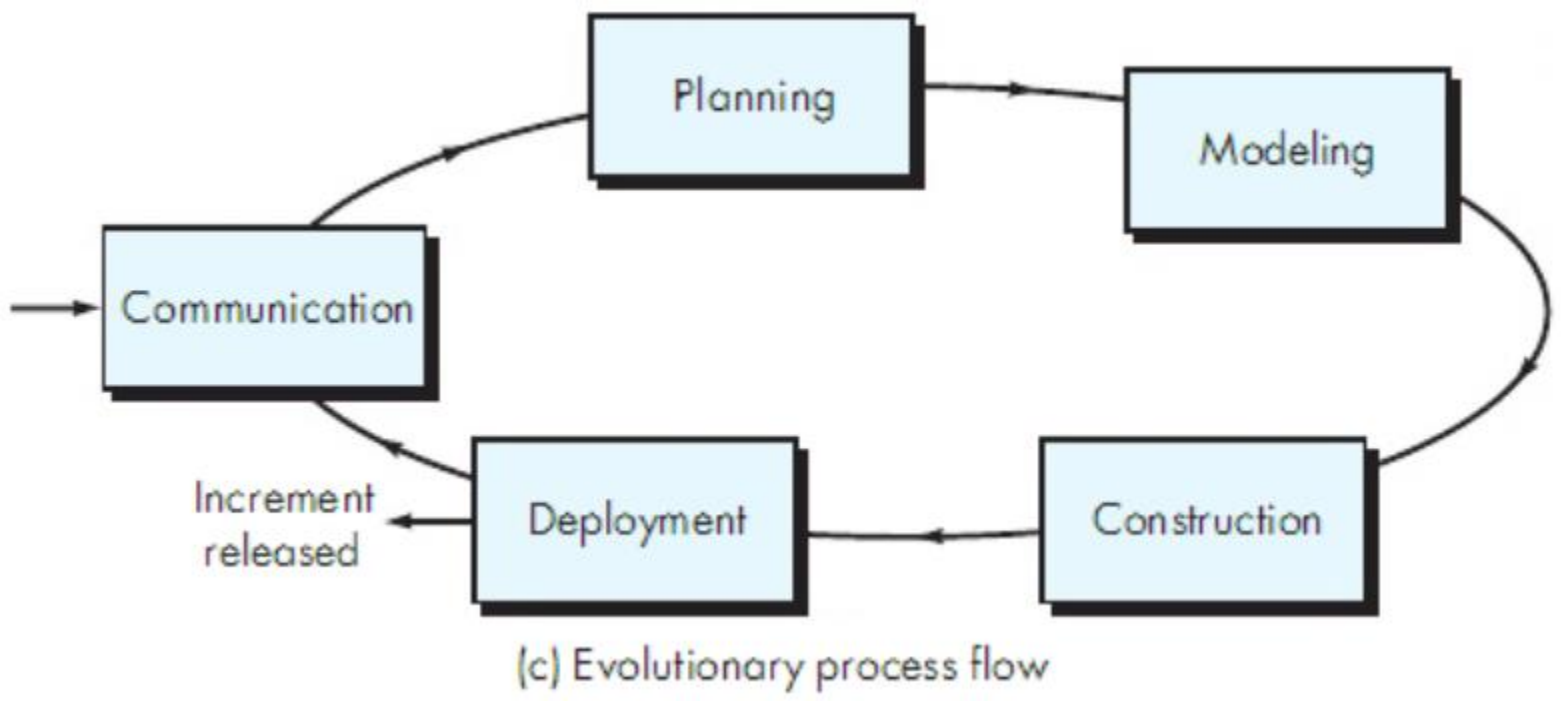


(a) Linear process flow



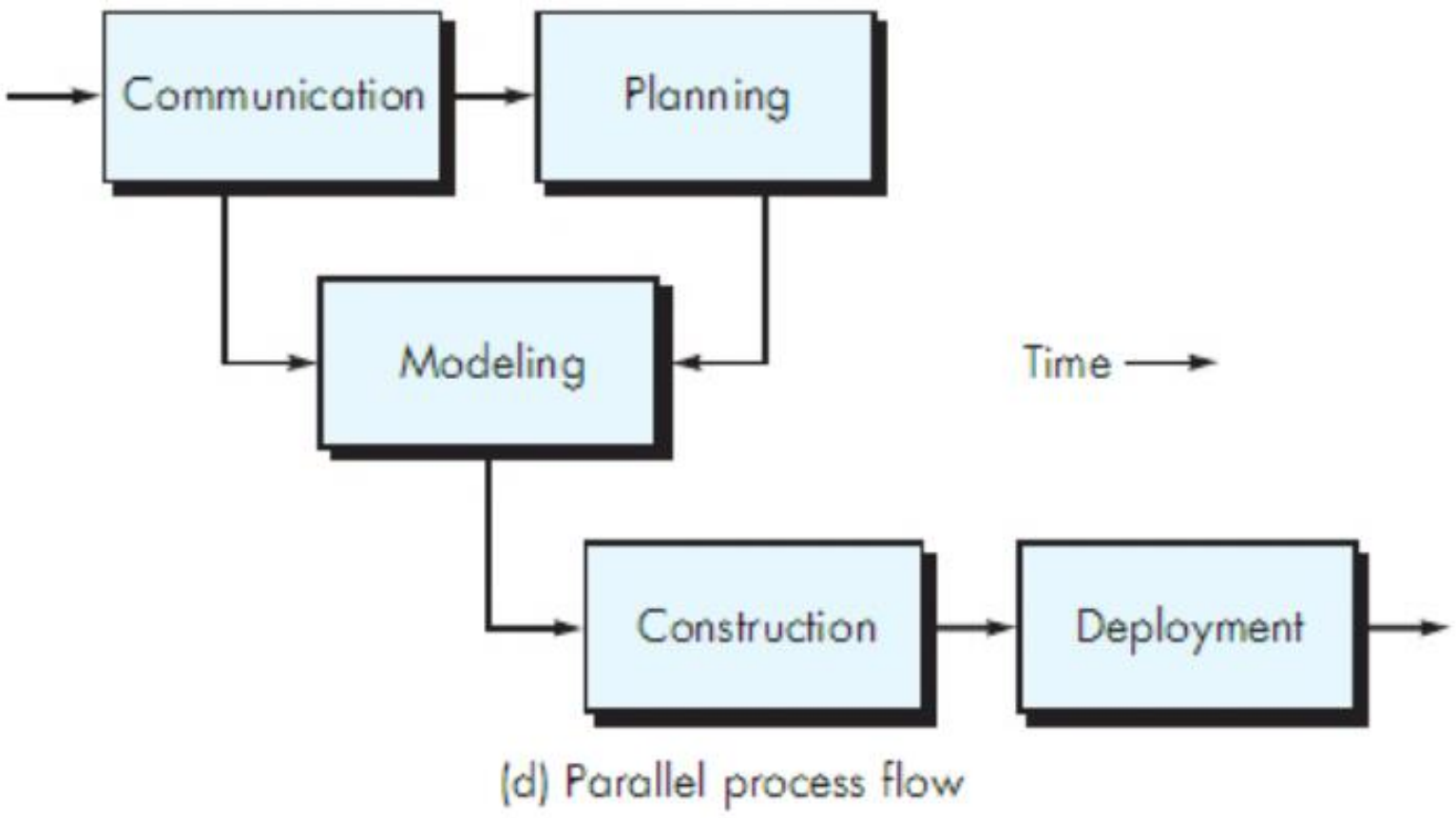
(b) Iterative process flow

Process Flow



(c) Evolutionary process flow

Process Flow



(d) Parallel process flow

CONTENTS

Defining a Framework Activity

- For a small software project, there is only a necessary action is *phone* conversation and the work tasks (*the task set*) may include:
  1. Make contact with stakeholder via telephone.
  2. Discuss requirements and develop notes.
  3. Organize notes into a brief written statement of requirements.
  4. Email to stakeholder for review and approval.

Defining a Framework Activity

- For a more complex software project with many stakeholders, each with a different set of (sometime conflicting) requirements, the communication activity might have six distinct actions:
  1. Inception
  2. Elicitation
  3. Elaboration
  4. Negotiation
  5. Specification
  6. Validation

  → *Each of these software engineering actions would have many work tasks and a number of distinct work products.*

CONTENTS

Identifying a Task Set

- a collection of software engineering that includes:
  1. work tasks
  2. related work products
  3. quality assurance points
  4. project milestones

Task Set

- *The actual work to be done to accomplish the objectives of a software engineering action.*
- Ex: elicitation (more commonly called "*requirements gathering*") is an important software engineering action that occurs during the communication activity.

→ *The goal of requirements gathering is to understand what various stakeholders want from the software that is to be built.*

Task Set

- For a small, relatively simple project, the task set for requirements gathering might look like this:
  1. Make a list of stakeholders for the project.
  2. Invite all stakeholders to on informal meeting.
  3. Ask each stakeholder to make a list of features and functions required.
  4. Discuss requirements and build a find list.
  5. Prioritize requirements.
  6. Note areas of uncertainty.

Task Set

- For a larger, more complex software project, a different task set would be required. It might encompass the following work tasks:
  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 and features based on stakeholder input.
  4. Schedule a series of facilitated application specification meetings.
  5. Conduct meetings.
  6. Produce informal user scenarios as part of each meeting.

Task Set

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.

- Both of these task sets achieve requirements gathering: but they are quite different in their depth and formality. The software team chooses the task set that will allow it to achieve the goal of each action and still maintain quality and agility.

Tip for Identifying a Task Set

Different project demand different task set.

The software team chooses the task set based on problem and project characteristics.

--> A software engineering action can be adapted to

- ✓ *the specific needs of the software project*
- ✓ *the characteristics of the project team.*

CONTENTS

Process Pattern

- *A process pattern* describes a process-related problem that is encounter during software engineering work:
  - ✓ identify the environment
  - ✓ suggest one or more proven solutions
- *A process pattern* provides *a template*
  - ✓ *a consistent method* for describing problem solutions

Process Pattern

1. Pattern Name
2. Forces
   - ✓ the environment
   - ✓ the issues
3. Type
   - ✓ Stage pattern
   - ✓ Task pattern
   - ✓ Phase pattern
4. Initial Context
   - ✓ the entry state

Process Pattern

5. Problem
6. Solution
   - ✓ how to implement the pattern successfully
7. Resulting Context
   - ✓ the conditions that will result
8. Related Patterns
   - ✓ a list of all process patterns
9. Known Uses and Examples

CONTENTS

- A generic Process Model
- Defining Framework Activity
- Identifying Task Set
- Process Patterns
- **Process Assessment and Improvement**

Process Assessment and Improvement

- *Assessment attempts to understand the current state of the software process with the intent of improving it.*

"Software organizations have exhibited significant shortcomings in their ability to capitalize on the experiences gained from completed projects".

NASA

Process Assessment and Improvement proposed

- Standard CMMI Assessment Method for Process Improvement (SCAMP)
- CMM-Based Appraisal for Internal Process Improvement (CBA IPI)
- SPICE (ISO/IEC15504)
- ISO 9001:2000for Software