CH1. System Analysis and Design with UML

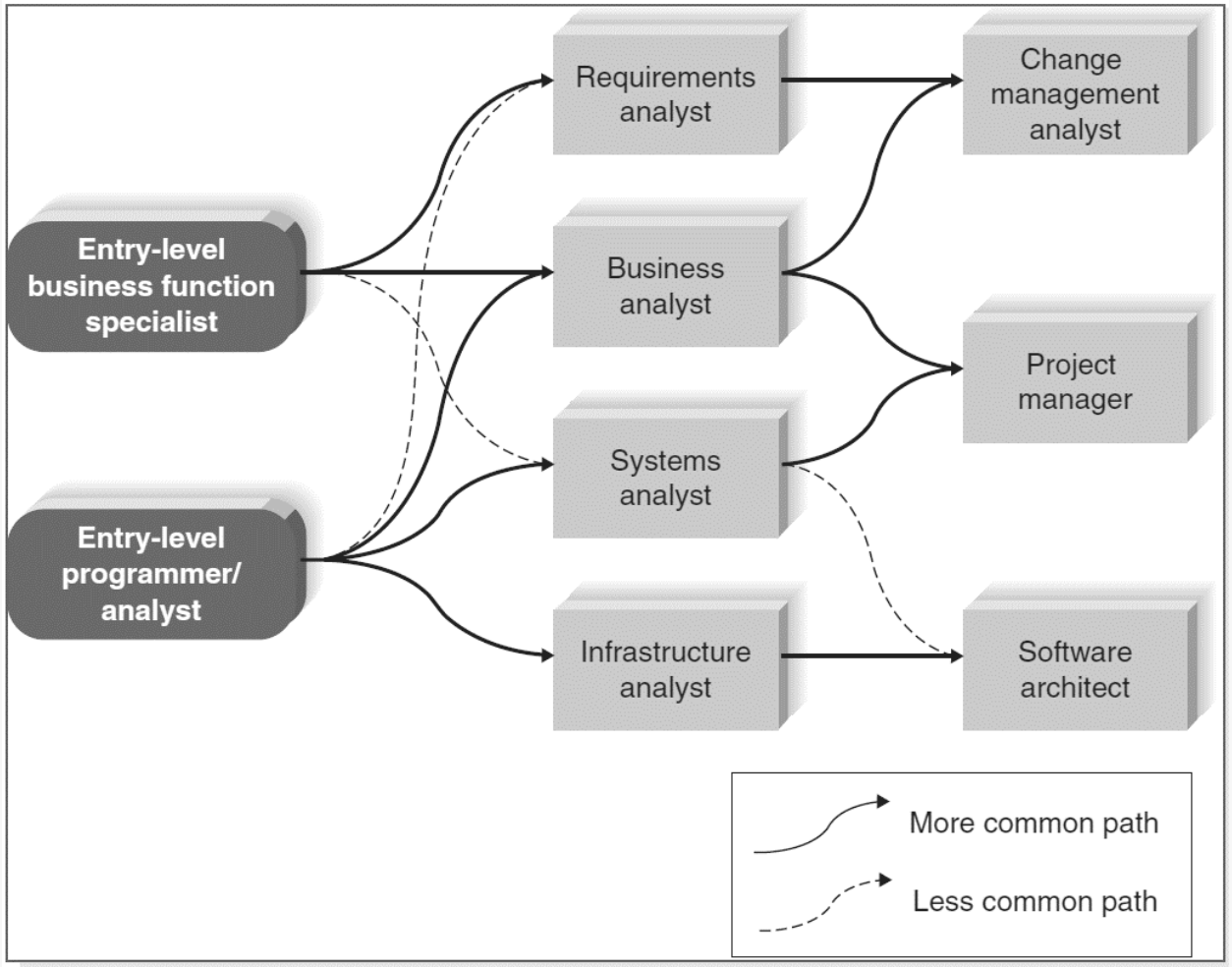
School of Computer Science
Prof. Euijong Lee

Objectives

- What is software development?
- Understand the fundamental system development cycle
- Understand software development methodologies

Career paths for system development

- **What we can be in the future?**

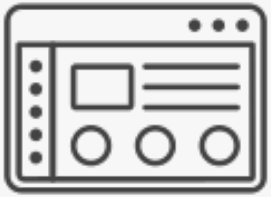


(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. *Systems analysis and design*. John wiley & sons.)

What is software development?

Requirements

Final System

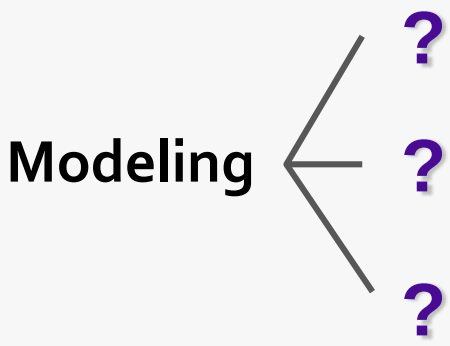


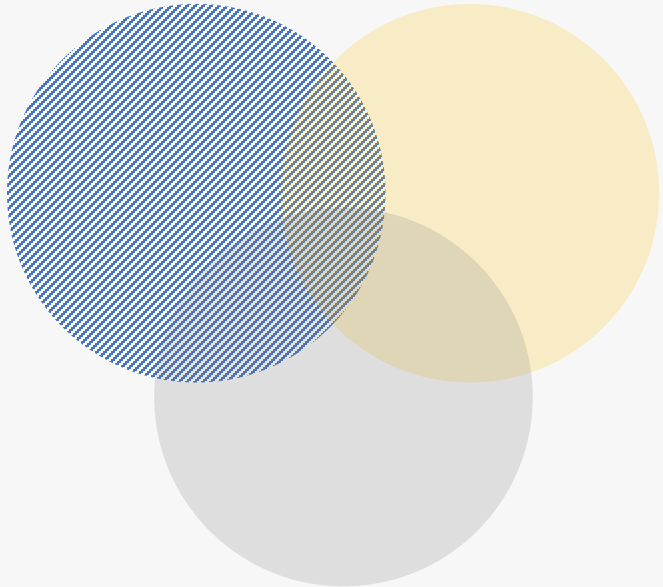
Requirements = Final System

Software Development is Continuous Modeling Activities

What is Modeling?

- What is Modeling ?
- What is Disciplined Approach for Modeling ?

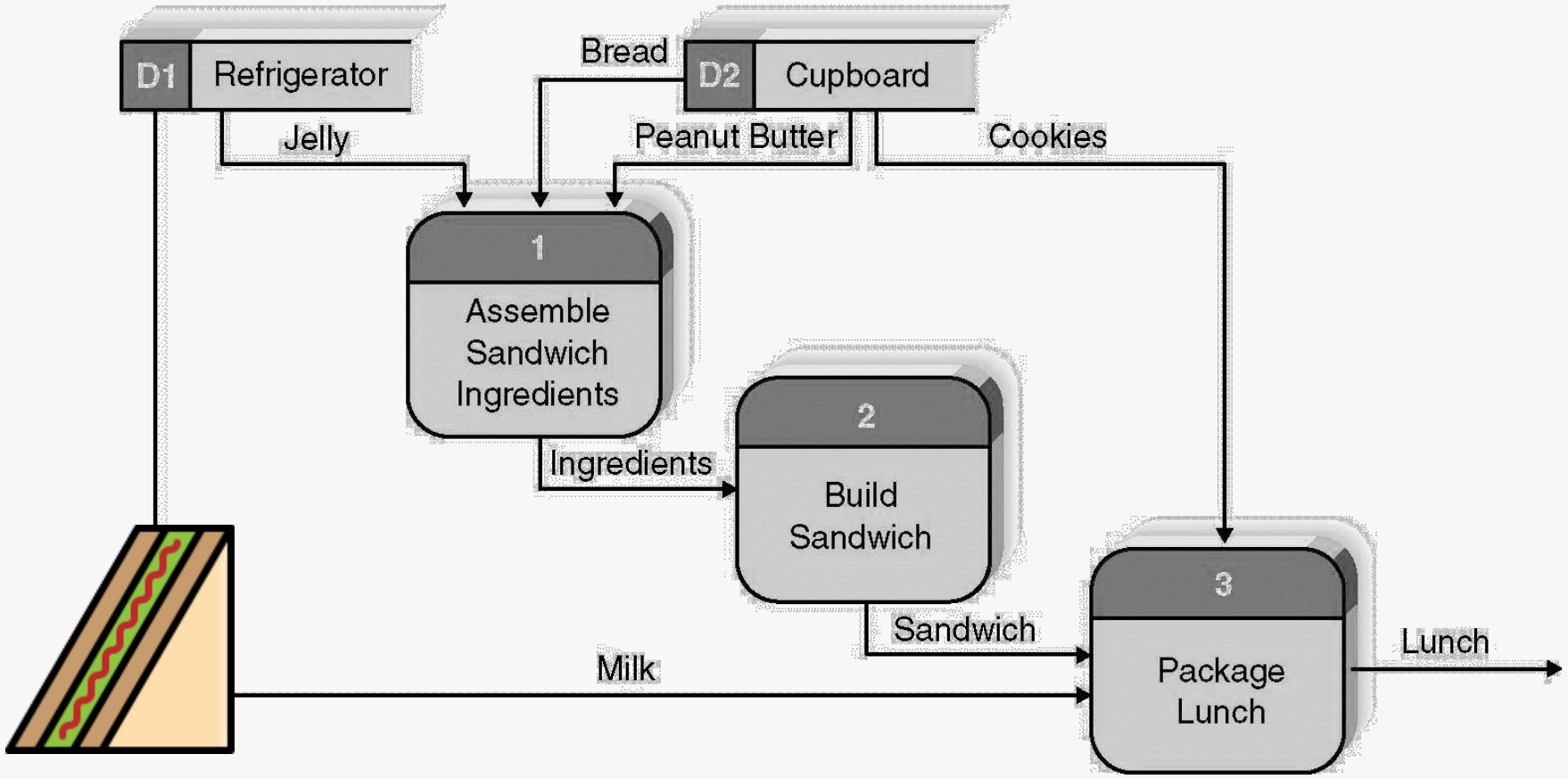




Development Process

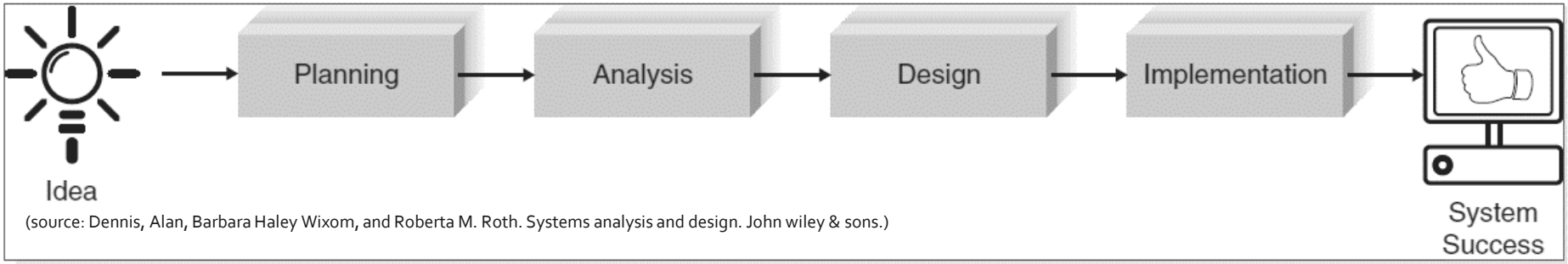
- **Process?**

  1). A 'simple' process for making lunch



Development Process

- **Software Development Life Cycle – SDLC**



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

Planning

Why an information system should be built ?
How the project team will go about building it ?

Analysis

What the system will do to support the user's needs ?

Design

How the system will operate in terms of the hardware, software, and network infrastructure ?

Implementation

System is actually built using programming language and its platform.

Activities in Each Phase

Planning Phase

- Identifying business value
- Analyze feasibility
- Develop work plan
- Staff the project
- Control and direct project

Analysis Phase

- Requirements gathering
- Analysis: Brain storming
- Process modeling
- Data modeling

Design Phase

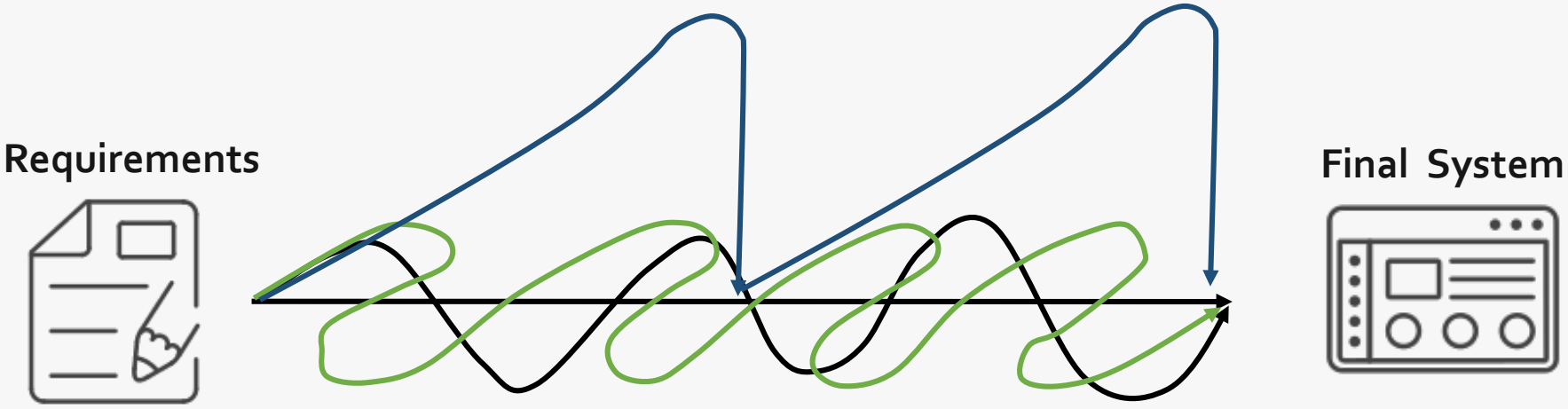
- Architectural design
- Interface design
- Database and file design
- Program design

Implementation Phase

- Construction
- Installation

Software Development Process

- Software Development is Continuous Modeling
- How to perform those activities ?



Software Process Models (1)

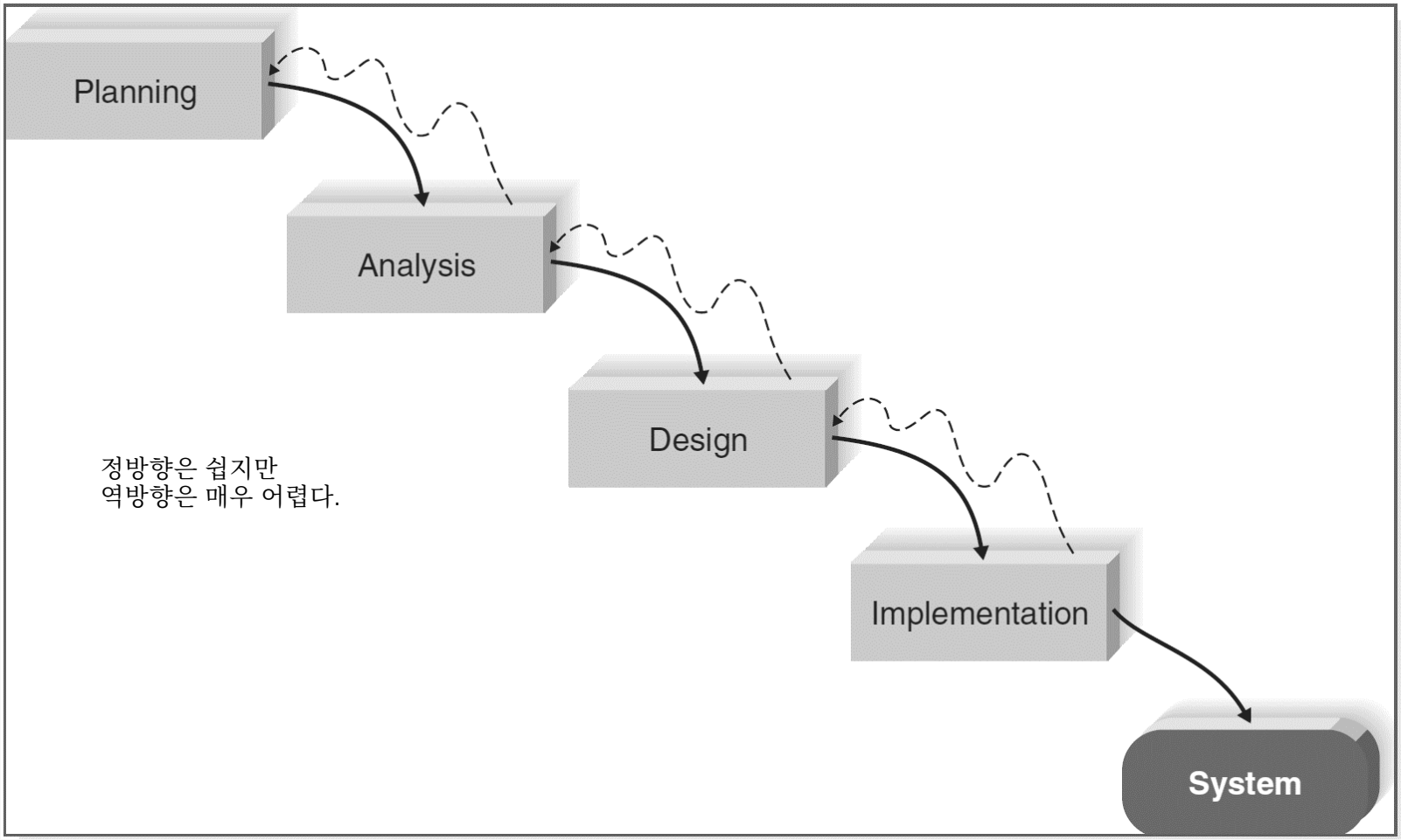
- **Structured Development**
  - **Projects move methodically from one to the next step**
  - **Generally, a step is finished before the next one begins**
  - **Representative models**
    - Waterfall model
    - V-model
    - Parallel development model

S.E

프로젝트가 각각의 단계별로 넘어갈 때, 딱딱 체계적으로 짜여져 있는 경우

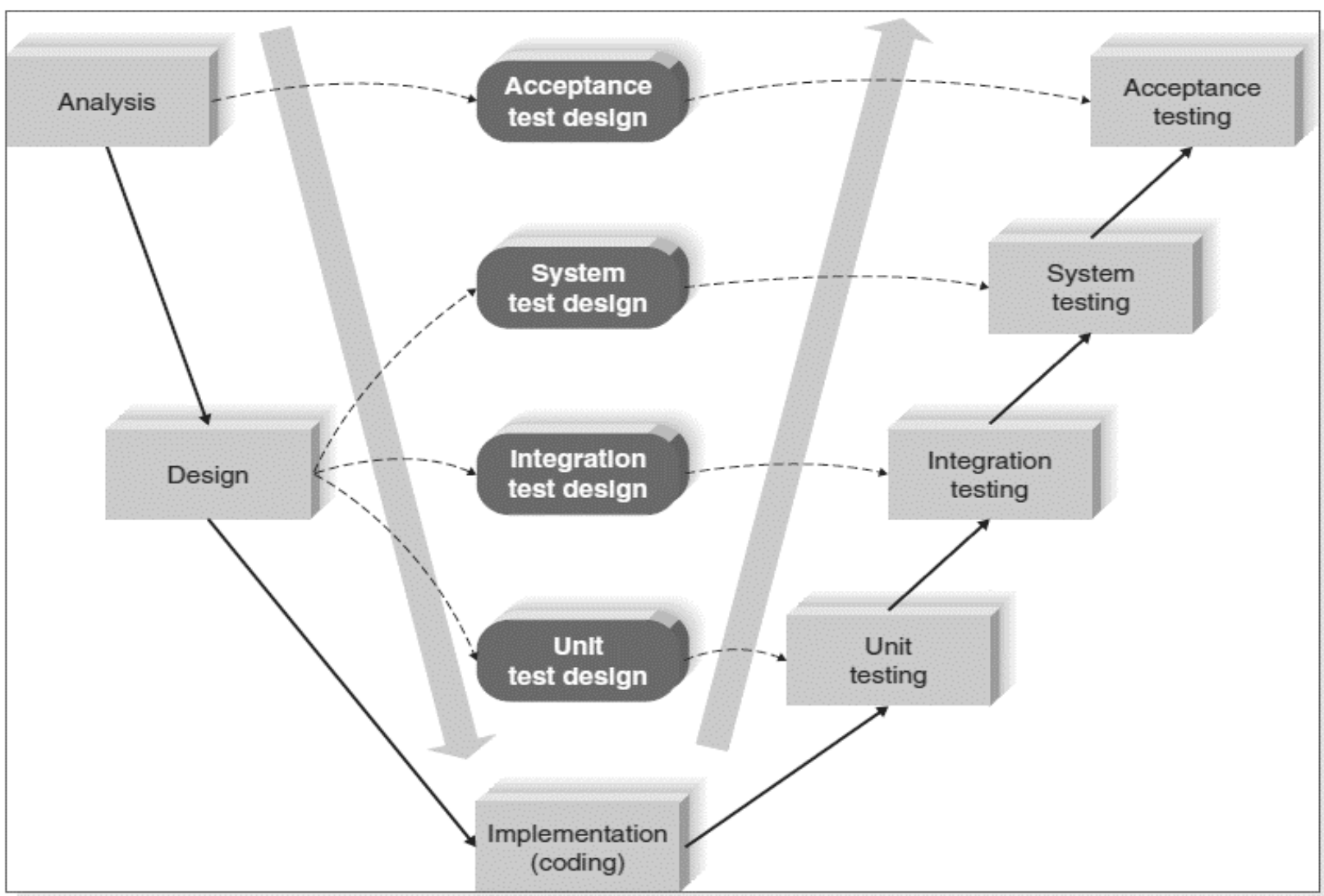
: Waterfall model

절차적인 개발 방법론



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

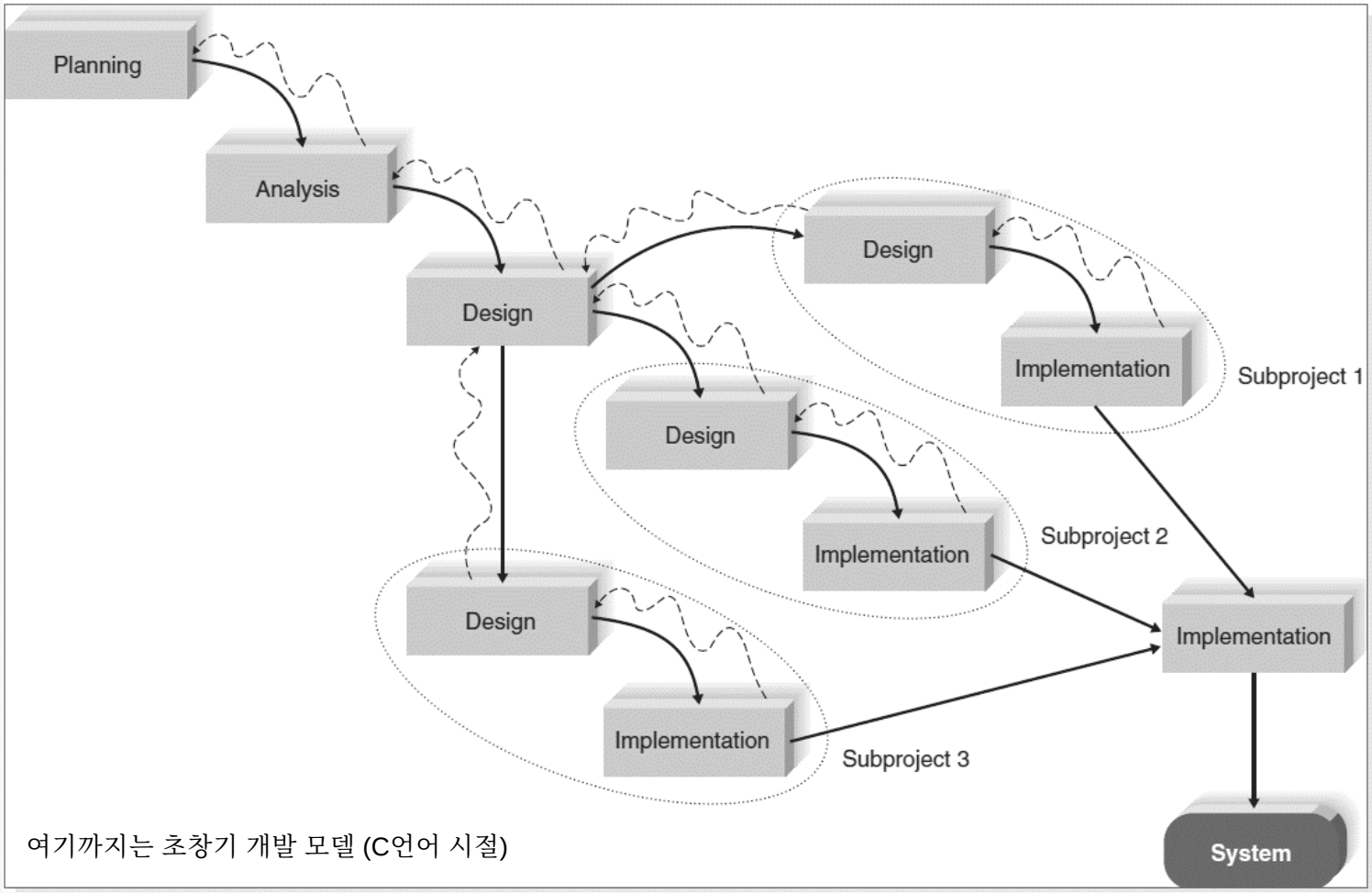
: V-model



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

: Parallel Development Model

Waterfall model이랑 별 차이는 없지만 병렬적인 모습이 추가됨



여기까지는 초창기 개발 모델 (C언어 시절)

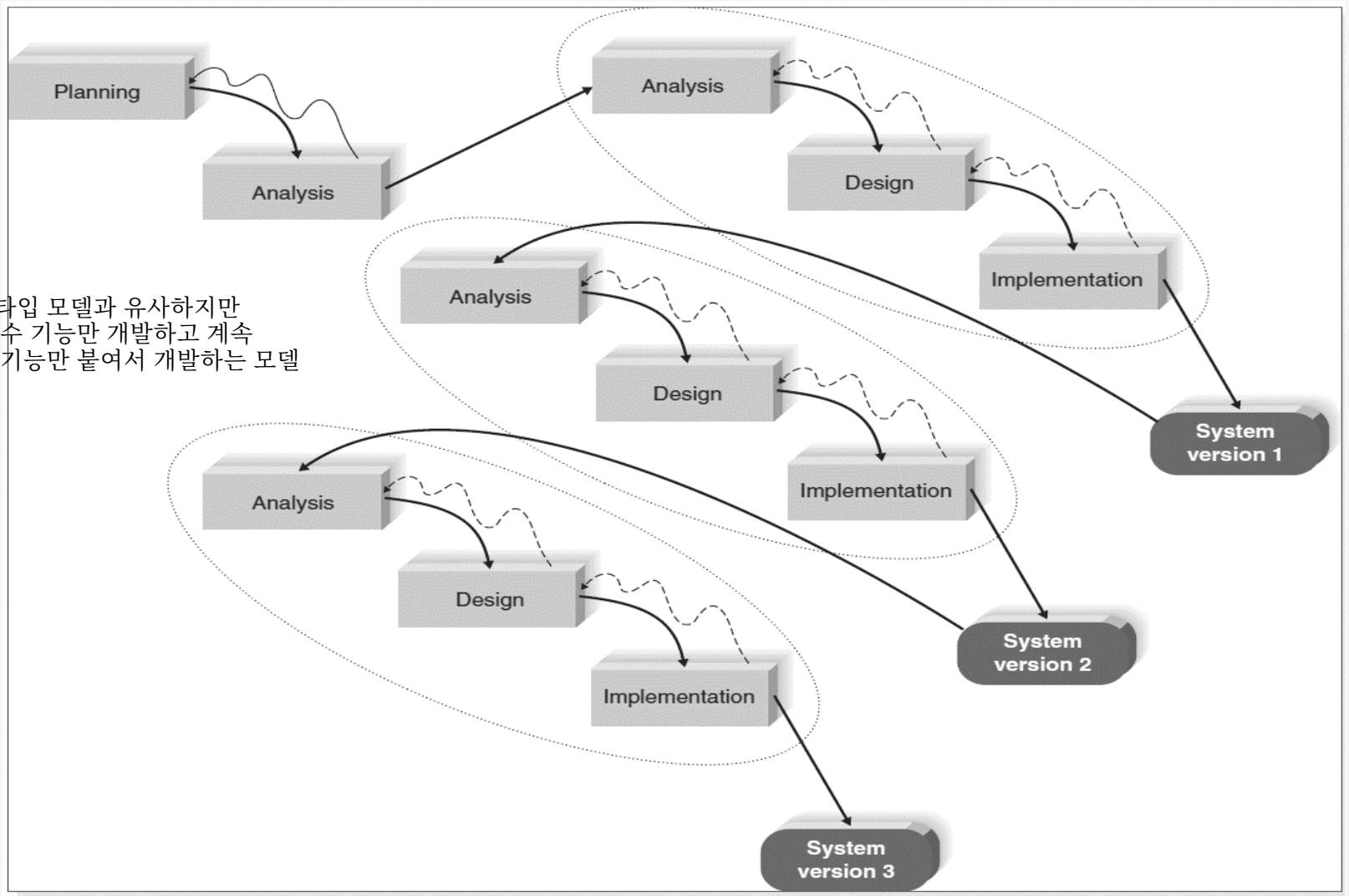
(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

Software Process Models (2)

- **Rapid Application Development**
  - **Critical elements**
    - CASE tools — Computer Added Software Engineering
    - JAD sessions — Joint Application Design
    - Fourth generation / visualization programming languages
    - Code generators
  - **Representative models**
    - Incremental development model(Phased development)
      : a series of versions
      단계적 개발
    - Prototyping
      : system prototyping
      사용자의 요구사항을 빠르게 받아들이기 위해서 임시로 만든 모델
    - Throwaway prototyping
      : design prototyping

: Phased Model (iterative development)

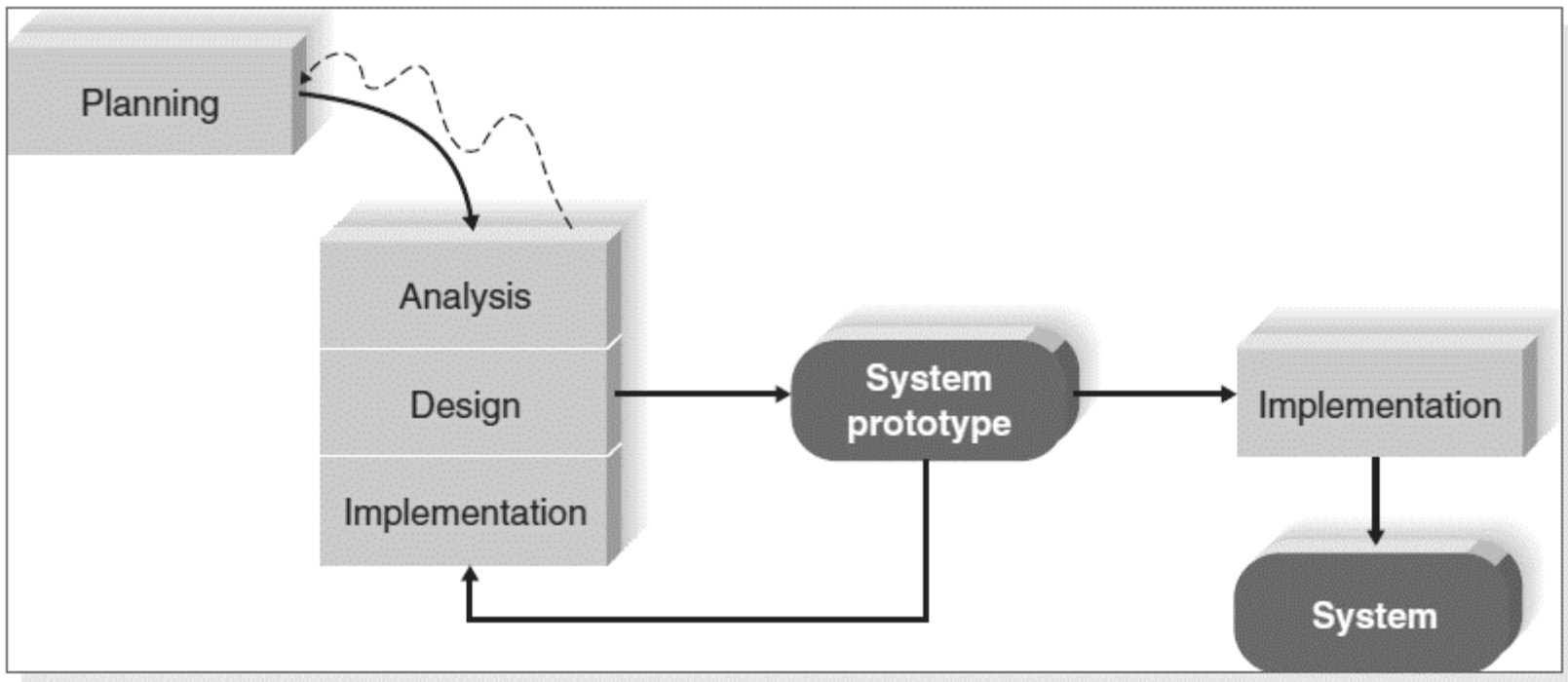
프로토타입 모델과 유사하지만
이건 필수 기능만 개발하고 계속
새로운 기능만 붙여서 개발하는 모델



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

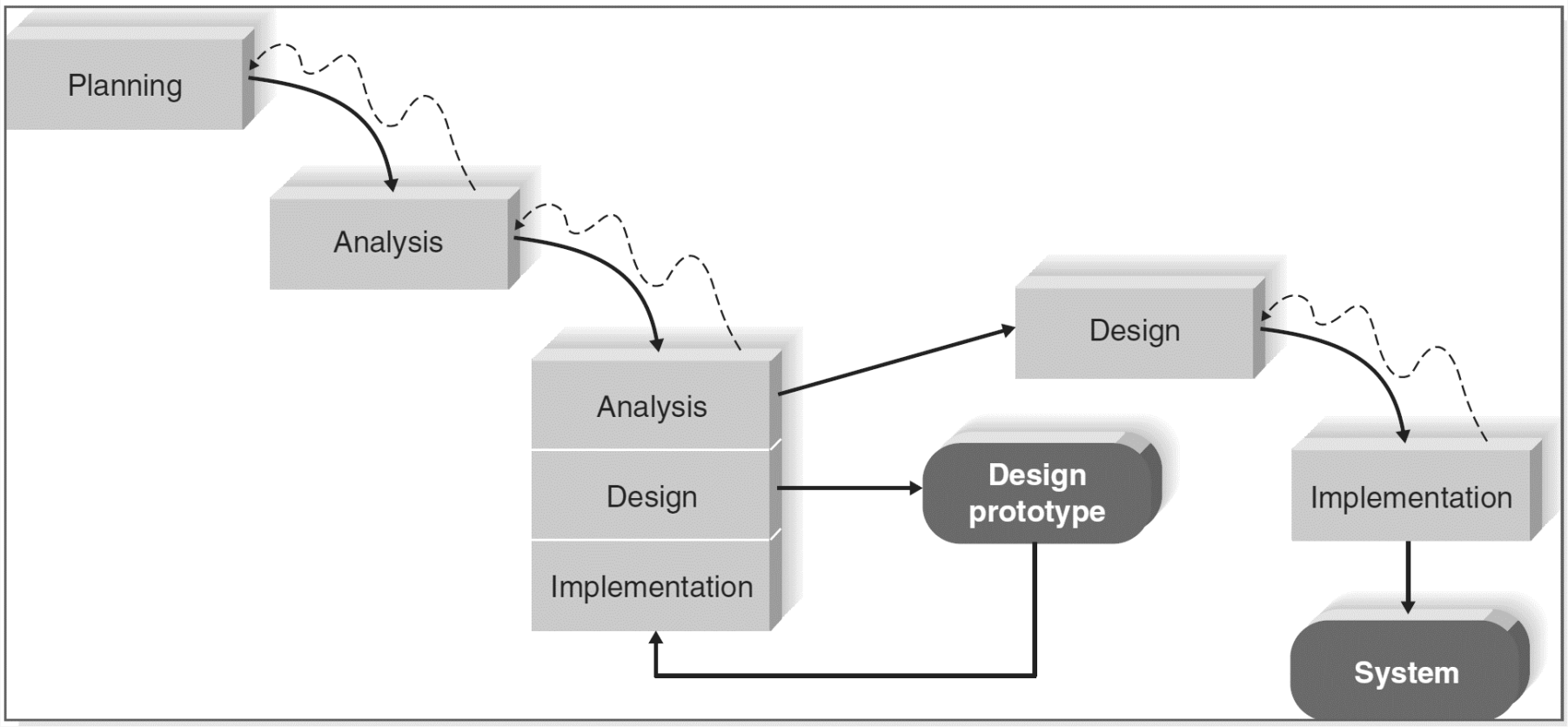
: Prototyping-based Model

대충 하나 만들어 놓고 분석, 설계, 구현을 반복하며 계속 구체화 시키면서 최종적으로 구현하는 모델



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

: Throwaway Prototyping Model



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

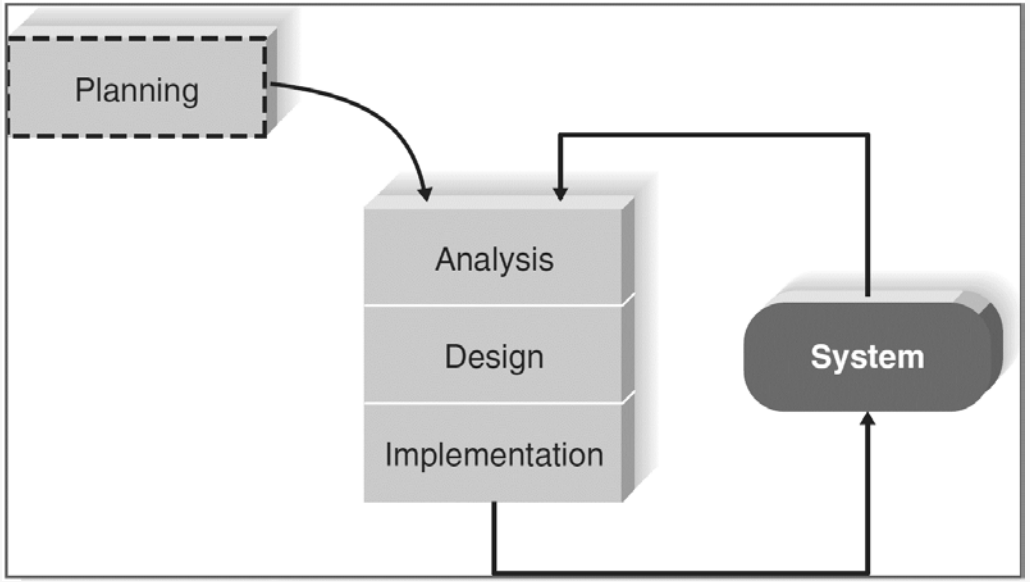
Software Process Models (3)

❖ **Agile Development**

- **Still popularizing today**
- **Simple, iterative development approach**
- **Representative models**
  - Extreme programming approach
  - Scrum approach

변화에 빠르게 적응하고, 유연하게 개발하는 방법

보통 2주에 한 번씩 분석, 설계, 구현을 시스템에 계속 시스템에 업데이트 하는 방식



(source: Dennis, Alan, Barbara Haley Wixom, and Roberta M. Roth. Systems analysis and design. John wiley & sons.)

Criteria for Selecting a Process Model – (1)

❖ **May be exist some variations**

Usefulness in Developing Systems	Waterfall	Parallel	V-Model
With unclear user requirements 사용자의 요구가 불분명한 경우	Poor	Poor	Poor
With unfamiliar technology 익숙하지 않은 기술을 사용하는 경우	Poor	Poor	Poor
That are complex 복잡한 시스템을 개발하는 경우	Good	Good	Good
That are reliable 신뢰성이 중요한 시스템을 개발하는 경우	Good	Good	Excellent
With short time schedule 개발 일정이 짧은 경우	Poor	Good	Poor
With schedule visibility 일정 가시성이 필요한 경우	Poor	Poor	Poor

Criteria for Selecting a Process Model – (2)

요구사항을 충분히 도출하고 사용하기 때문에
Excellent

Usefulness in Developing Systems	Iterative	System Prototyping	Throwaway Prototyping	Agile Development
With unclear user requirements	Good	Excellent	Excellent	Excellent
With unfamiliar technology	Good	Poor	Excellent	Poor
That are complex	Good	Poor	Excellent	Poor
That are reliable	Good	Poor	Excellent	Good
With short time schedule	Excellent	Excellent	Poor	Excellent
With schedule visibility	Excellent	Excellent	Poor	Good

Software Development Methodology

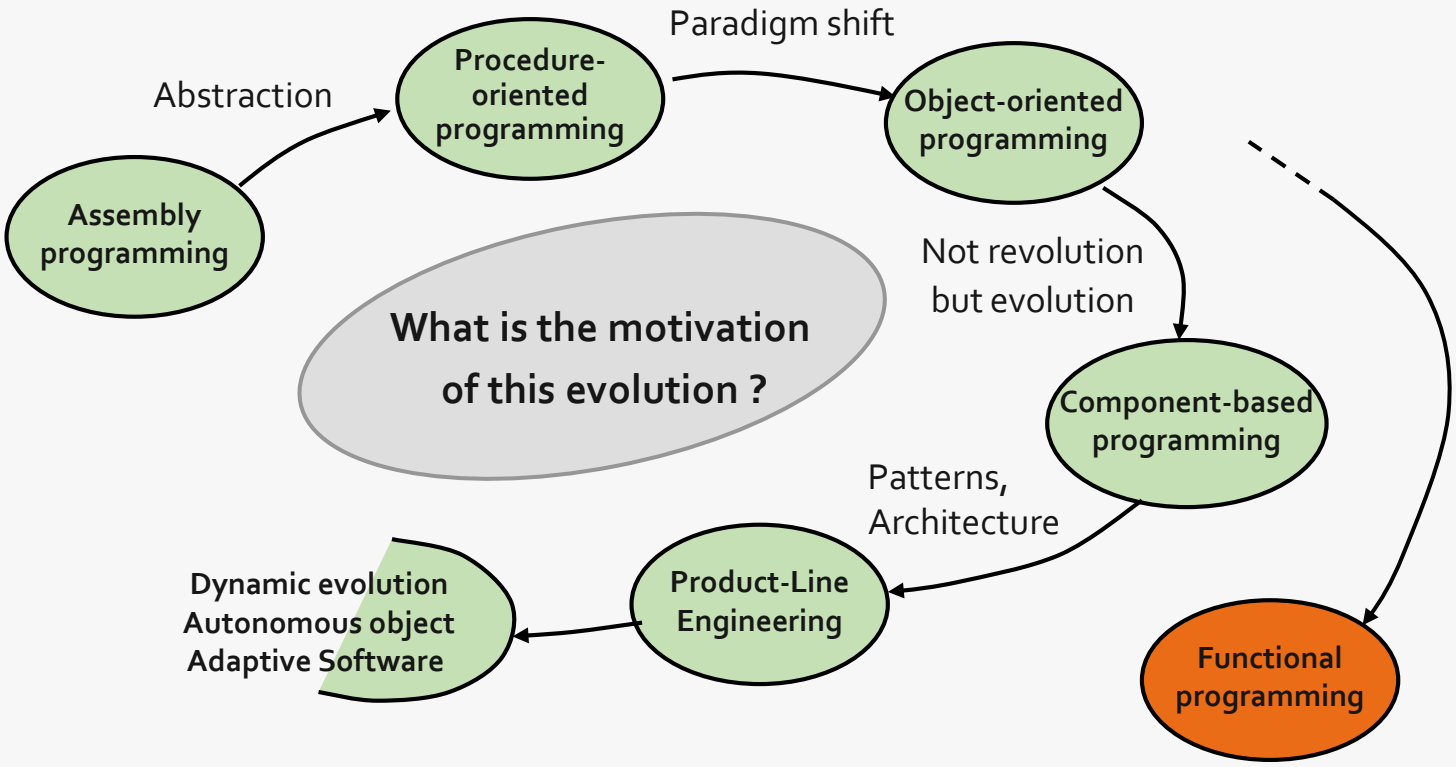
- **Methodology**
  - Formalized techniques to implementing the SDLC/Process models
  - Describes "How to develop"

- **Representative Methodology**
  - Structured Analysis and Structured Design(SASD)
  - Information Engineering
  - Object-Oriented Analysis and Design
  - Component-Based Software Development
  - Product-Line Engineering
  - In future ?

Software Development Methodologies

- **Structured Analysis and Structured Design (SASD)**
  - Process-centered; Data flow diagram, Structure chart, etc
- **Information Engineering**
  - Data-centered; Entity-relationship diagram, Process decomposition diagram, CRUD matrix, etc
- **Object-Oriented Analysis and Design**
  - Object-oriented; Class diagram, Sequence diagram, etc
- **Component-Based Software Development**
  - Object-oriented; Component diagram, Reusable components, etc
- **Product-Line Engineering**
  - Object-oriented; Software architecture, Reusable components, etc

Evolution of Software



Summary and Discussion

- Software Development Process vs Methodology
- What is the deliverables in software development phases ?