Université de Paris

Génie Logiciel
Elements of a software project

Sylvain Lobry

23/09/2022

Elements of a software project

What is a software project?

Definition:

A software project is the complete procedure and activities to achieve an intended software product.

Elements of a software project

What is a software project?

Definition:

A software project is the complete procedure and activities to achieve an intended software product.

- Functional objectives
- Technical specifications

Elements of a software project

What is a software project?

Definition:

A software project is the complete procedure and activities to achieve an intended software product.

- Functional objectives
- Technical specifications
- Definition of the scope
- Planning
- Development
- Risk analysis
- Management
- Monitoring

Elements of a software project

Some people of a software project

- Maître d’ouvrage (= project owner or client): stakeholder that benefits from the project’s results
  - Identifies the needs
  - Defines the goals
  - Finances the project
  - Oversees the project’s planning and realization
  - Take general decisions if needed
- Maître d’oeuvre (= contractor): proposes and implements a solution to realize the project

Elements of a software project

Definition of the scope

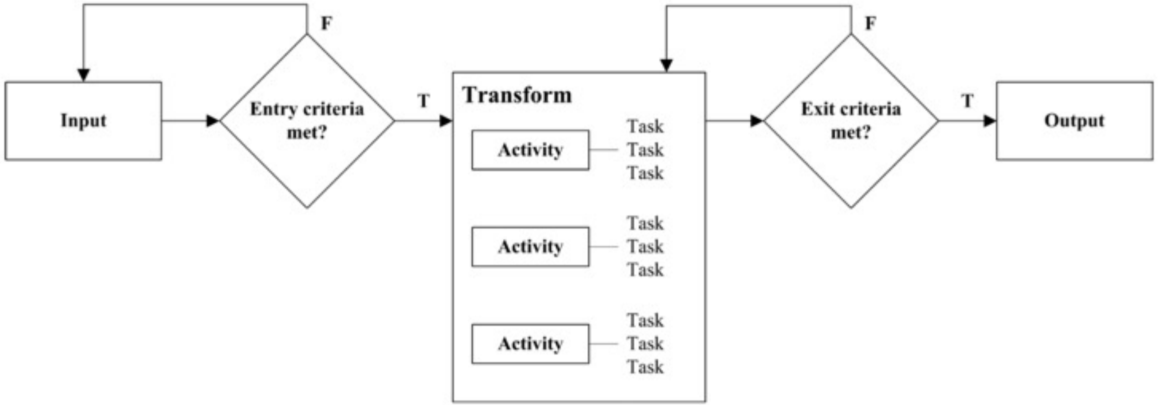
- Scope must be a balance of:
  - Time
  - Cost
  - Quality

Elements of a software project

Software processes

Defintion:

a software process is a set of interrelated activities and tasks that transform input work products into output work products, SWEBOK v3



Elements of a software project

Main software activities

- List of activities to be conducted during a software project
- Each can be seen as a process

Elements of a software project

Main software activities

- Objectives definition
- Requirement analysis
- Feasibility analysis
- Requirements specifications
- Design
- Implementation
- Unit testing
- Integration
- Validation
- Deployment
- Maintenance

Elements of a software project

Main software activities

- Objectives defintion: understanding what will be the usage (in its context) of the software

Elements of a software project

Main software activities

- Requirement analysis: determining the needs of the stakeholders

Elements of a software project

Main software activities

- Feasability analysis: determining which outcomes can be achieved in the specific context of the project

Elements of a software project

Main software activities

- Requirements specifications: formalization of the requirements in the form of a document that can be systematically reviewed, evaluated and approved

Elements of a software project

Main software activities

- Design: precise definition of the components of the software based on the requirements

Elements of a software project

Main software activities

- Implementation: building-up the program following the design and instructions.

Elements of a software project

Main software activities

- Unit testing: verifying individually that each component of your software answer its specification.

Elements of a software project

Main software activities

- Integration: connection of the different sub components of the program.

Elements of a software project

Main software activities

- Validation: validation that the software, as a whole, is answering the initial objectives and expectations from the customer.

- Not to be confused with verification!
- Verification = analysis (often without executing code) during development period to check whether a specific requirement is met

Elements of a software project

Main software activities

- Deployment: activities to make the software available for use.

Elements of a software project

Main software activities

- Maintenance: to modify the application after its deployment to fix bugs, improve performance or improve functionalities

Elements of a software project

Main software activities

- Objectives definition
- Requirement analysis
- Feasibility analysis
- Requirements specifications
- Design
- Implementation
- Unit testing
- Integration
- Validation
- Deployment
- Maintenance

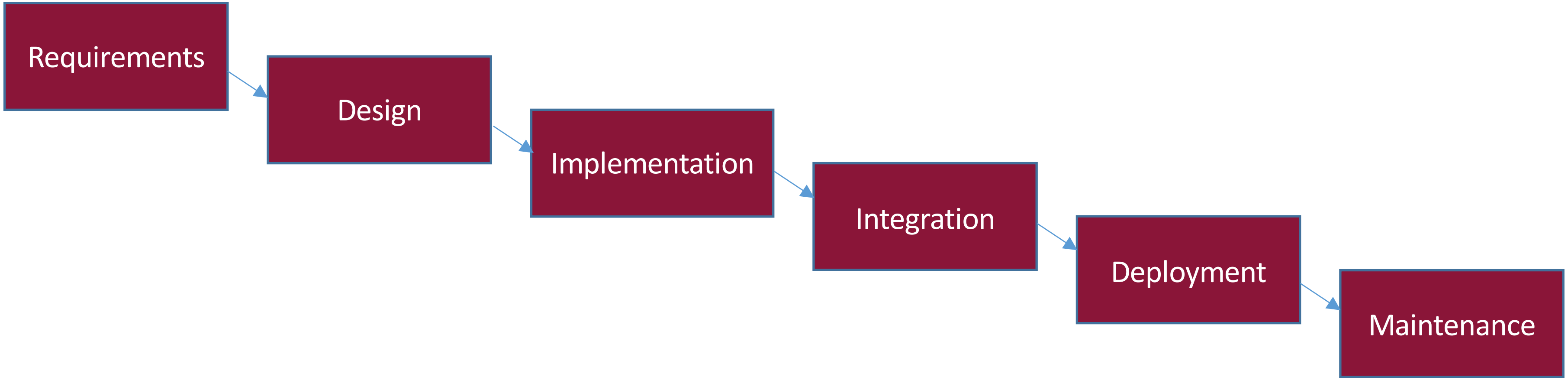
Software Development Life Cycle

- A software Development Life Cycle (SDLC) puts the different processes in order
- Also know as Software process model
- Chosen at the start of the project
- Brings discipline to software development
- 4 SDLC models today

Software Development Life Cycle

Waterfall model

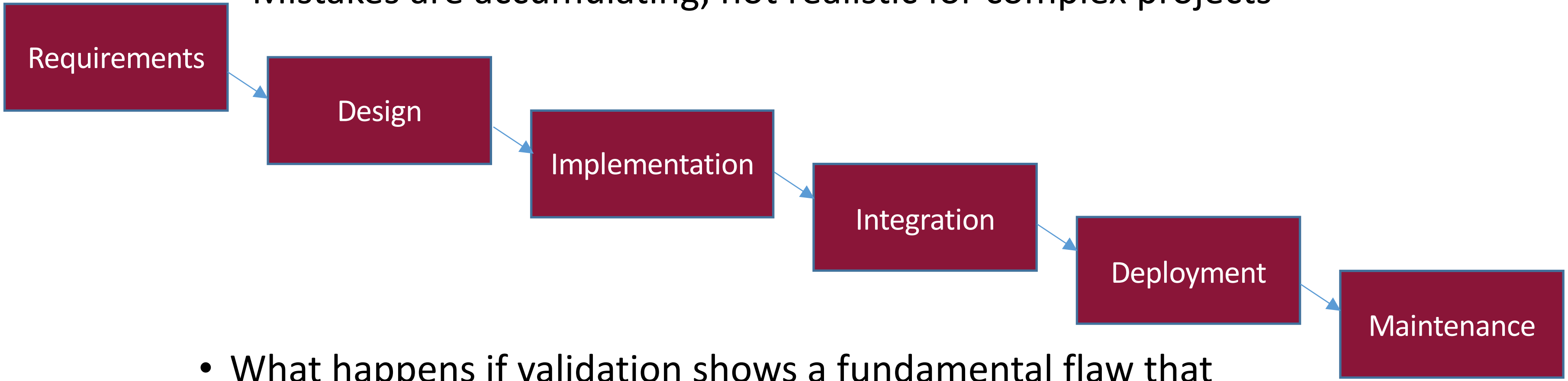
- Simplest SDLC model
- Proposed by Royce in 1970



Software Development Life Cycle

Waterfall model

- + Easy to plan and to follow
- +- Requirements cannot change
- - Mistakes are accumulating, not realistic for complex projects



Requirements

Design

Implementation

Integration

Deployment

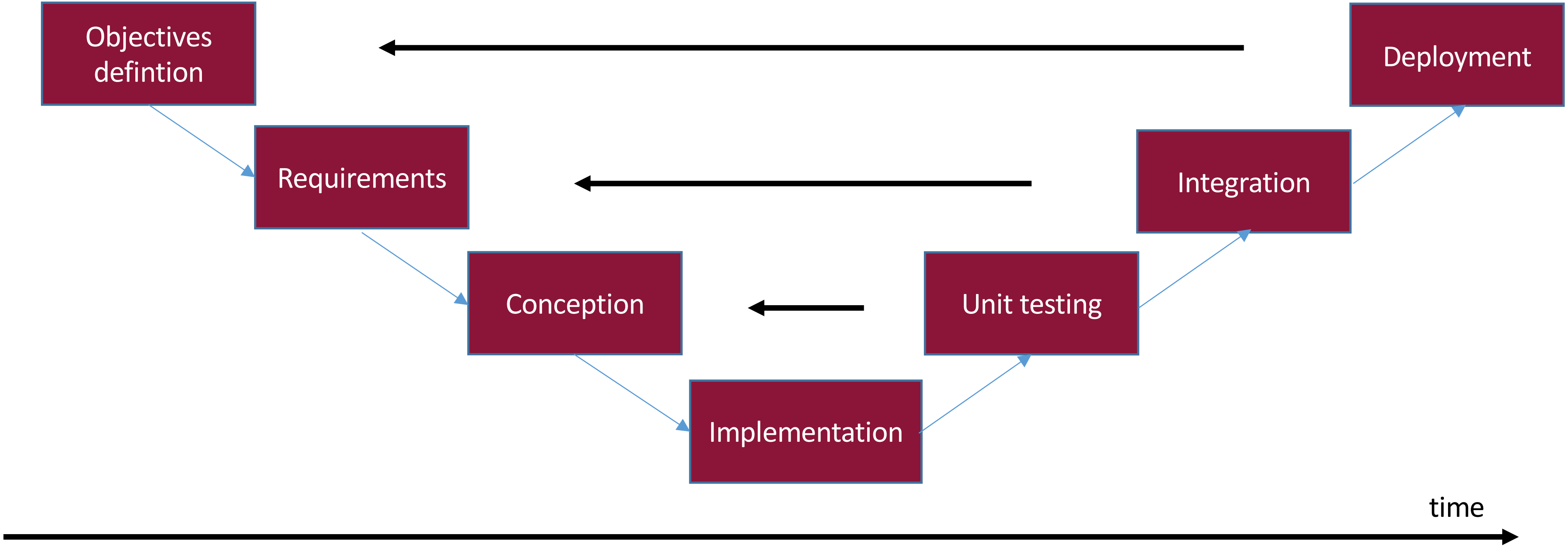
Maintenance

- What happens if validation shows a fundamental flaw that requires a design change ?

Software Development Life Cycle

V-model

- Extension of waterfall



Software Development Life Cycle

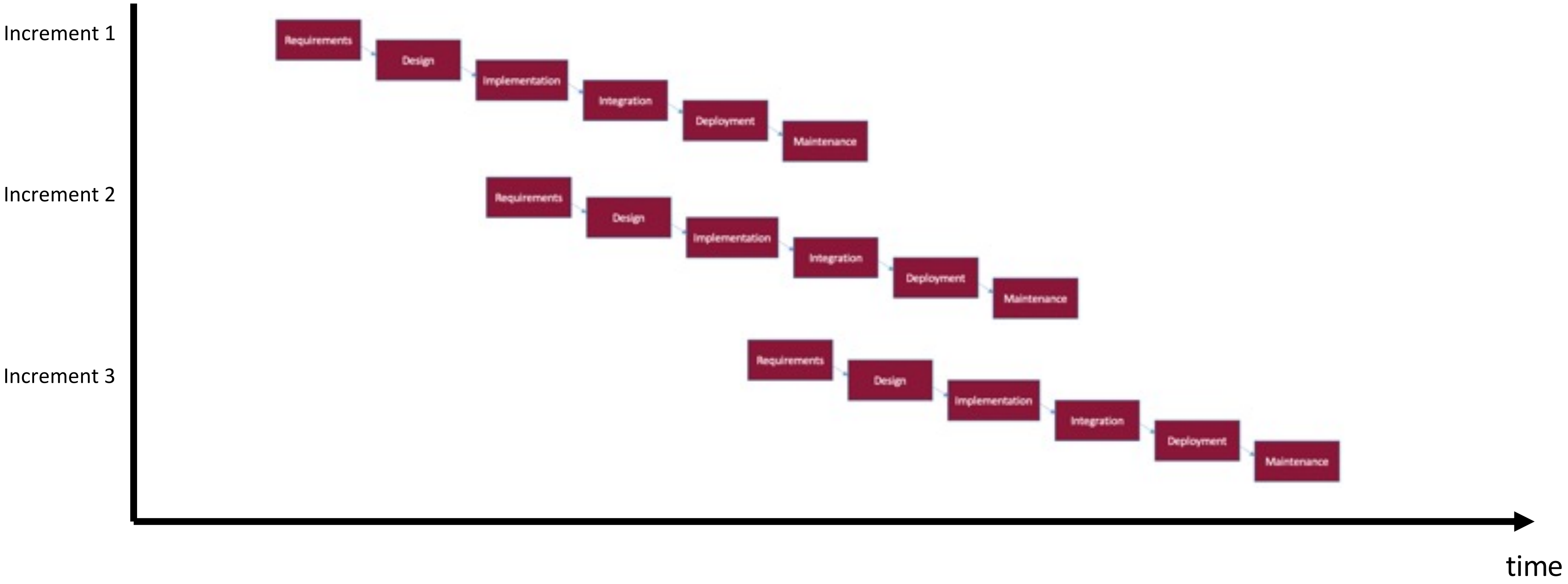
V-model

- Extension of waterfall
- Still simple
- With each components, verification (downward phase) or validation (upward) procedures are defined
- Still not flexible enough for complex projects

- In general, what you have been doing

Software Development Life Cycle

Incremental model



Software Development Life Cycle

Incremental model

- Software broken down in sub-components
- First increment: core functionality
- Subsequent increments take into account feedback

- + Aligns better with the customer needs
- + Fast delivery
- - Fundamental flaws can exist

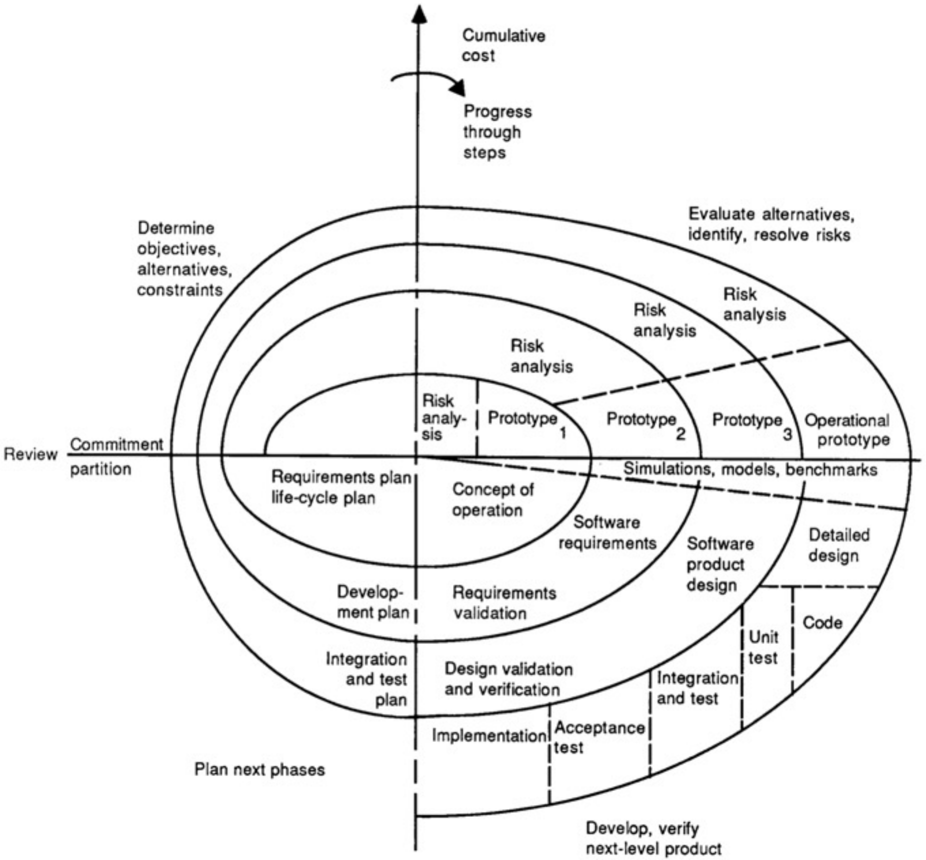
Software Development Life Cycle

Spiral model

- Based on 4 quadrants
- Can be seen as a generalization of previous models
- Risk driven model

- + Suitable for complex projects
- - Requires experience, costly

Boehm, 1988



Software Development Life Cycle

SDLC models

- 4 models
- Different levels of complexity
- Relatively rigid
- Solution since 2000s : Agile (coming up later in this class)