Software Process Models I

By
Suradet Jitprapaikulsarn

What is Software Process Model?

- A descriptive representation of software process
- Define guideline for software development

Software Process Models

- Linear Sequential
  - Waterfall
- Incremental
  - Incremental
  - Rapid Application Development (RAD)
- Evolutionary
  - Prototyping
  - Spiral
- Component-Based Software Engineering (CBSE)
- Computer-Aided Software Engineering (CASE)
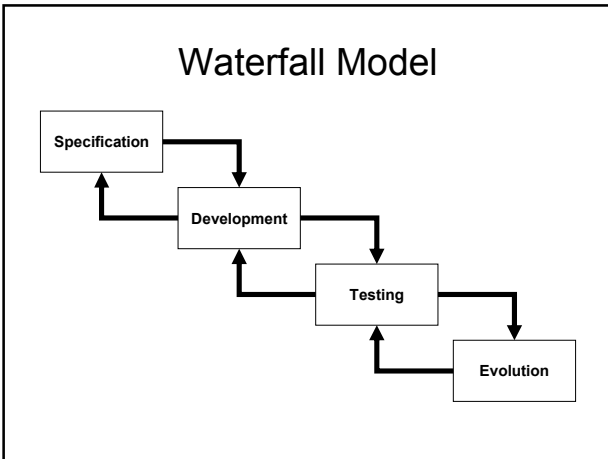- Rational Unified Process
- Other models

Waterfall Model

Waterfall Model

The Good
- Similar to process model used in other engineering disciplines
- Specify work products of each phase

The Bad
- Inflexible to adapt to changing requirements
- Early commitment
- A working program is not available until late in the project

---

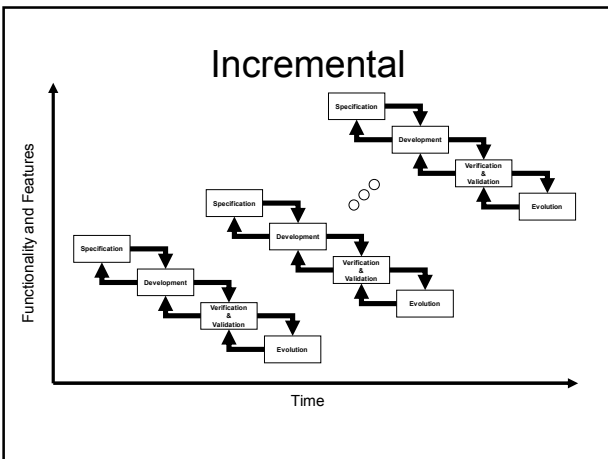# Incremental

Incremental

The Good

- Increments can be planned to manage risks
- Once an increment is complete, it can be put into service

The Bad

- System needed to be properly modularized
- The process usually conflicts with the normal procurement model

Rapid Application Development (RAD)

- Emphasize a short development cycle (1-3 months)
- Time frame is decided beforehand and generally unchangeable
- High-speed incremental model
- Using components in development

Rapid Application Development (RAD)

The Good

- Fast to produce work product
- High Reusability

The Bad

- System needed to be properly modularized
- Performance and scalability could be problematic
- Not appropriate when using with new technology

Prototyping

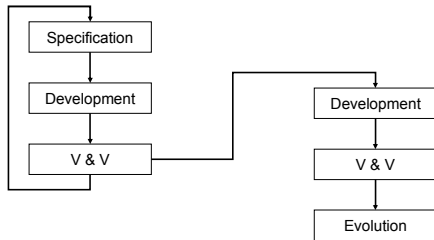```
┌──────────────┐
│ Specification│
└──────┬───────┘
       ↓
┌──────────────┐   ┌──────────────┐
│ Development │   │ Development │
└──────┬───────┘   └──────┬───────┘
       ↓   ↓
┌──────────────┐   ┌──────────────┐
│   V & V   │───────▶│   V & V   │
└──────────────┘   └──────┬───────┘
                               ↓
                        ┌──────────────┐
                        │ Evolution   │
                        └──────────────┘
```

# Two types of prototyping

- Throwaway
- Evolutionary

# Prototyping

**The Good**
- Requirements are better accommodated
- Early problem detection
- Incremental specification

**The Bad**
- Users may think that the product is complete
- The process maybe unclear
- Poorly structured system

4