Software Development Processes

Lecture Objectives:

1) Explain the difference between general software and real-time software
2) List the three tracks of software engineering lifecycles
3) Define the term software process and software methodology
4) Compare and contrast the aspects of the Waterfall model, the V model, spiral models, and other software development models.
5) Explain the ROPES Software Development Process (Rapid Object Oriented Process for Embedded Systems)
6) Explain the relationship between Systems Engineering and Software Engineering
7) Define the concept of an Architecture

- Complete the homework sheet

Homework

What is Real-Time System

Time critical / Time sensitive System.

↳ Missile System.

What is a Real-Time System

- A real time system is a system in which
  - correctness of the system depends not only on the logical results, but also on the **time** in which the results are produced.
  - works in a reactive and time-constrained environment
- Examples
  - MP3 decoding, digital TV receivers, radar systems, space mission controls, car ABS system

What is an Embedded System

Small computer built into another device

Maybe

- True

- Robot Arm
- Microwave
- Uses microcontroller

What is an Embedded System

- a combination of hardware & software (a "computational engine") to perform a **specific** function
- is part of a larger system, say a real-time system, that may not be a "computer"
- works in a reactive and time-constrained environment
- Examples:
  - Microwave, ~~VCR~~, dishwasher, mobile phone, car ABS system, seat and mirror positioning system, etc.

Embedded Rules!

- 2009 -> Estimated 15.4 billion in microcontroller sales
  - Microchip shipped 5 billionth PIC micro in 2006
- Embedded processors account for ≈100% of worldwide microprocessor production!
- "Of the 11 billion CPUs sold in 1997, only a few tens of millions went into PCs and workstations, representing approximately zero percent of the market."-Jack Ganssle
- 1999: #embedded processors in the home
  - estimated at 40-50 and growing

Computer engineers do embedded!!!!

How do you write a program?

Watch TV until bed and

hack out code.

Play WoW

Bad Coding Things.

How do we solve problems?

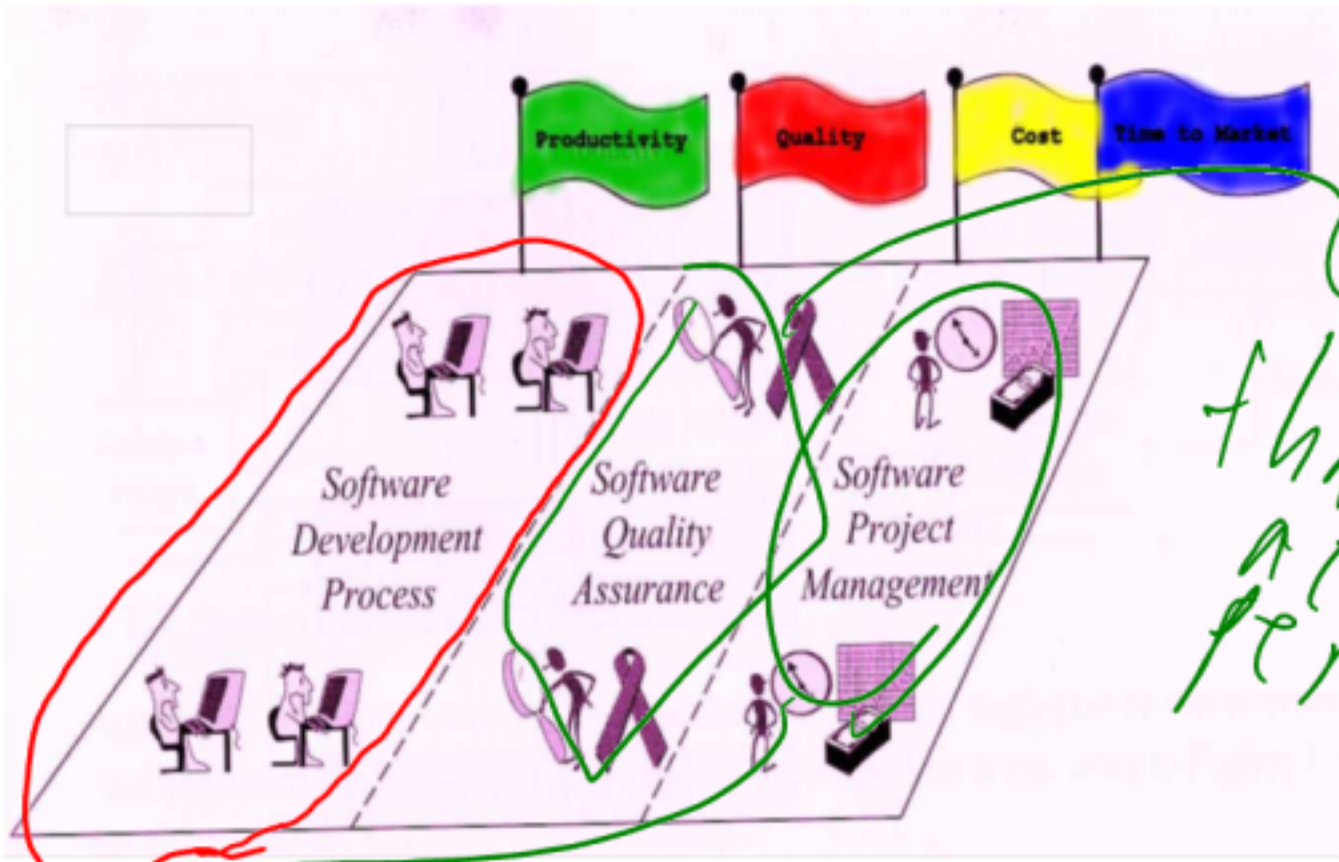
Polya's Principles

1. Understand the problem.
2. Develop a plan of attack.
   ⇒ How are we going to Solve the problem.
3. Implement the plan.
4. ~~Verify~~ Reflect on the plan.

How does this tie into software engineering

- Software Engineering is merely problem solving!
  - Problem is proposed by a customer
  - Software Engineers come up with an understanding of the problem and plan of attack.
  - Software Engineers supervise developers implementing the solution
  - After problem is solved, an analysis is conducted on the developed solution.

Software Life-Cycle Activities



Ensuring that DGYs activities are performed properly.

- Transforms the initial concepts into an operational system

- Overseeing the control of the project.

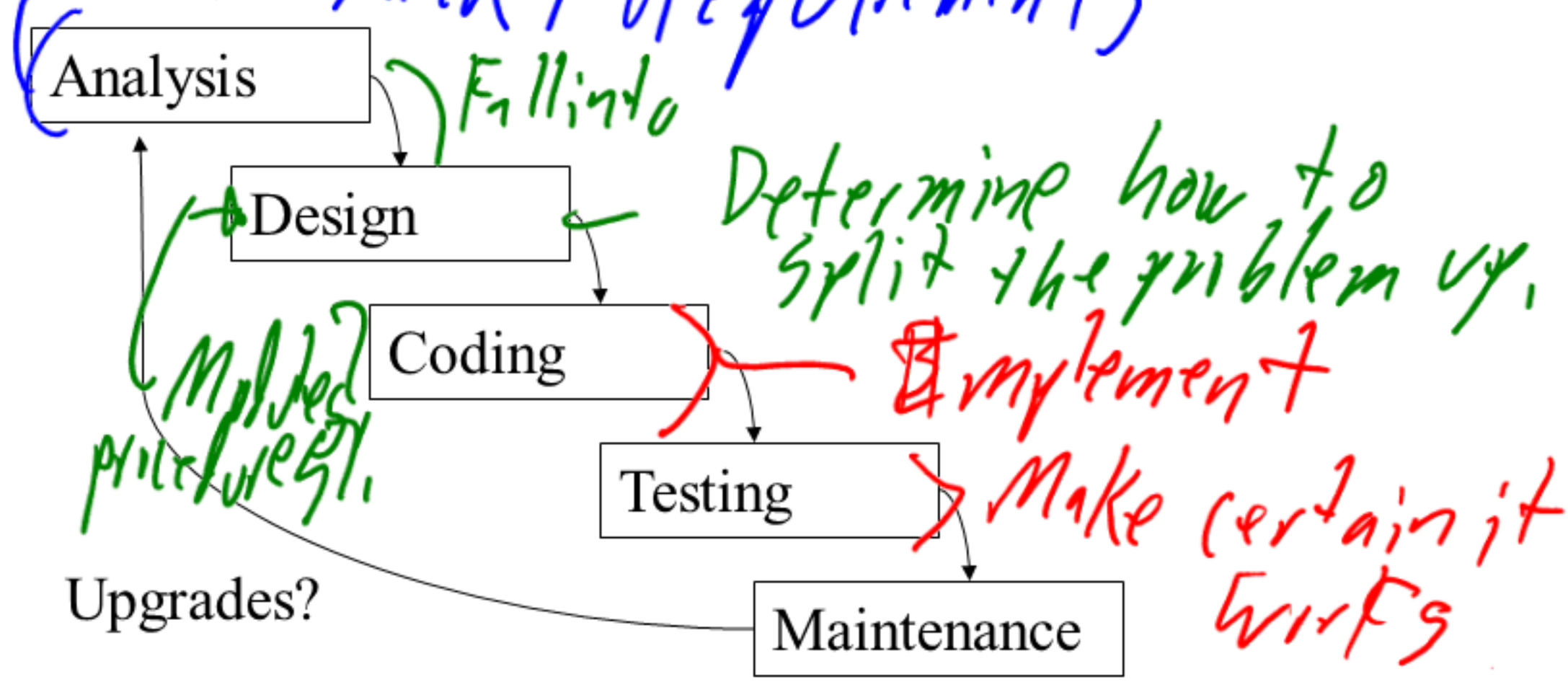
Software Process Definition

- A *software process* is a series of phases of activities performed to construct a software system. Each phase produces some artifacts which are the inputs to other phases. Each phase has a set of entrance criteria and a set of exit criteria.
- A *software methodology* defines the steps or how to carry out the activities of a software process.

- Traditional approach

Proposed by Royce in 1970

Waterfall Model



Understand / Requirements

Finding

Determine how to split the problem up.

Implement

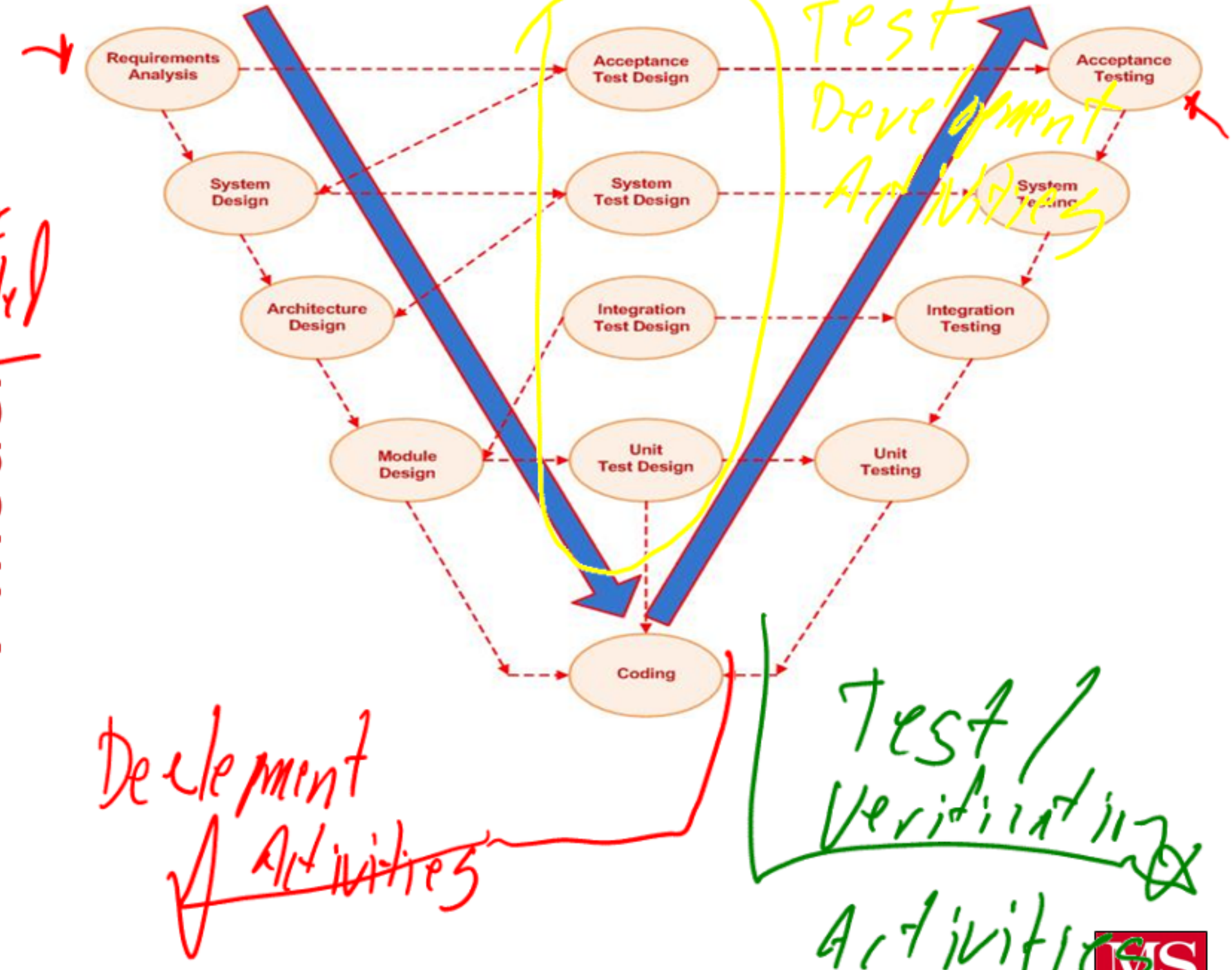
Make certain it works

Make pictures!

Not very practical.

MSOE

V Model


Requirements Analysis
Acceptance Test Design
Acceptance Testing
System Design
System Test Design
System Testing
Architecture Design
Integration Test Design
Integration Testing
Module Design
Unit Test Design
Unit Testing
Coding

to

Phases

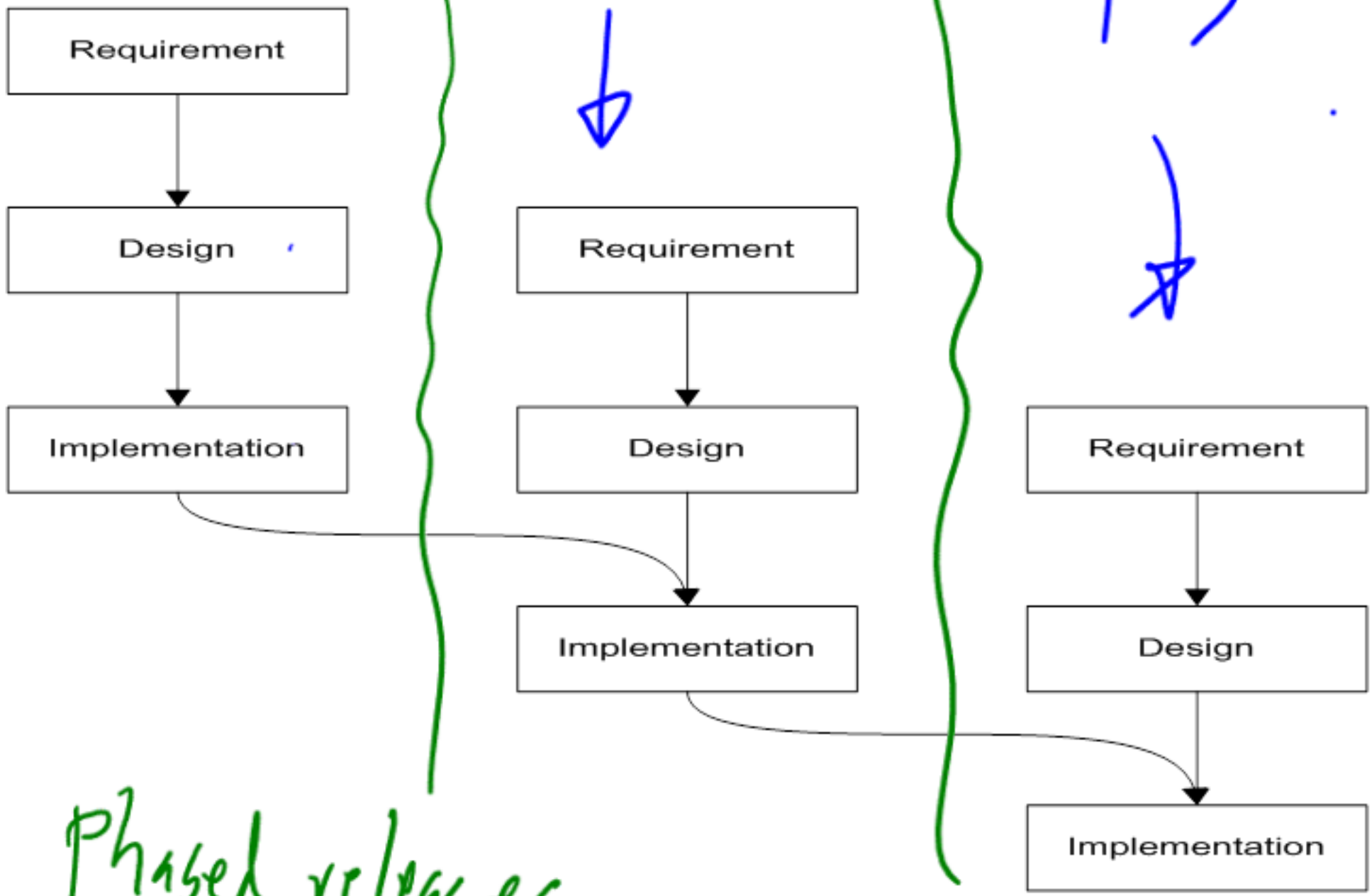
Parallel Development

P1

P2

P3

Incremental Model



Phased releases

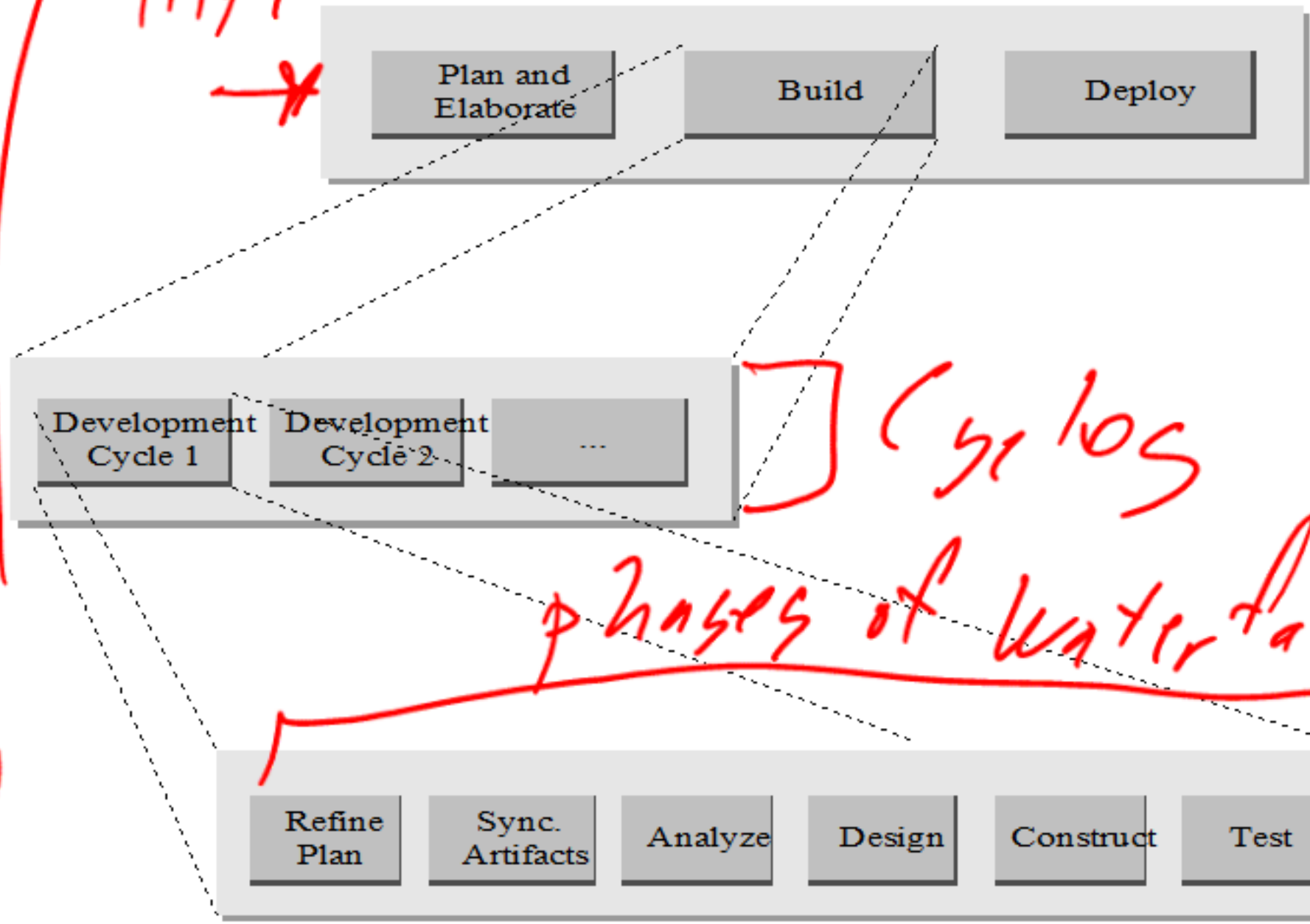
t n

The Spiral Process

Development in pieces

High level

Iterative Development



Cycles

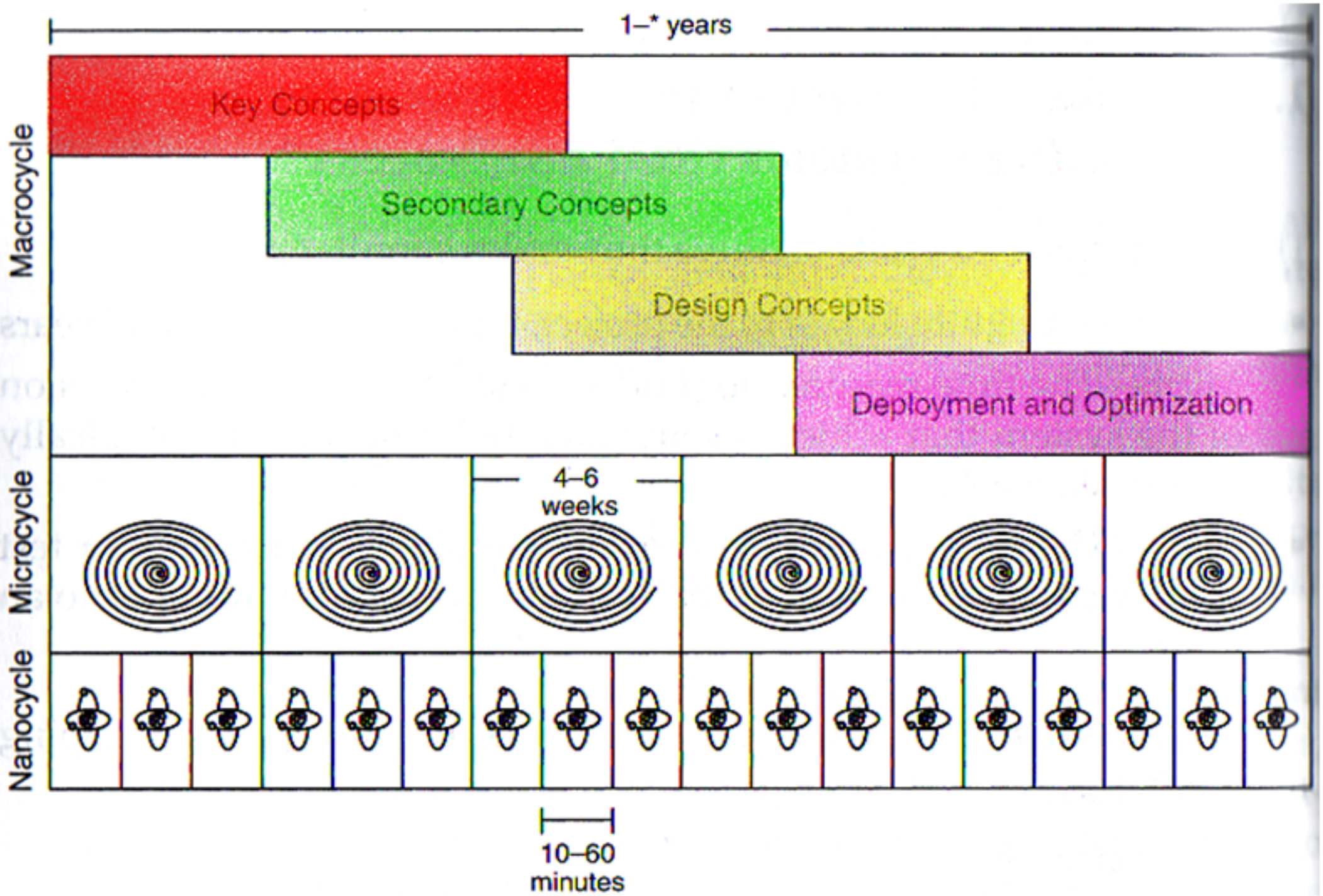
phases of waterfall

The ROPES Process

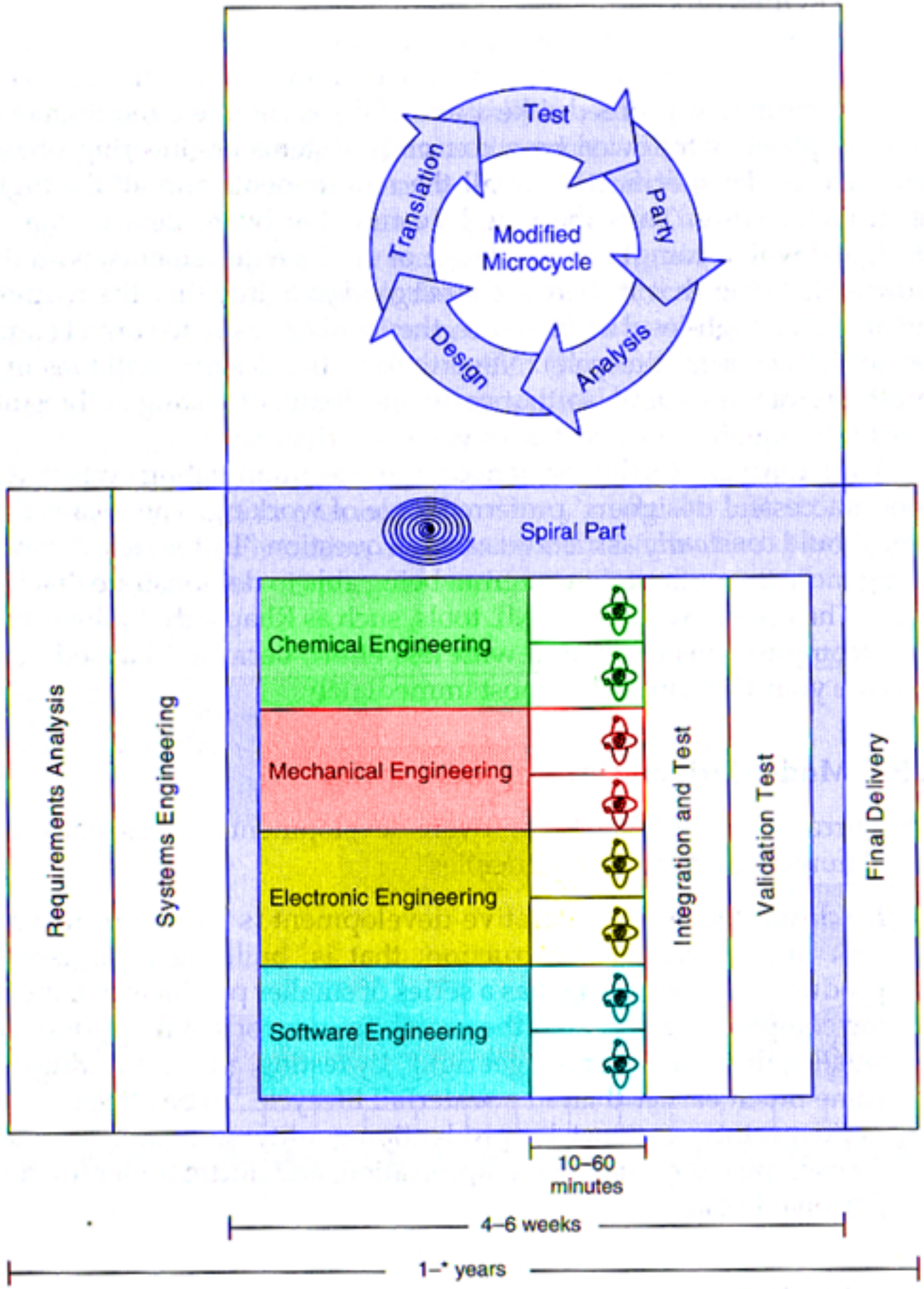
- Operates on three time scales simultaneously
  - Macro
  - Micro
  - Nano

Next

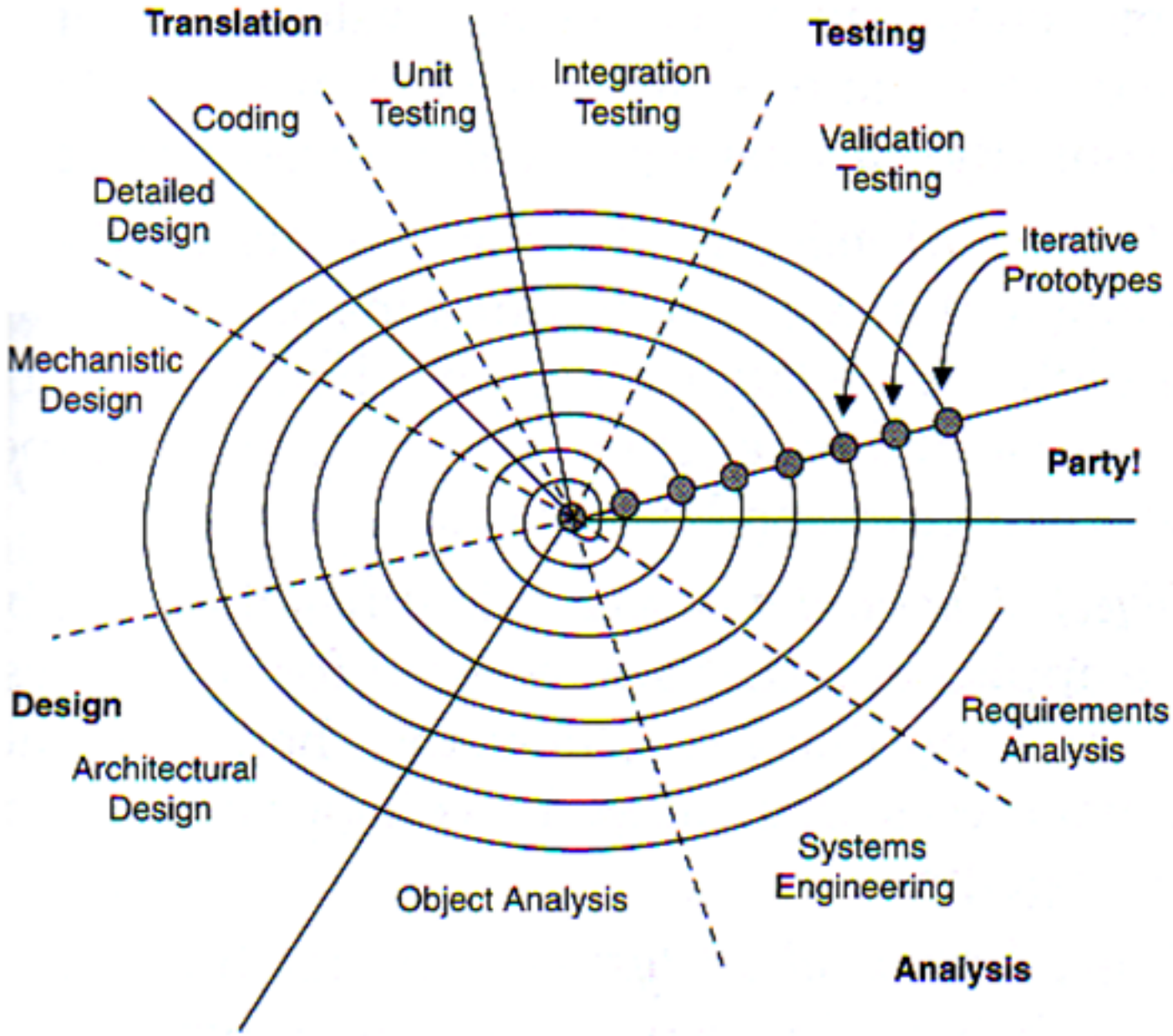
Ropes Spiral Macrocycle



Ropes Semi-Spiral Lifecycle



Ropes Spiral



SEI CMMI Model

- Capability Maturity Model - Integration
  - Software Engineering Institute
  - Carnegie Mellon University

0 – Incomplete (chaotic, unpredictable)
1 – Performed (weak process at best)
2 – Managed (disciplined process)
3 – Defined (standard, consistent)
4 – Quantitatively Managed (predictable)
5 – Optimizing (continuous improvement)

- Most organizations at level 0 or 1

- Many now reaching levels 3-5
  - Motorola (most groups at level 3 or above)
  - Aerospace Firms (many at levels 4 & 5)
    - Winning contracts because their bids mean something

Quo

Status

- Mature process is reliable and high quality

Systems Engineering Versus Software Engineering

- Systems Engineering
  - The definition, specification, and high level architecture of a system that is to be realized with multiple disciplines
  - Typically includes electrical, mechanical, software, chemical, and other engineering fields
- Software Engineering
  - The definition, specification, and high level architecture of a software system to be realized with multiple components
  - Typically includes multiple pieces to be solved

Architecture

- A set of strategic decisions that effect the structure, behavior, or functionality of a system