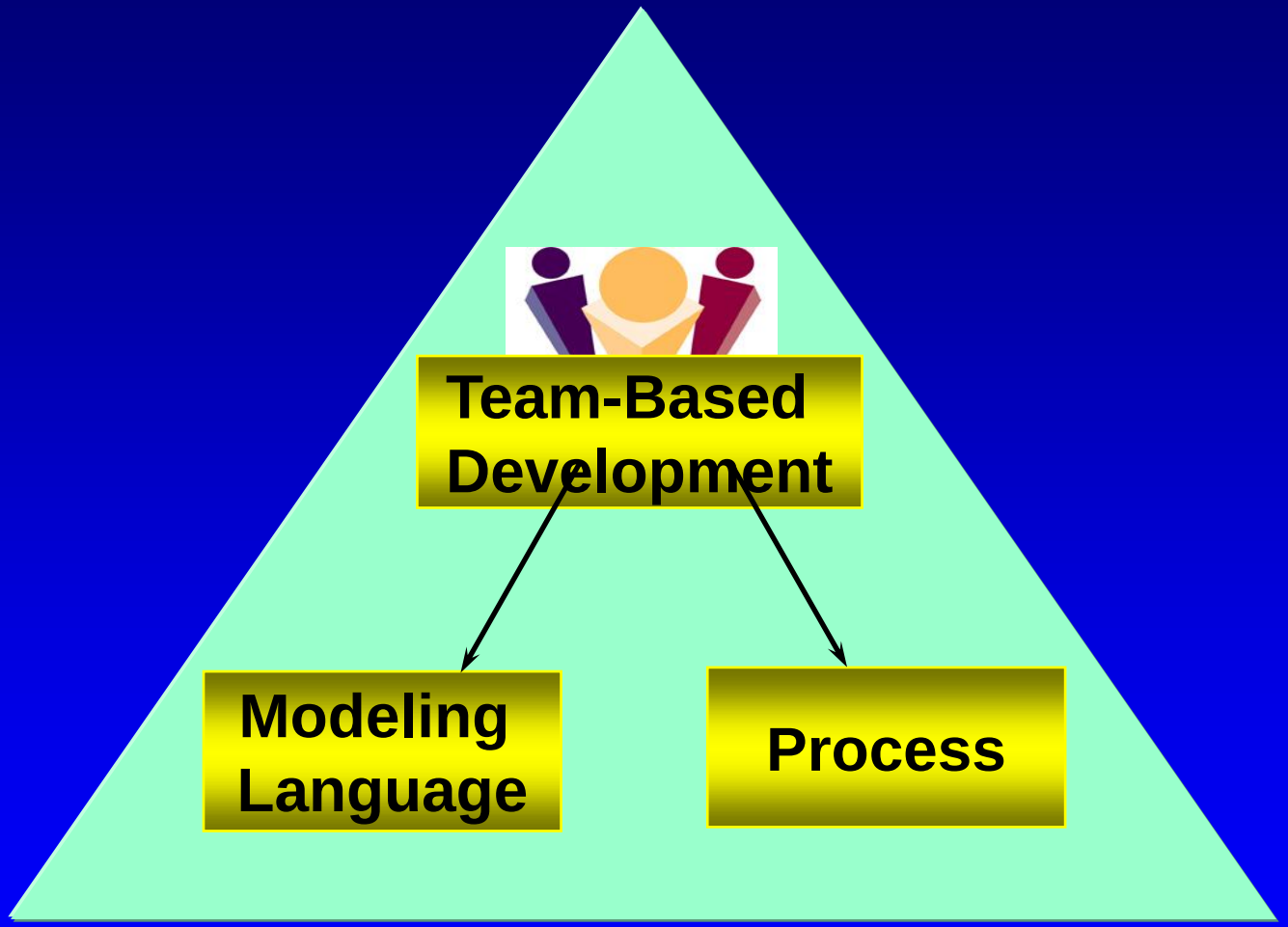
Introduction to Rational Unified Process

R

Objectives: Rational Unified Process

- Describe the **Unified Modeling Language (UML)**
- Define what a **software development process** is
- Describe the **Rational Unified Process**
- Explain the four **phases** of the Rational Unified Process and their associated milestones
- Define **iterations** and their relation to phases
- Explain the relations between:
  - Models and workflows
  - Phases, iterations, and workflows
- Define **artifact**, **worker**, and **activity**
- State the importance of automated tool support

Building a System - A Language Is Not Enough



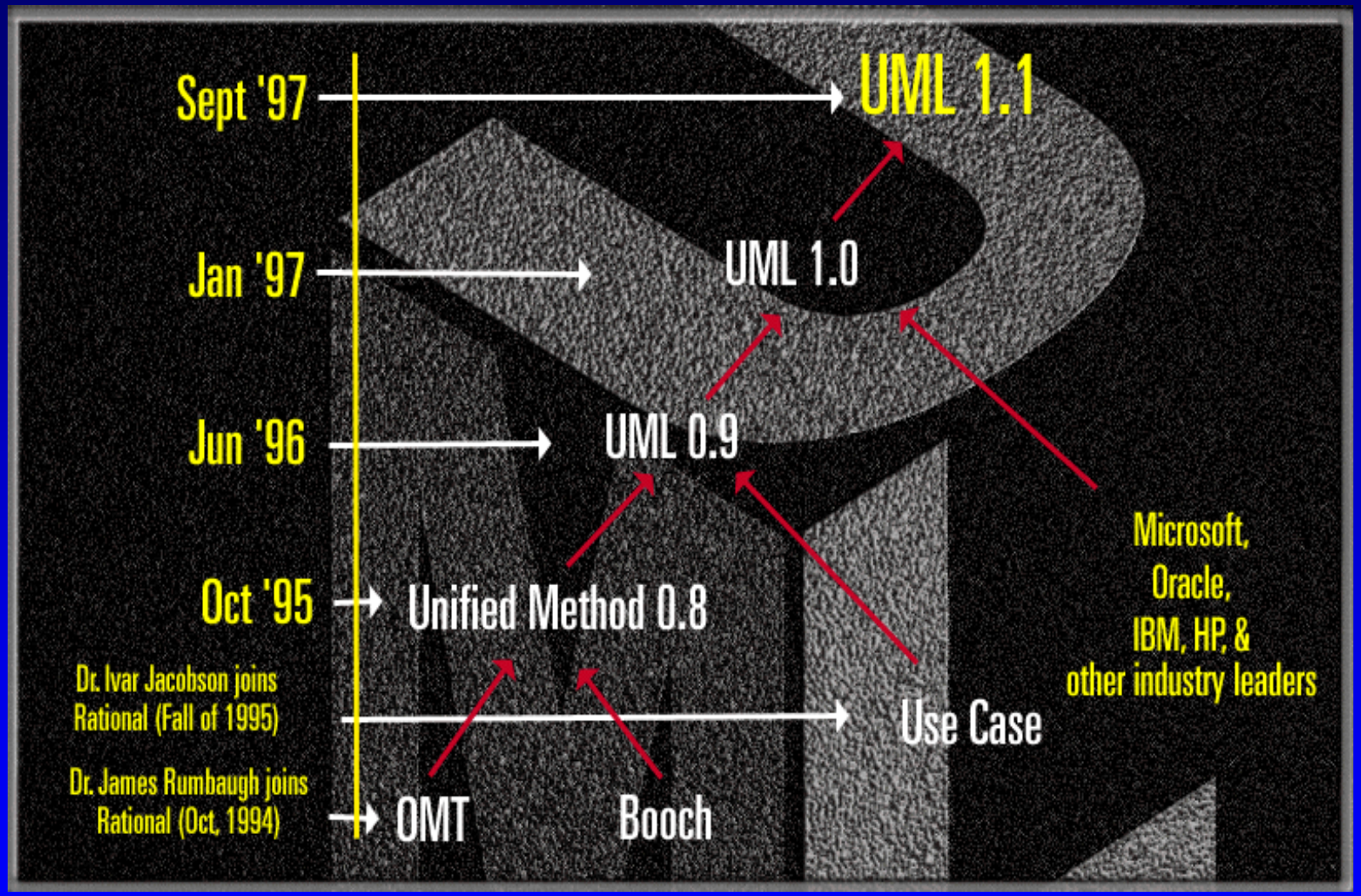
What Is the UML?

- The Unified Modeling Language (UML) is a language for
  - Specifying
  - Visualizing
  - Constructing
  - Documenting

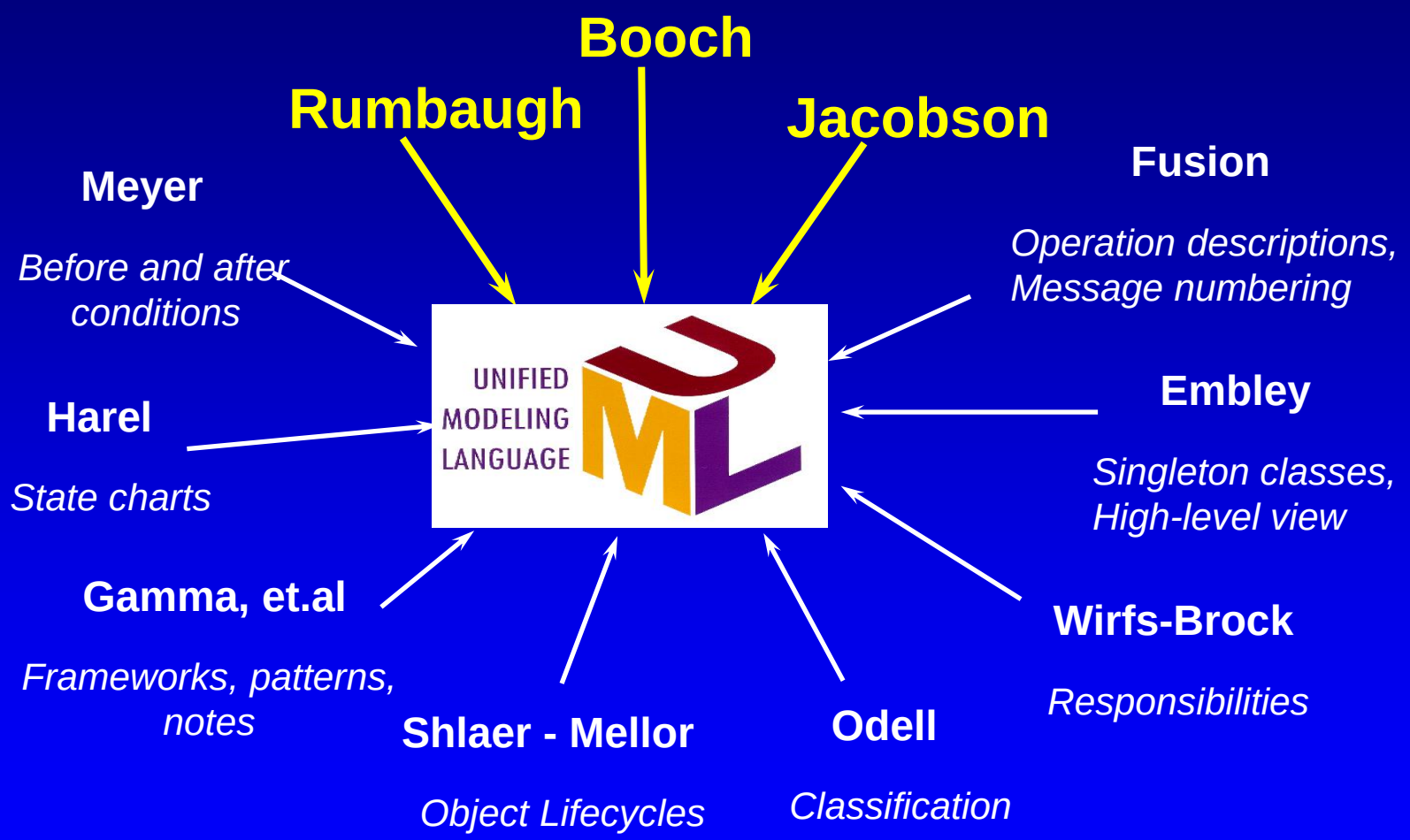
  the artifacts of a software-intensive system

R

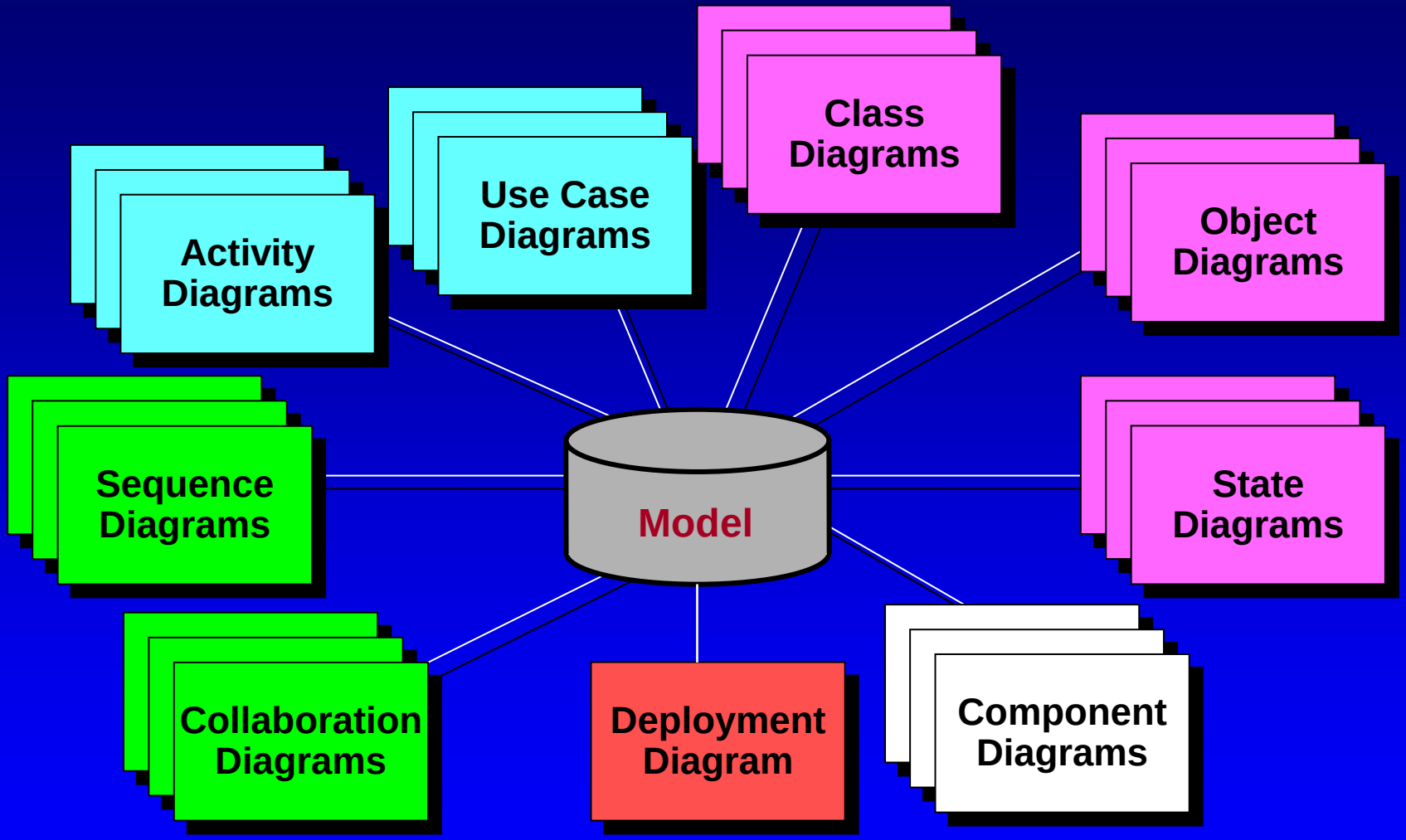
UML History



Inputs to UML

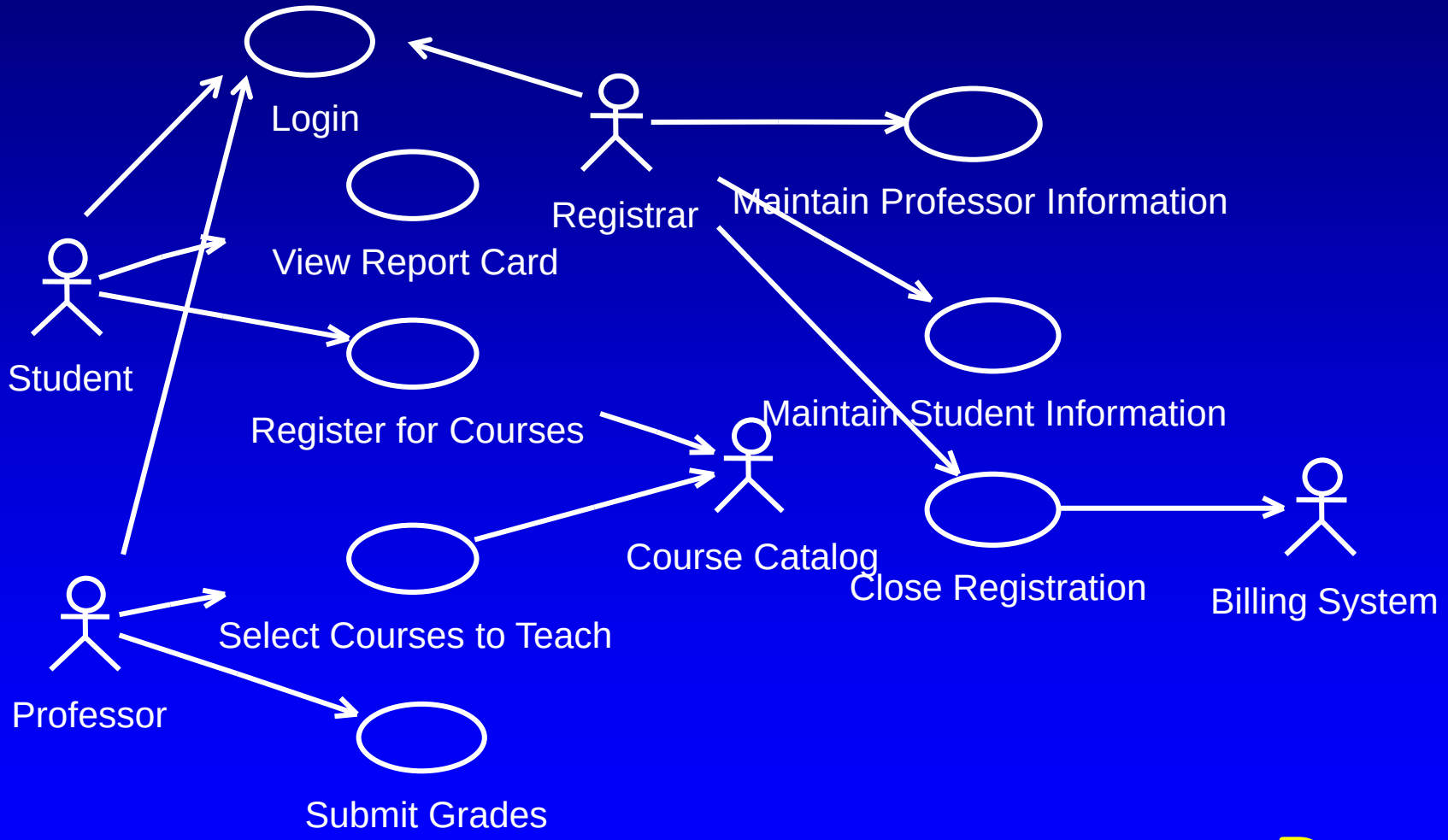


The UML Provides Standardized Diagrams



A Sample UML Diagram: Use Cases

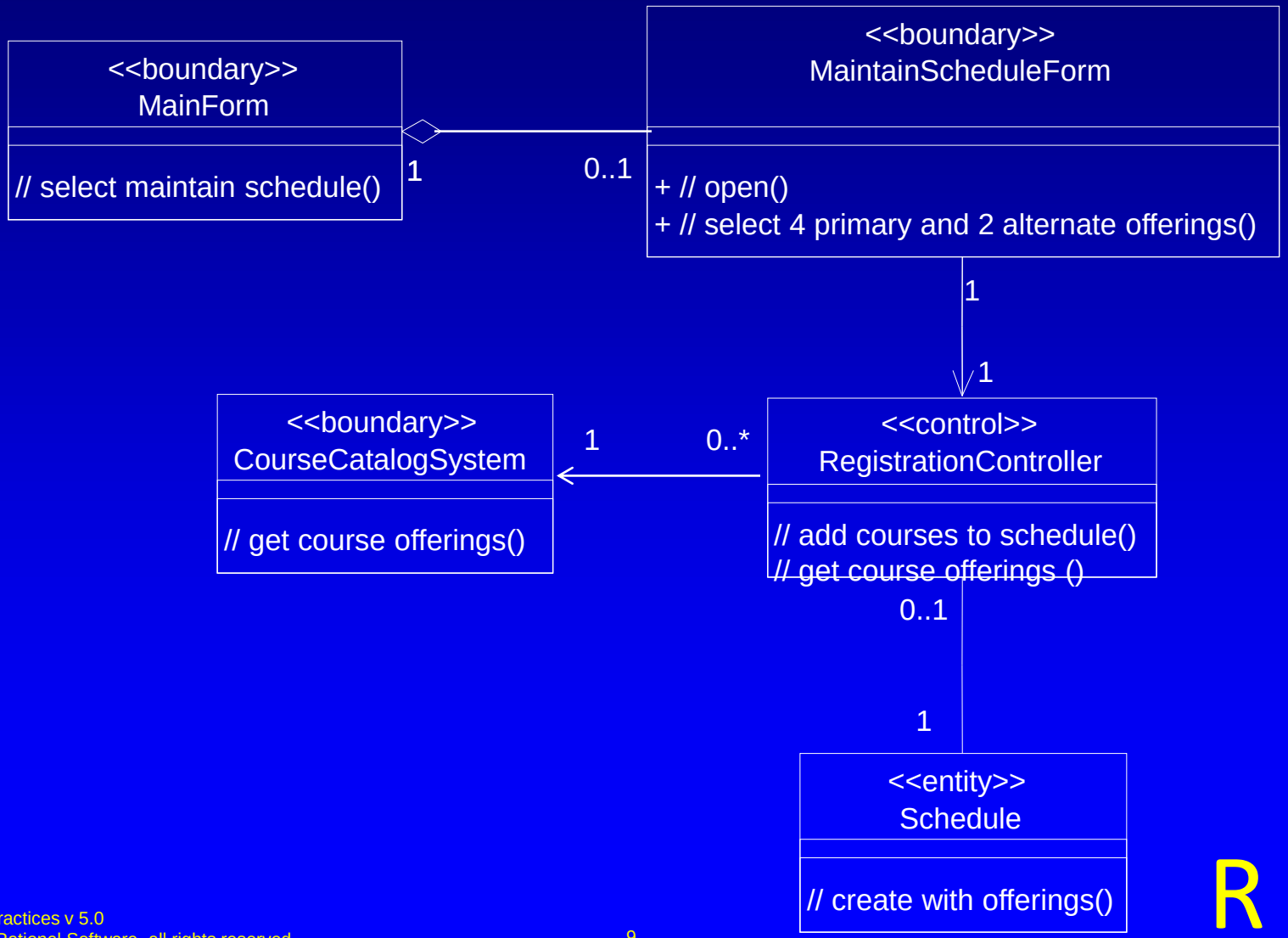
A University Course Registration System



R

A Sample UML Diagram: Classes

A University Course Registration System

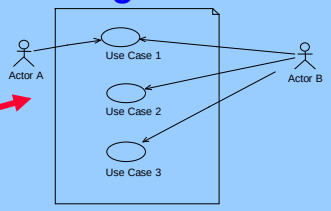


UML Diagrams Are Key System Artifacts

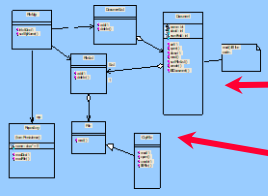
Domain Expert

User Interface Definition

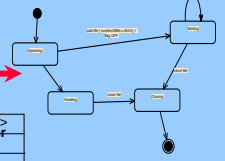
Use-Case Diagram



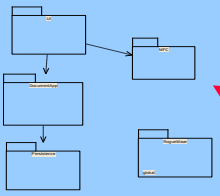
Class Diagram



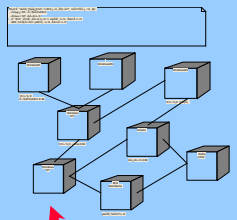
State Diagram

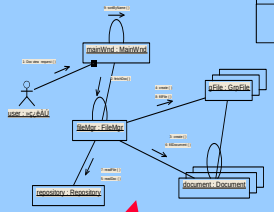


Class

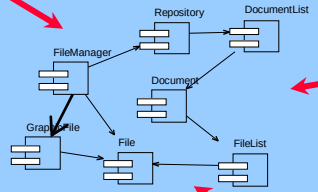


Deployment Diagram

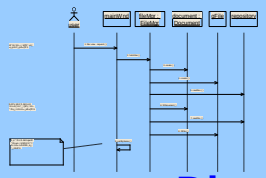




Package Diagram



Collaboration Diagram



Component Diagram

Forward Engineering(Code Generation)
and
Reverse Engineering

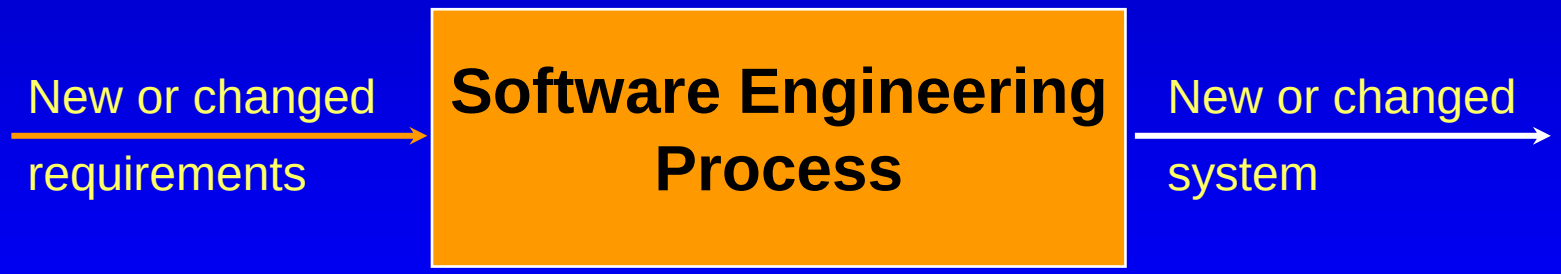
Source Code edit, compile, debug, link

Executable System

Sequence Diagram

What Is a Process?

A process defines **Who** is doing **What**, **When** and **How** to reach a certain goal. In software engineering the goal is to build a software product or to enhance an existing one



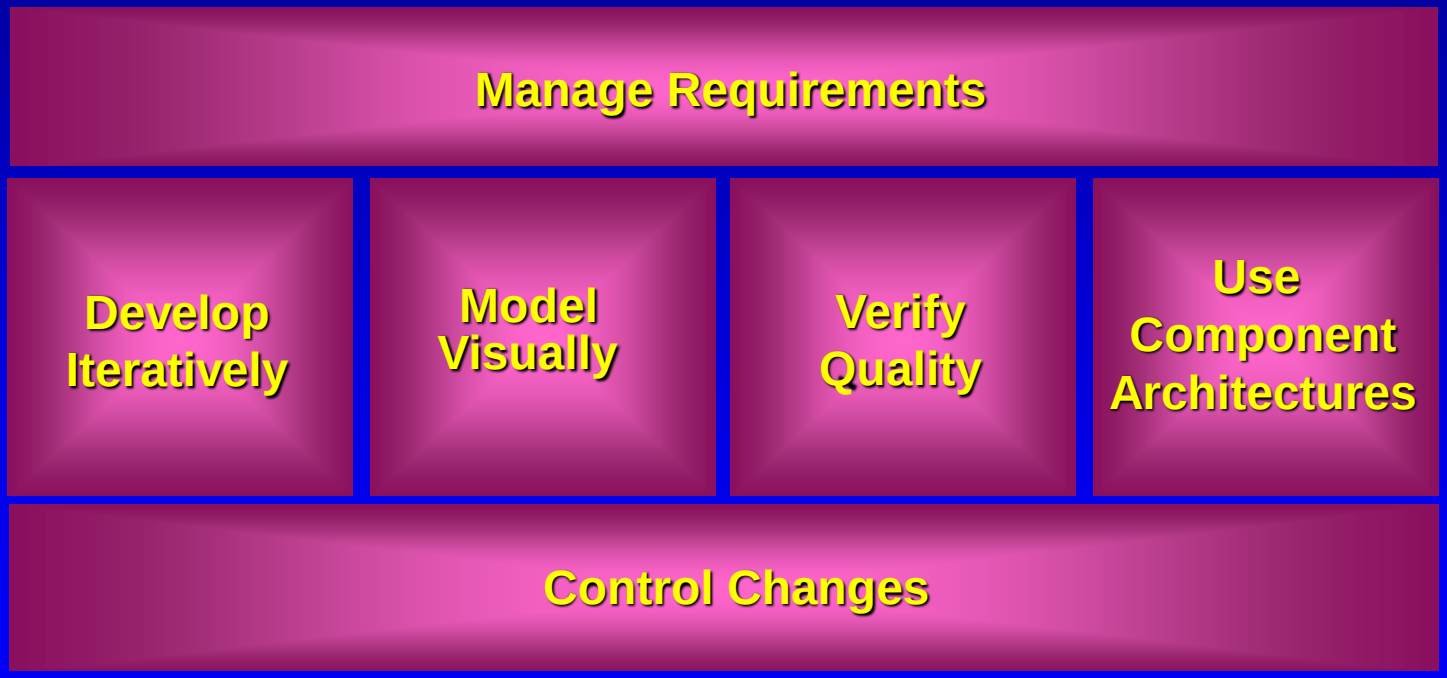
An Effective Process ...

- Provides guidelines for efficient development of quality software
- Reduces risk and increases predictability
- Captures and presents best practices
  - Learn from other's experiences
  - Mentor on your desktop
  - Extension of training material
- Promotes common vision and culture
- Provides roadmap for applying tools
- Delivers information on-line, at your finger tips

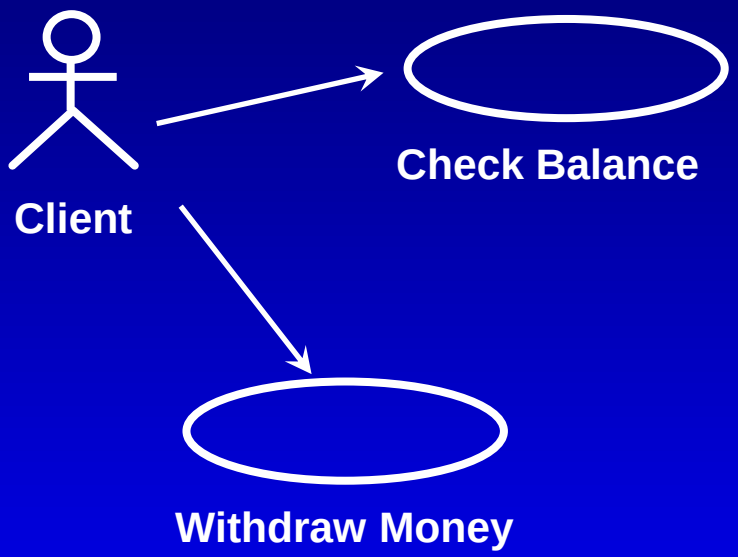
R

Rational Unified Process Delivers Best Practices

Rational Unified Process describes how to effectively implement the six best practices for software development



Rational Unified Process Is Use-Case Driven



Use Cases for a Cash Machine

An **actor** is someone or something outside the system that interacts with the system

A **use case** is a sequence of actions a system performs that yields an observable result of value to a particular actor

R

Use Cases Include a Flow of Events

Flow of events for the Withdraw Money Use Case

1. The use case begins when the client inserts her ATM card. The system reads and validates information on the card.
2. The system prompts for the PIN. The system validates the PIN.
3. The system asks which operation the client wishes to perform. The client selects “Cash withdrawal.”
4. The system requests the amount. The client enters the amount.
5. The system requests the account type. The client selects checking or savings.
6. The system communicates with the ATM network . . .

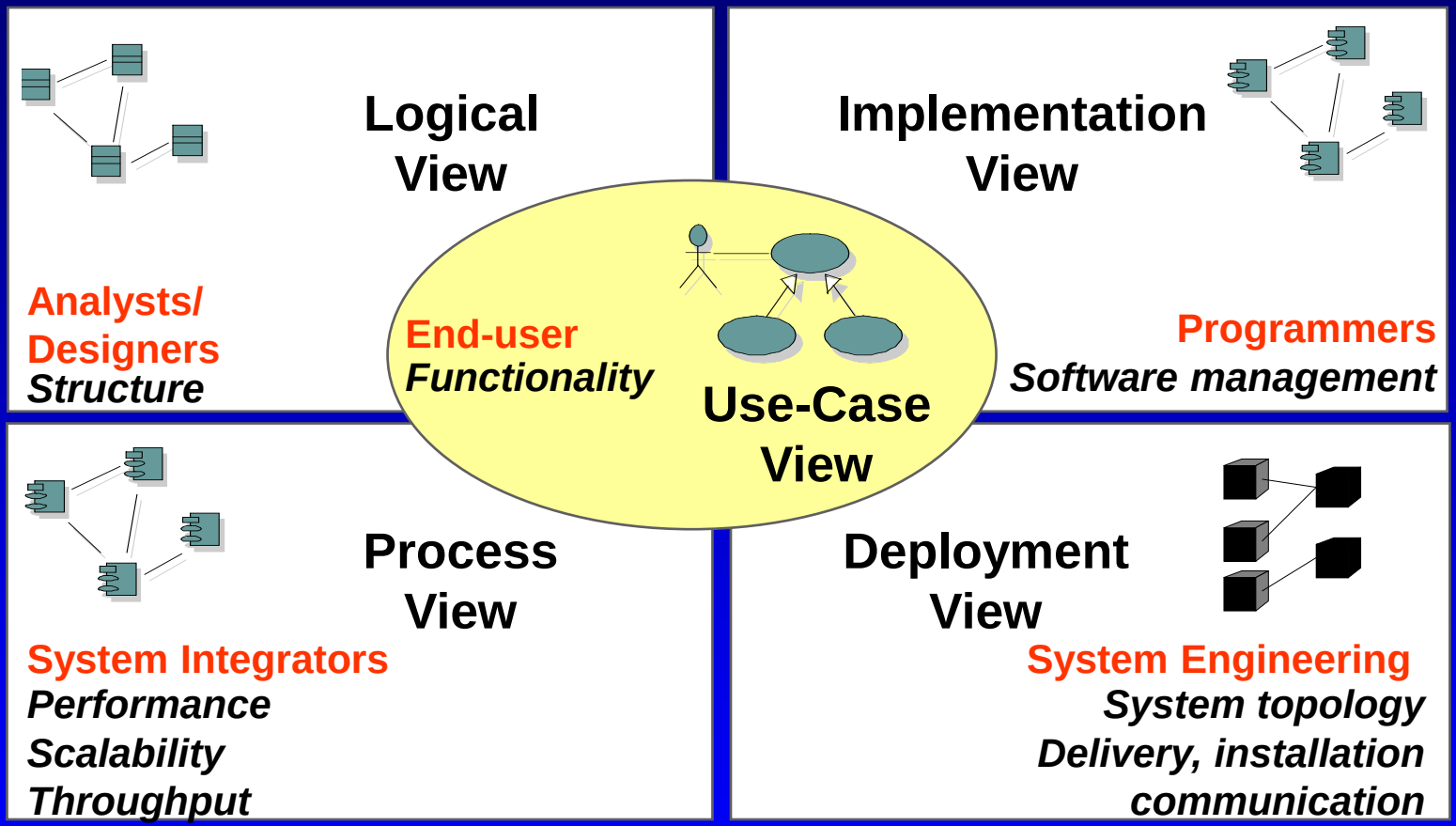
Benefits of a Use-Case Driven Process

- Use cases are concise, simple, and understandable by a wide range of stakeholders
  - End users, developers and acquirers understand functional requirements of the system
- Use cases drive numerous activities in the process:
  - Creation and validation of the design model
  - Definition of test cases and procedures of the test model
  - Planning of iterations
  - Creation of user documentation
  - System deployment
- Use cases help synchronize the content of different models

Rational Unified Process Is Architecture-Centric

- Architecture is the focus of the elaboration phase
  - Building, validating, and baselining the architecture constitute the primary objective of elaboration
- The Architectural Prototype validates the architecture and serves as the baseline for the rest of development
- The Software Architecture Description is the primary artifact that documents the architecture chosen
- Other artifacts derive from architecture:
  - Design guidelines including use of patterns and idioms
  - Product structure
  - Team structure

Representing Architecture: The 4+1 View Model

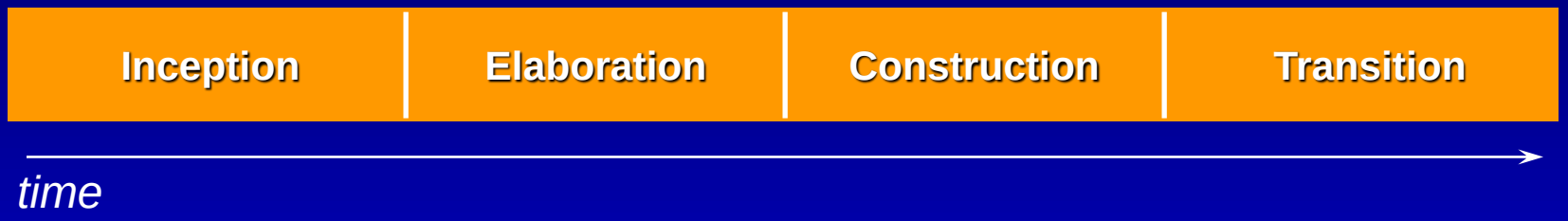


R

Benefits of an Architecture-Centric Process

- Architecture lets you gain and retain intellectual control over a project, to manage its complexity, and to maintain system integrity
- Architecture provides an effective basis for large-scale reuse
- Architecture provides a basis for project management
- Architecture facilitates component-based development
  - A component fulfills a clear function in the context of a well-defined architecture
  - A component conforms to and provides the physical realization of a set of interfaces
  - Components exist relative to a given architecture

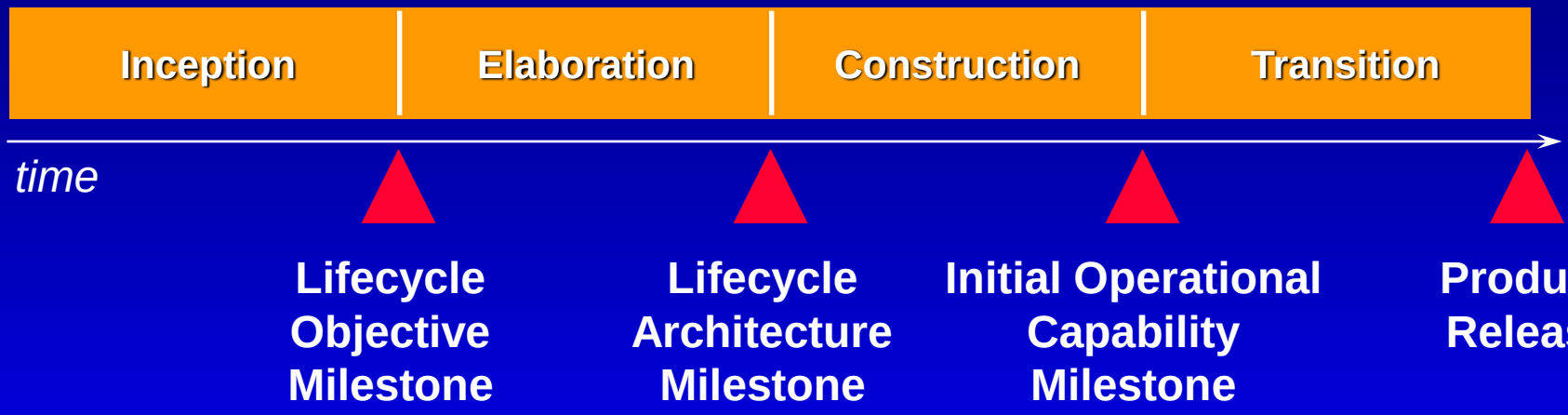
Process Architecture - Lifecycle Phases



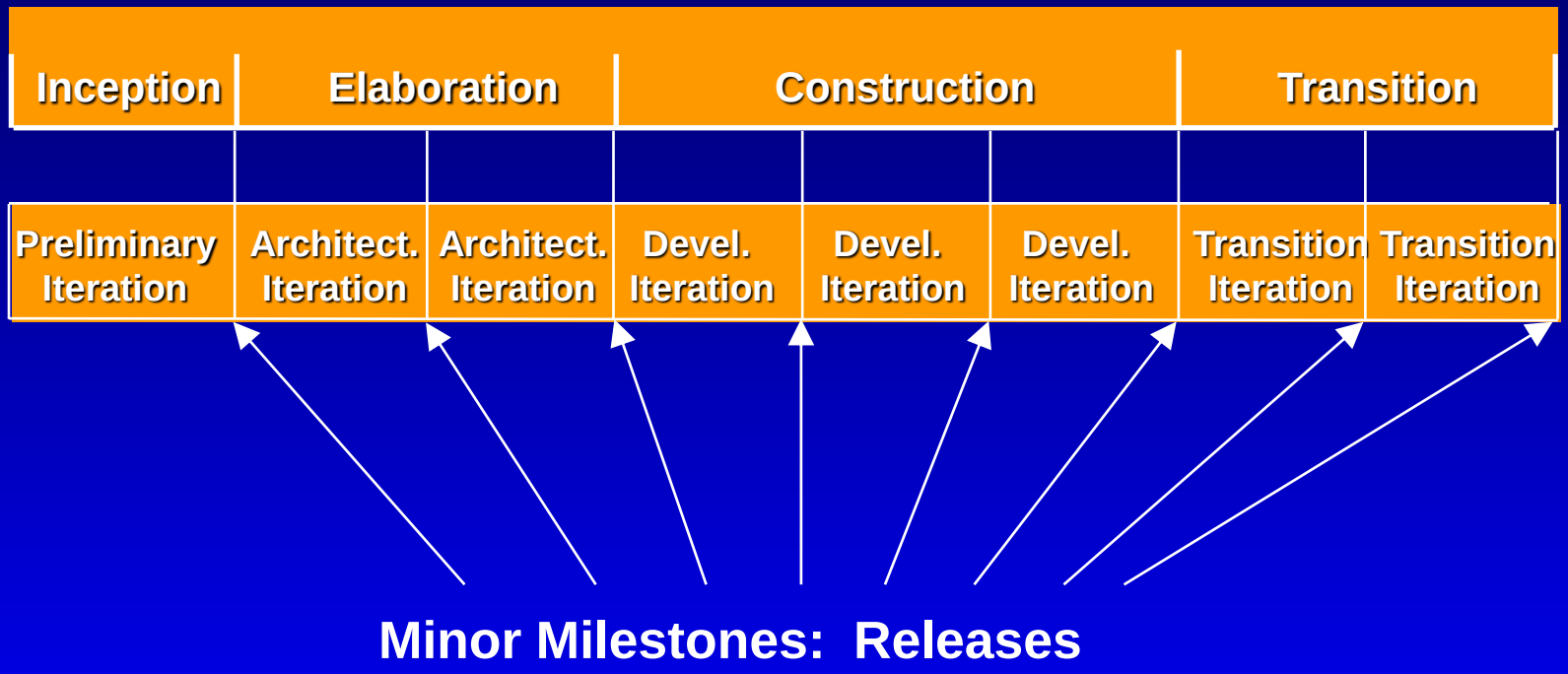
The Rational Unified Process has four phases:

- **Inception** - Define the scope of project
- **Elaboration** - Plan project, specify features, baseline architecture
- **Construction** - Build the product
- **Transition** - Transition the product into end user community

Phase Boundaries Mark Major Milestones



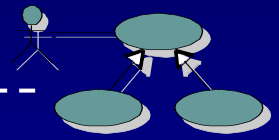
Iterations and Phases



An **iteration** is a distinct sequence of activities with an established plan and evaluation criteria, resulting in an executable release (internal or external)

Major Workflows Produce Models

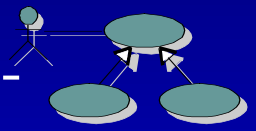
Business Modeling



Business Model

supported by

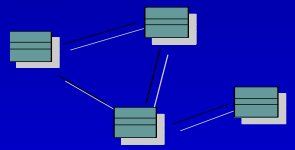
Requirements



Use-Case Model

realized by

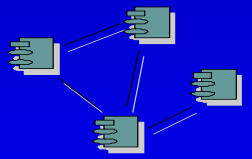
Analysis & Design



Design Model

implemented by

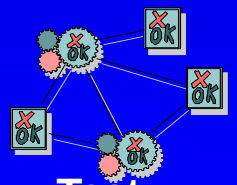
Implementation



Implementation Model

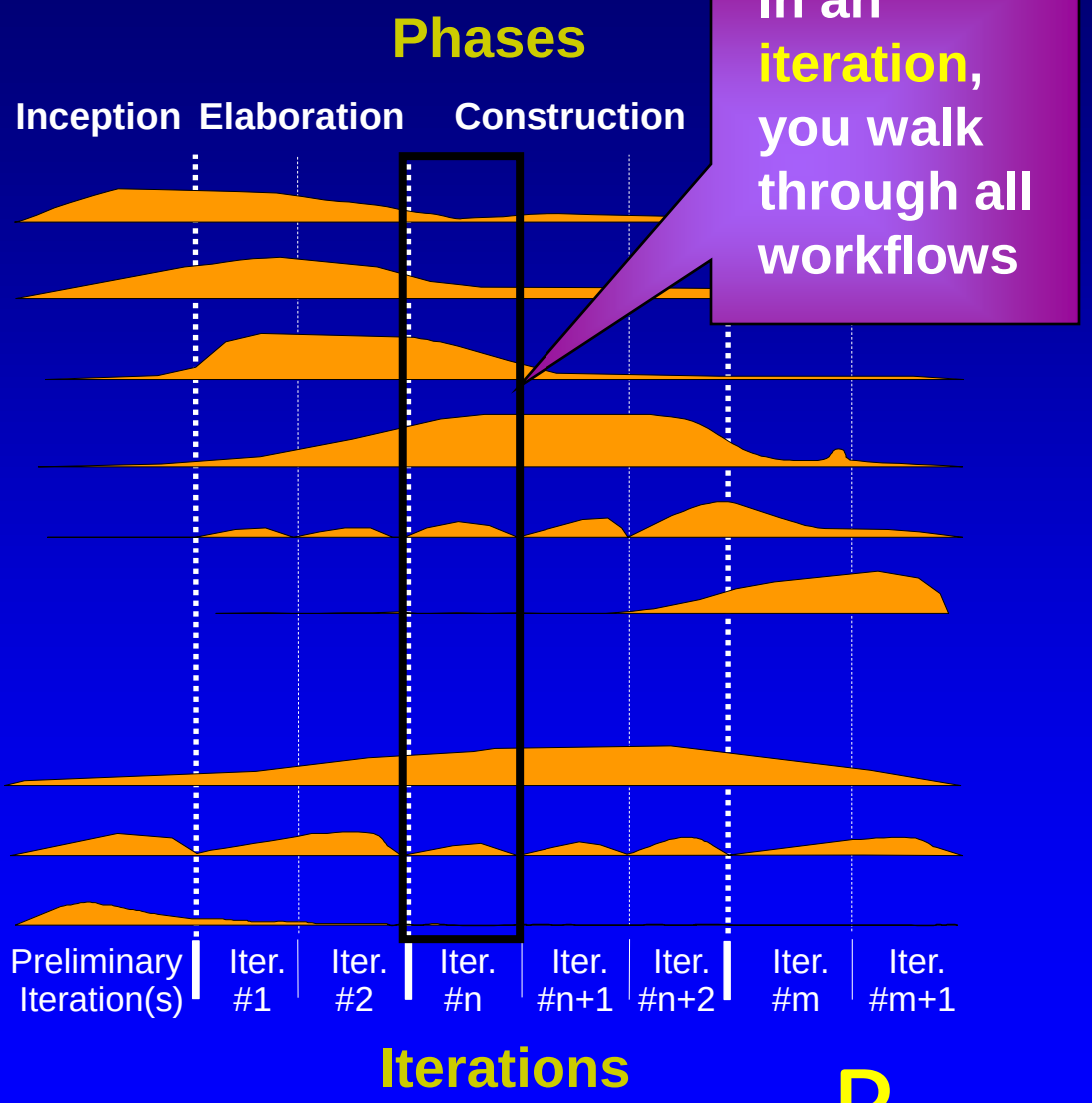
verified by

Test



Test Model

Bringing It All Together: The Iterative Model

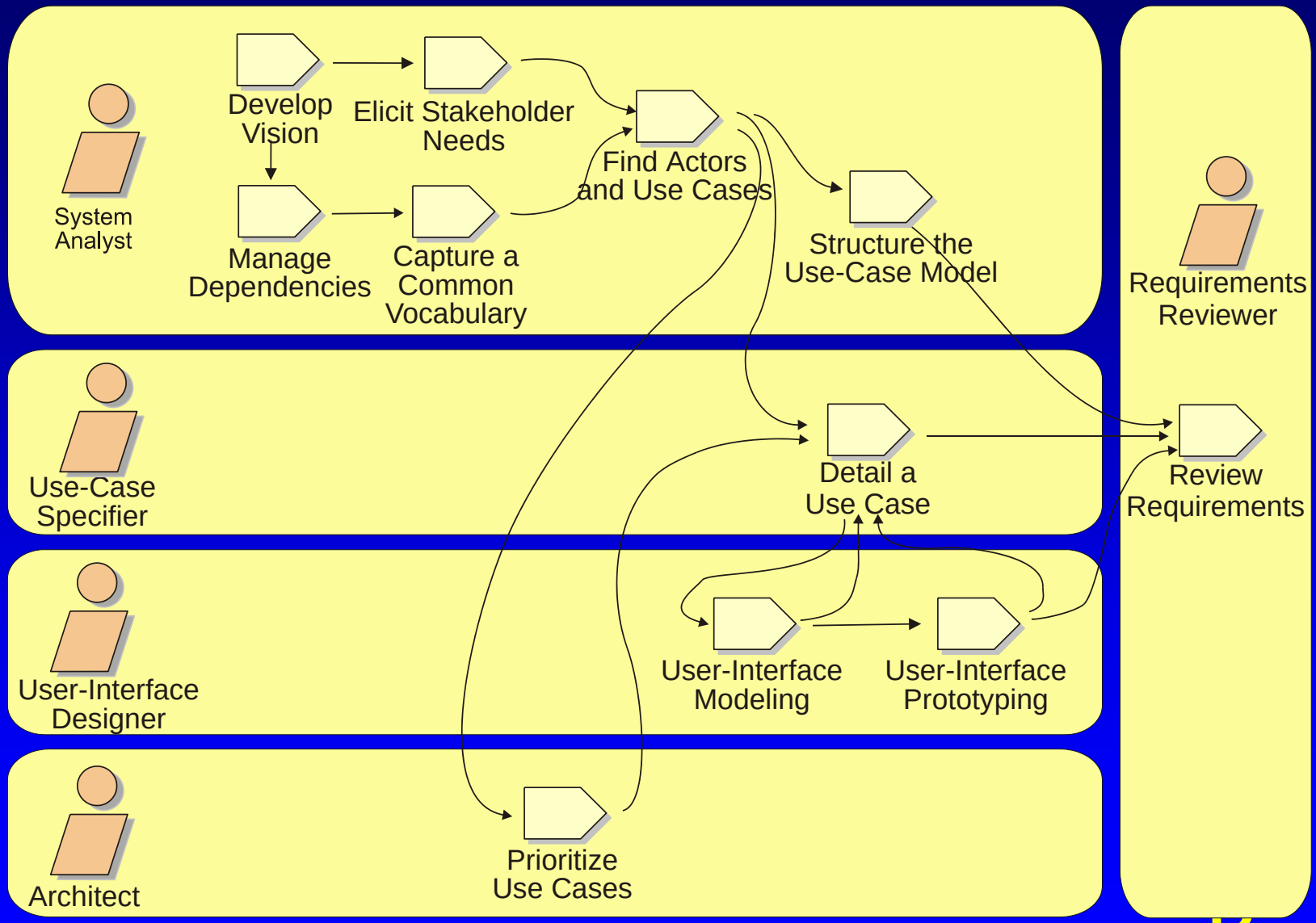


In an **iteration**, you walk through all workflows

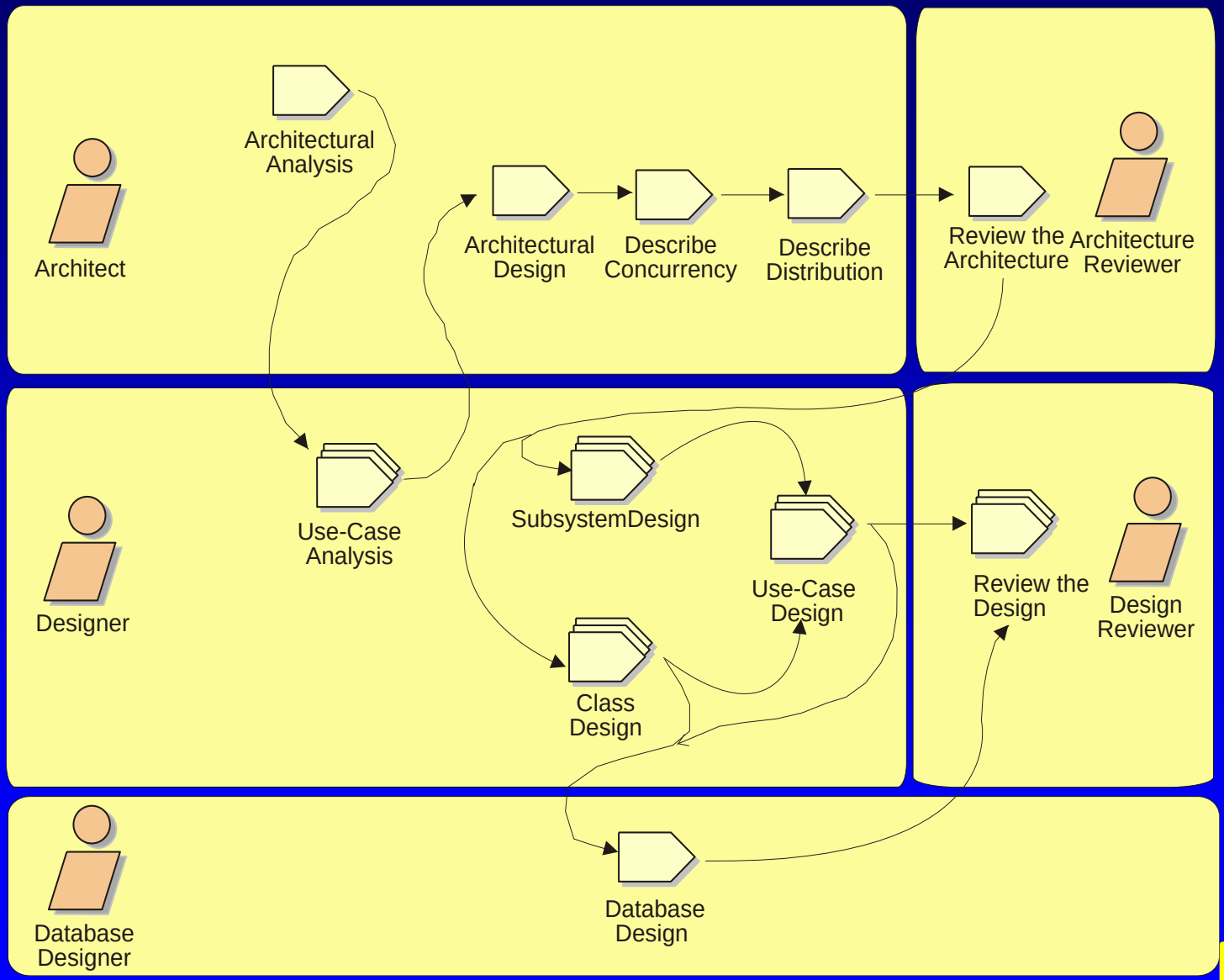
Workflows group activities logically

R

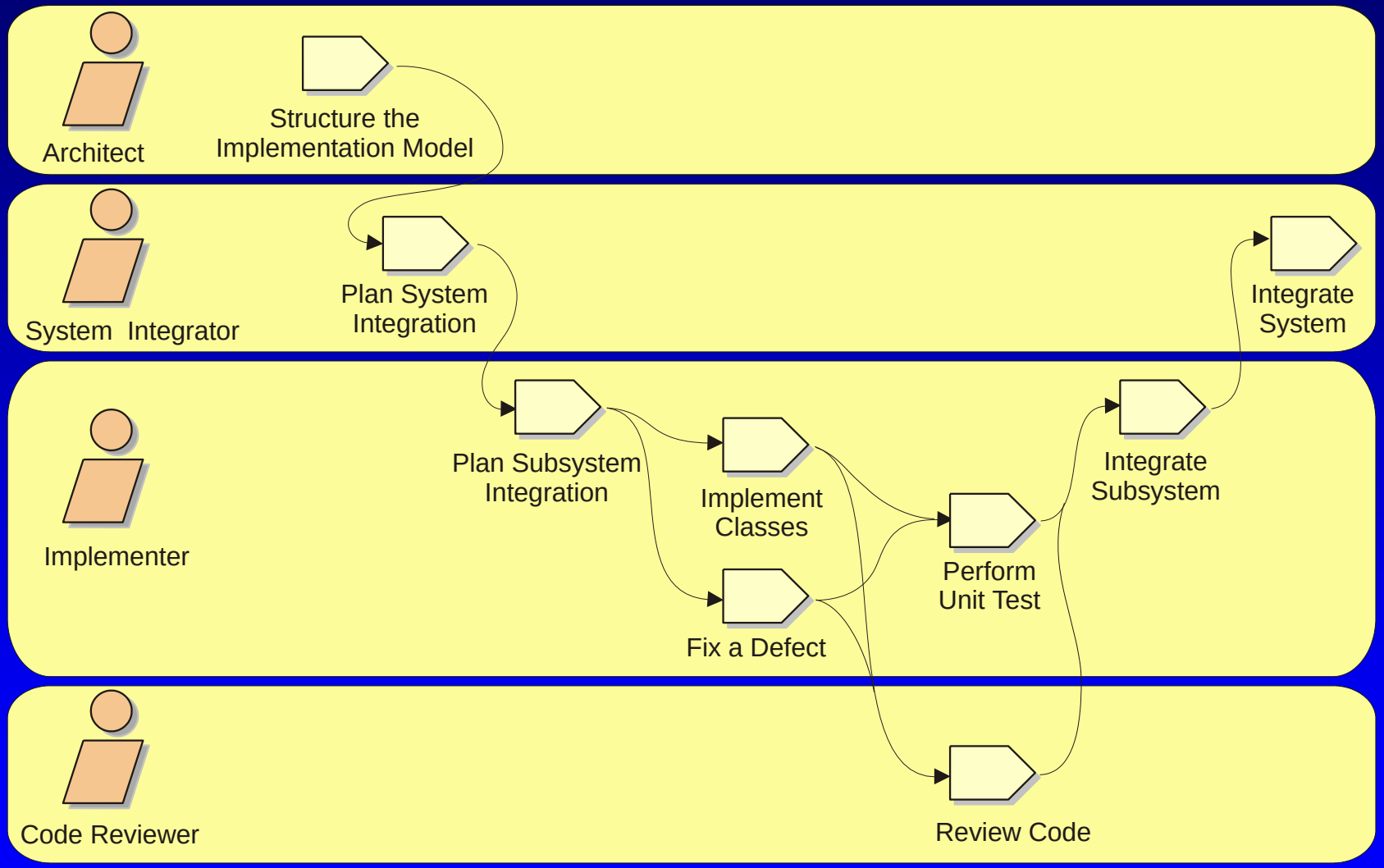
Requirements Workflow



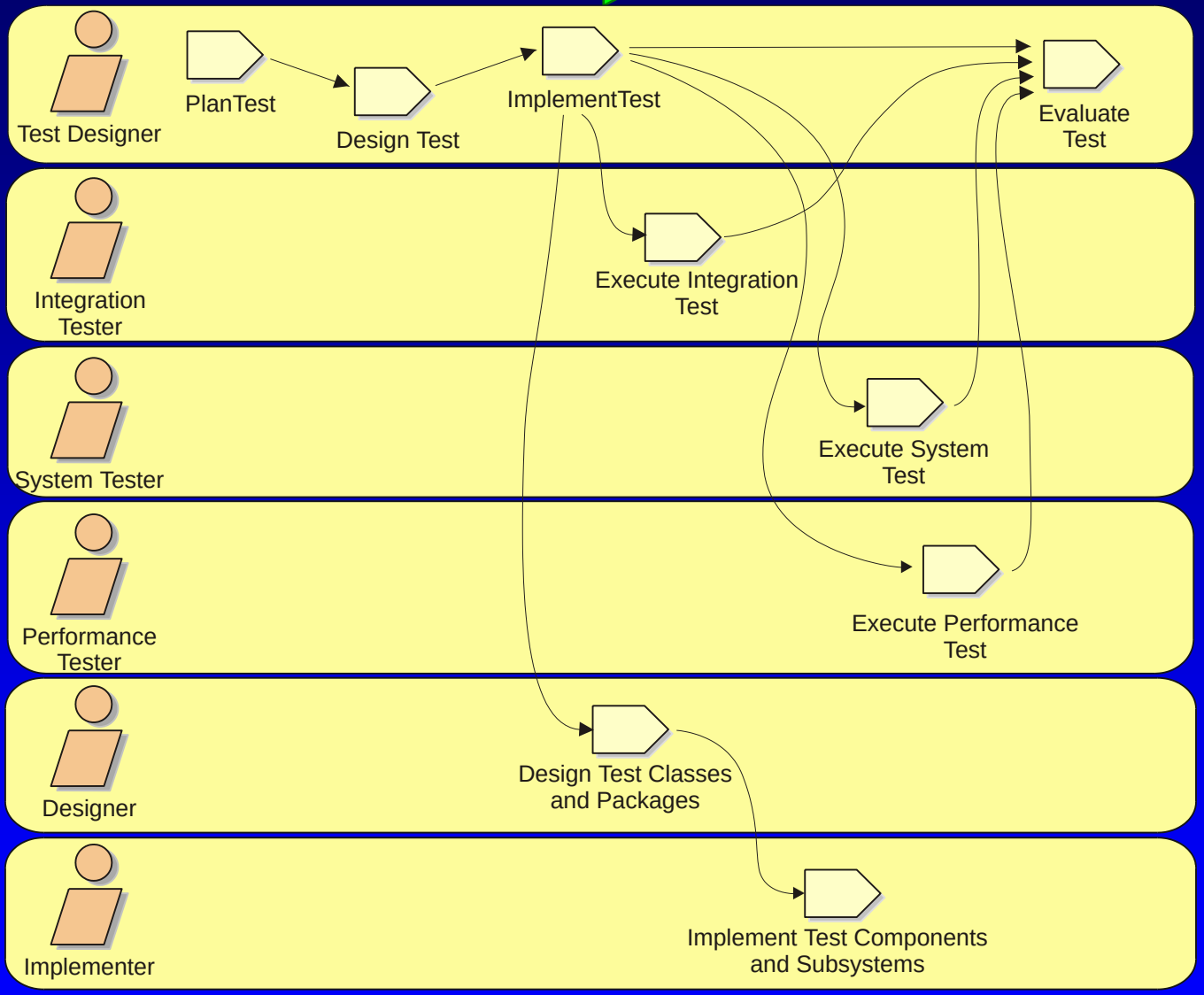
Analysis & Design Workflow



Implementation Workflow



Test Workflow



Summary: Rational Unified Process

- The Unified Modeling Language (UML) is a language for specifying, visualizing, constructing, and documenting the artifacts of a software-intensive system
- A software development process defines **Who** is doing **What, When** and **How** in building a software product
- The Rational Unified Process has four phases: **Inception, Elaboration, Construction and Transition**
- Each phase ends at a major milestone and contains one or more iterations
- An **iteration** is a distinct sequence of activities with an established plan and evaluation criteria, resulting in an executable release