



IBM Software Group

# Essentials of Rational XDE

## Organizing Models in XDE

**Rational**® software



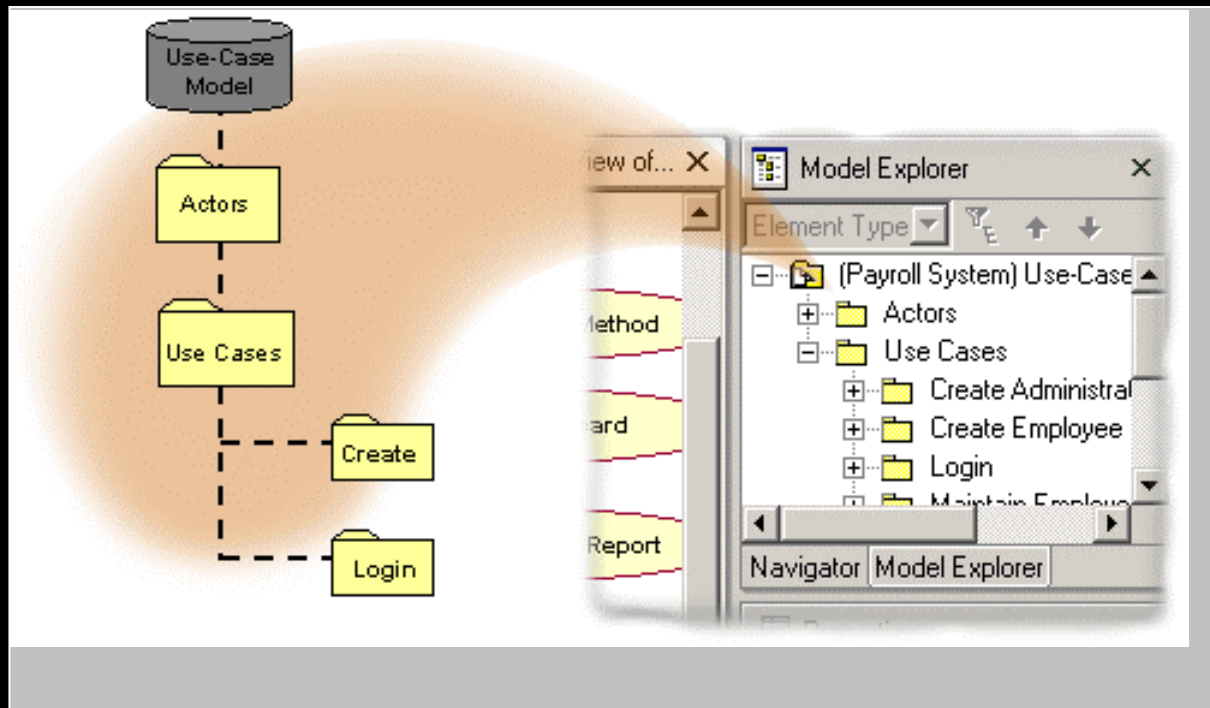
# Objectives

---

- ◆ Explain how packages group and organize your model
- ◆ Show how to use model templates for organizing projects by employing a:
  - Use-case model
  - Analysis model
  - Design model

# Create Packages to Organize Your Model

- ◆ Packages group and organize your model
- ◆ Each package name should be simple and adequately describe the package contents



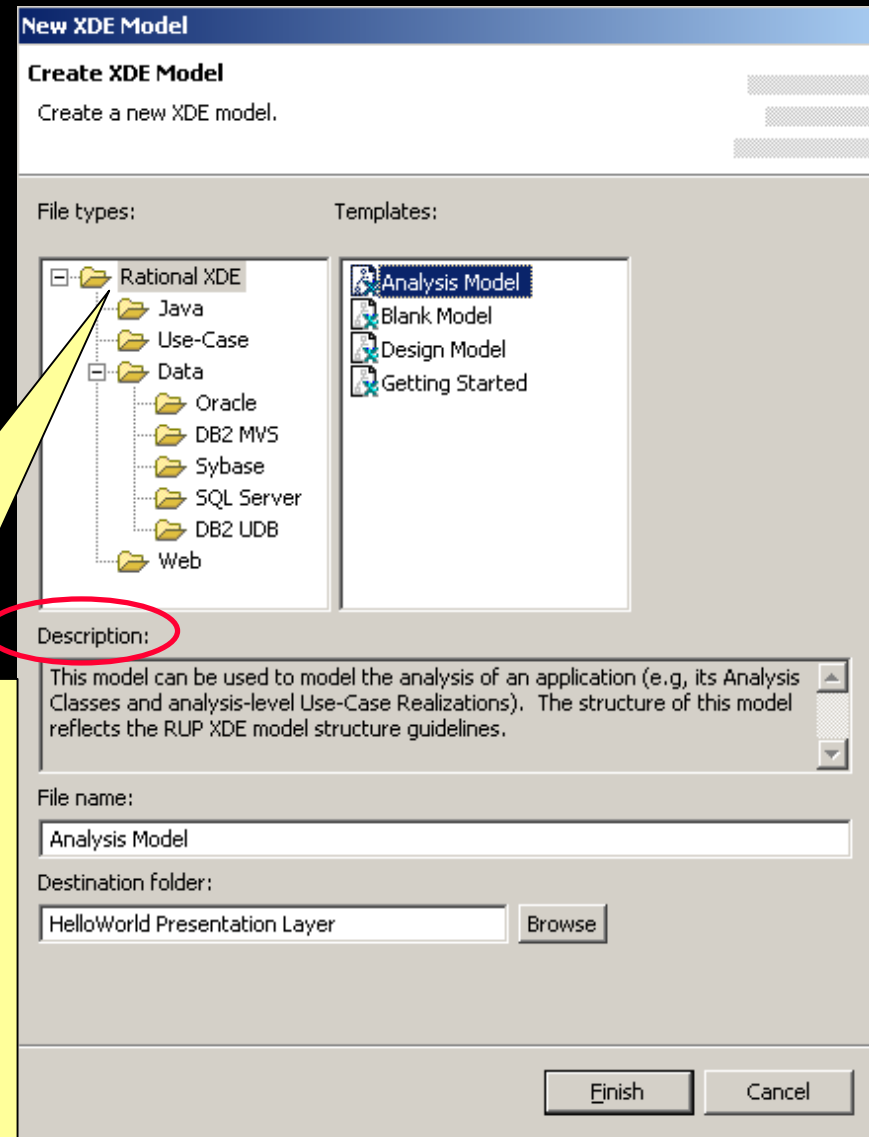
# Rational XDE Model Templates

- ◆ Use model templates for organizing project artifacts:
  - Use-Case model
  - Analysis model
  - Design model
- ◆ Provide RUP support via package structure
- ◆ Number of projects and model configuration is an architectural decision
  - Templates provide a starting point

# Rational XDE Model Templates (continued)

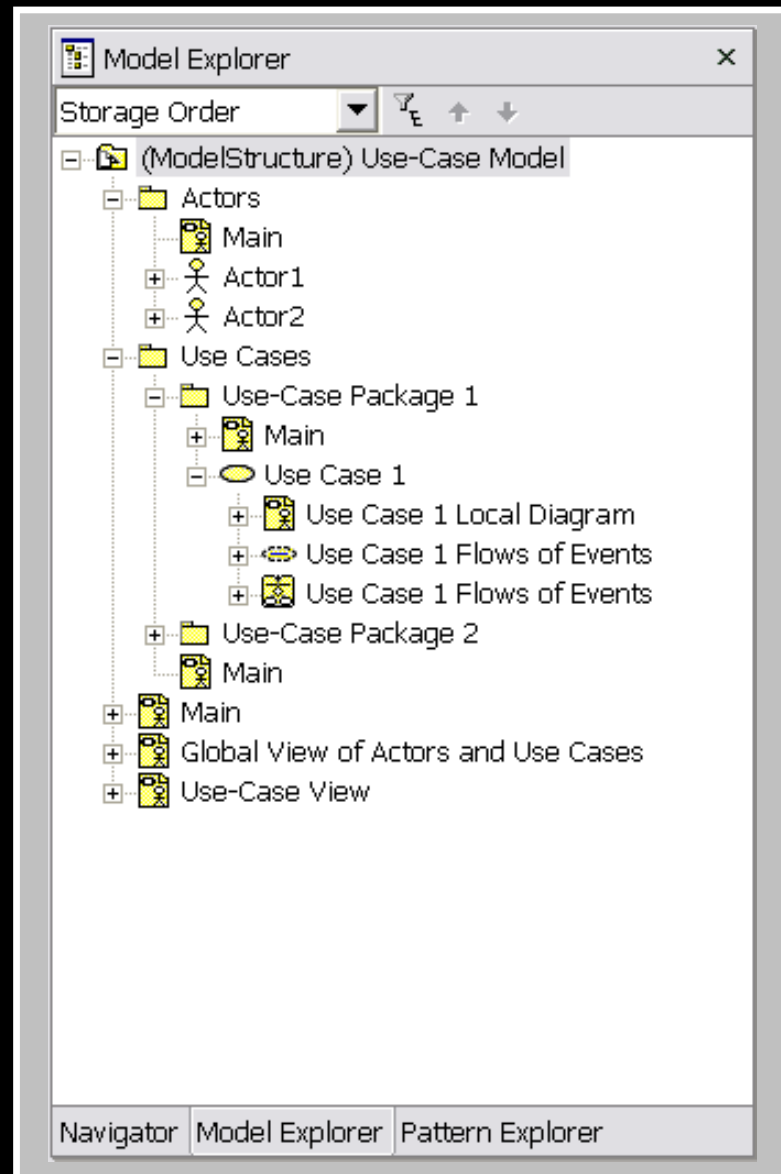
- ◆ Assist you when starting a project
- ◆ Meet most modeling objectives
- ◆ Come with descriptions, displayed in the dialog box

To select a model template, click on **Rational XDE** in the **File types** list box and select a model template from the **Templates** list box that meets your needs.



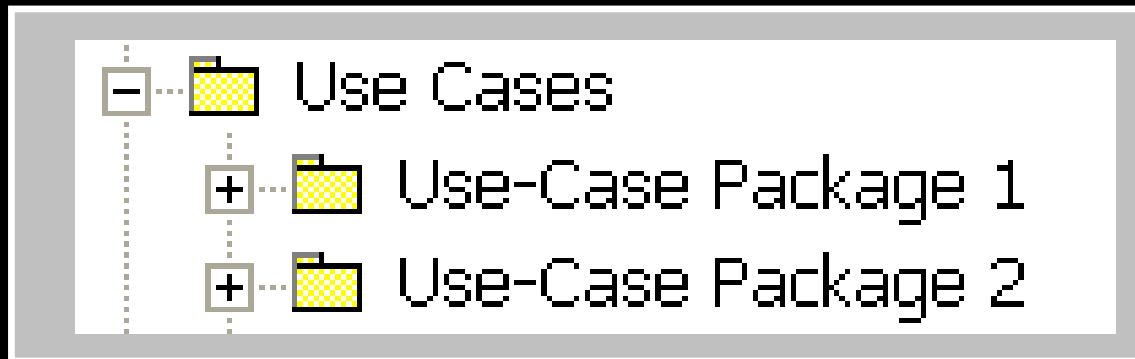
# Use-Case Model

- ◆ Contains the actors and use cases and any diagrams that are needed to clarify different aspects of the use cases.
- ◆ Maintains the recommended structure illustrated here:



# Create Additional Use-Case Packages

- ♦ To organize the contained model elements in the **Actors** and **Use-Case** packages
- ♦ For example:



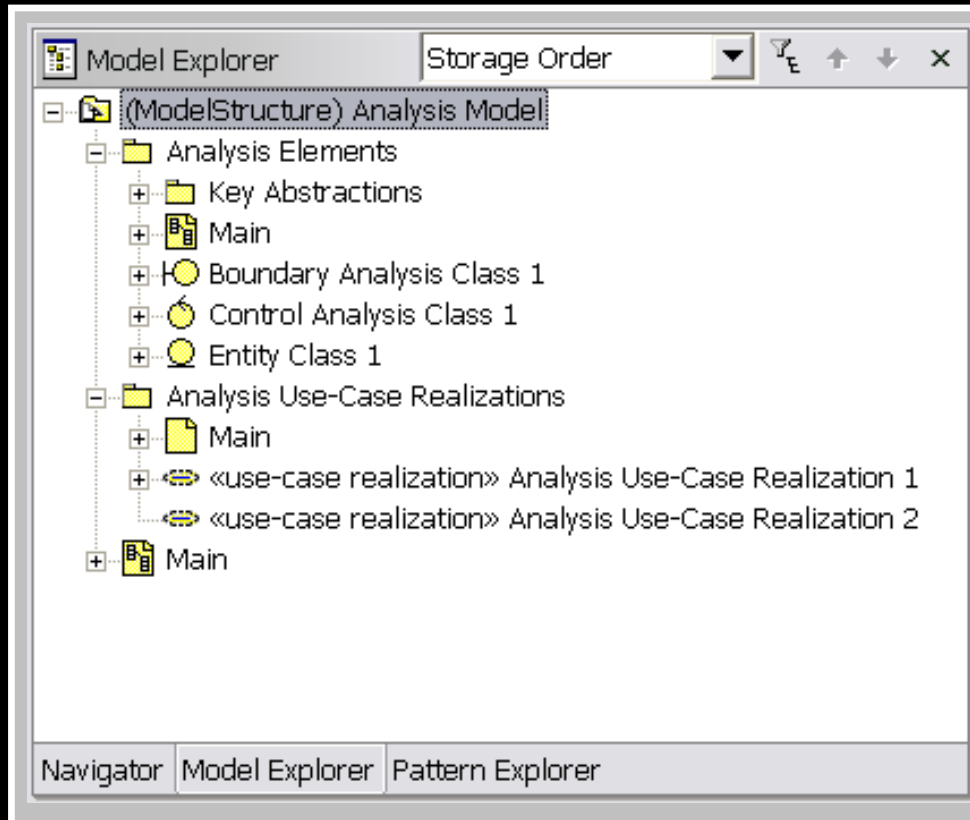
# Additional Use-Case Model Diagrams

- ◆ In addition to the use-case model diagrams that contain actors and use cases, additional diagrams can be included to clarify different aspects of the use cases.
  - Local use-case diagram
    - Specifies a single use case and the actors that participate in that use case
  - Sequence diagram
    - Specifies a Single Flow of Events for a particular use case
  - Activity diagram
    - Graphically describes the flow of events described in a use-case description



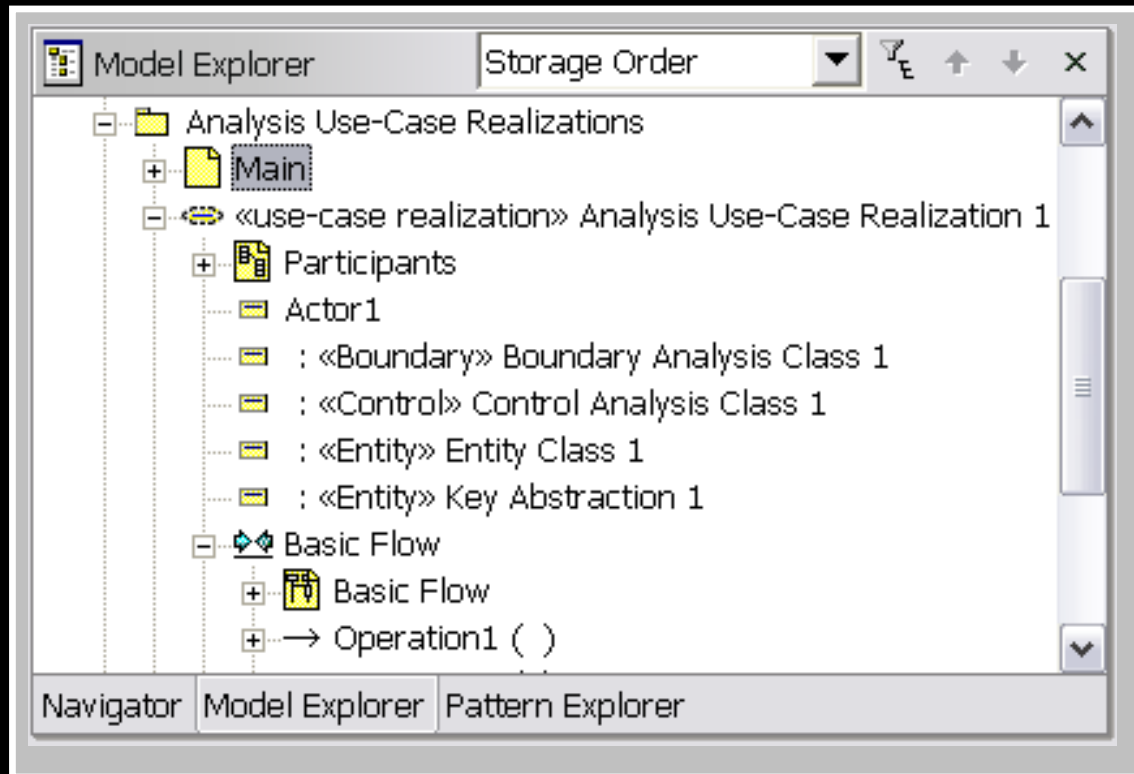
# Analysis Model

- ♦ Is where the analysis classes and use-case realizations reside.
- ♦ Maintains the recommended structure illustrated here:



# Analysis Use-Case Realization Package

- ◆ Contains the analysis-level use-case realizations
- ◆ Maintains the recommended structure illustrated here:

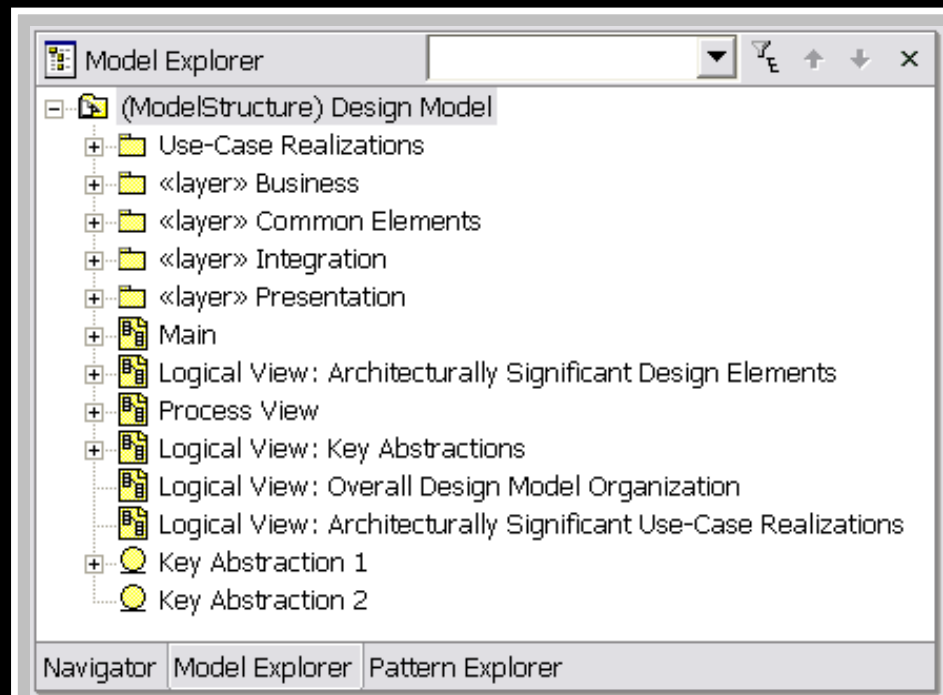


# The Design Model

- ◆ Describes the design of the application
- ◆ Contains elements that span multiple XDE code models
- ◆ Contains the logical partitions that inspire the organization of the individual code models
- ◆ Provides a place for design elements

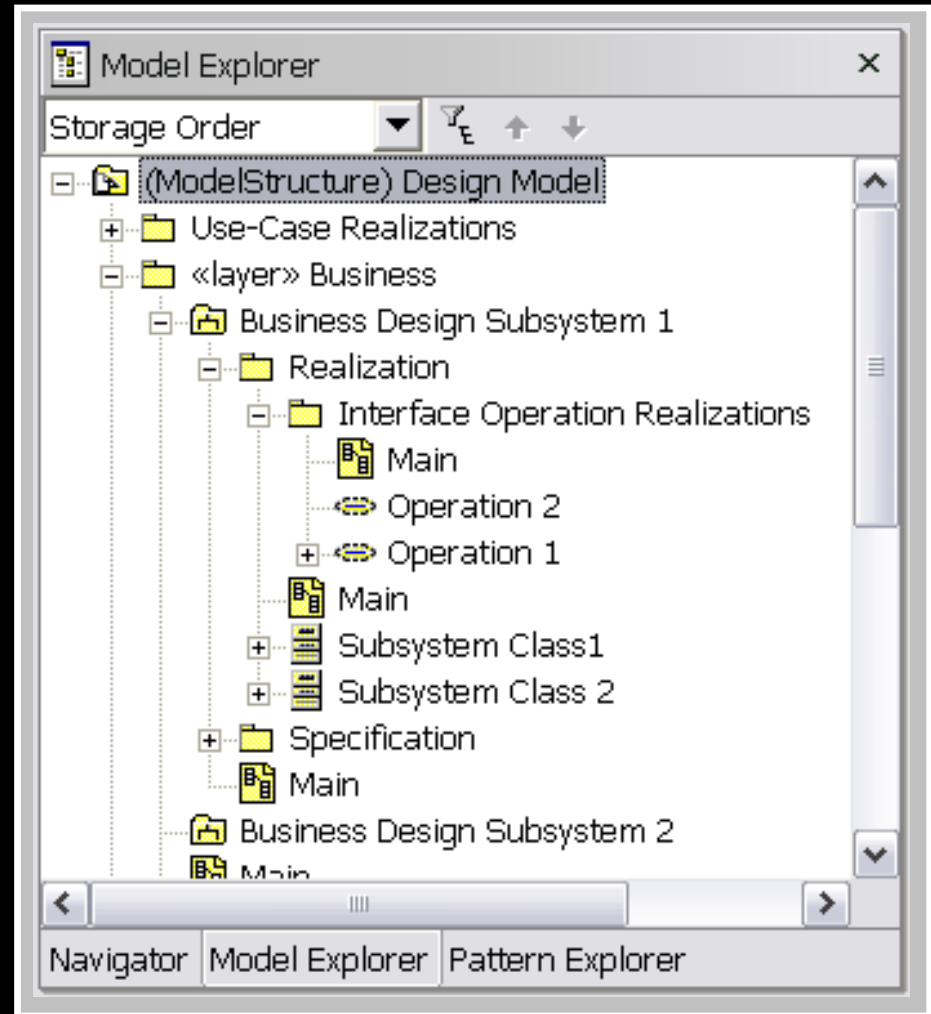
# The Design Model (continued)

- ◆ Contains layer packages that reference the design elements of the system, such as:
  - Design classes
  - Interfaces
  - Design subsystems
- ◆ Maintains the recommended structure illustrated here:



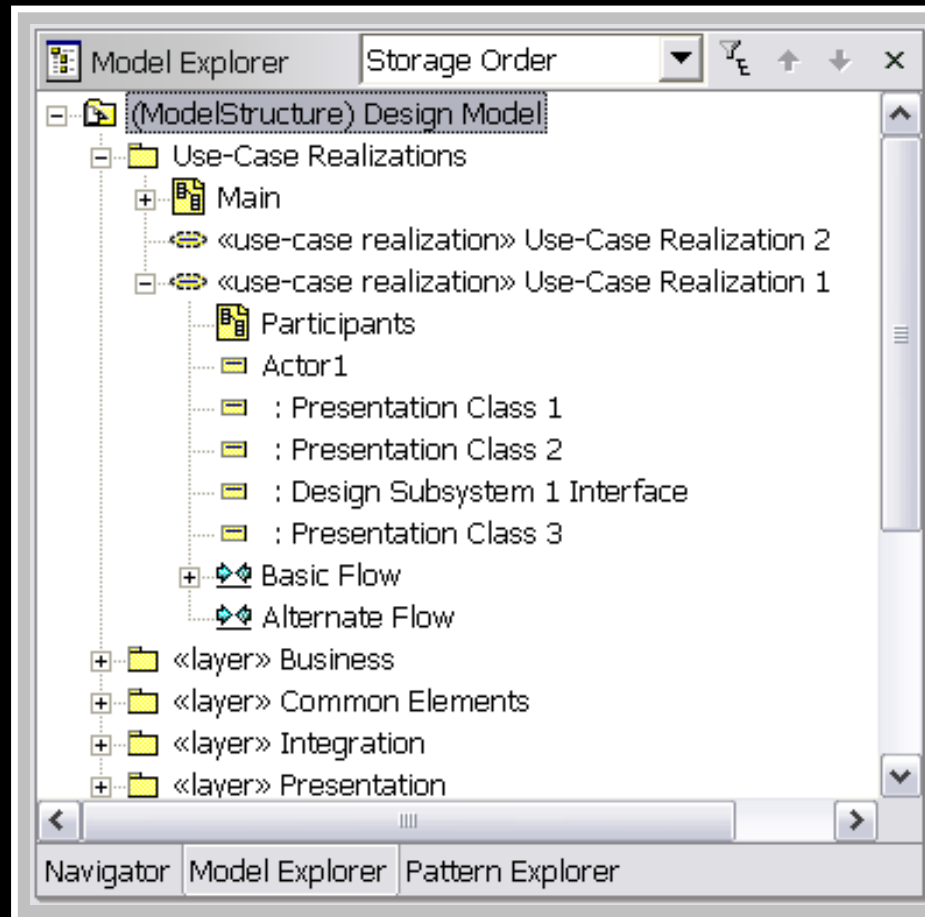
# Design Subsystem

- ◆ Supports the definition of separate “Specification” and “Realization” packages within the design subsystem package.
- ◆ Maintains the recommended structure illustrated here:



# Design Use-Case Realization

- ◆ Each use-case realization is associated with a use case in the use-case model



# Lab: Organizing Models in Rational XDE

- ◆ In this lab, you will:
  - Create a new project
  - Add model templates for:
    - Use case
    - Analysis
    - Design
  - Explore models



# More Information

- ◆ Other IBM courseware that provides insights into the RUP and XDE includes:
  - PRJ270: Essentials of the Rational Unified Process
  - DEV475: Mastering Object-Oriented Analysis and Design with UML
  - DEV490: Mastering Rational XDE for J2EE Enterprise Development with WSAD 5.0



# Summary

---

- ◆ Model templates assist you in starting a project. Choose the model template that best suits your modeling objectives.
- ◆ A use case is a model element that describes behaviors that a system performs to yield observable, valuable results for actors.
- ◆ Analysis classes represent an early conceptual model for things in the system that have responsibilities and behavior.
- ◆ The design model contains architecture definitions and the design details of the architecture.

