
Phân Tích và Thiết Kế Hướng Đối Tượng Sử dụng UML

cuu duong than cong . com

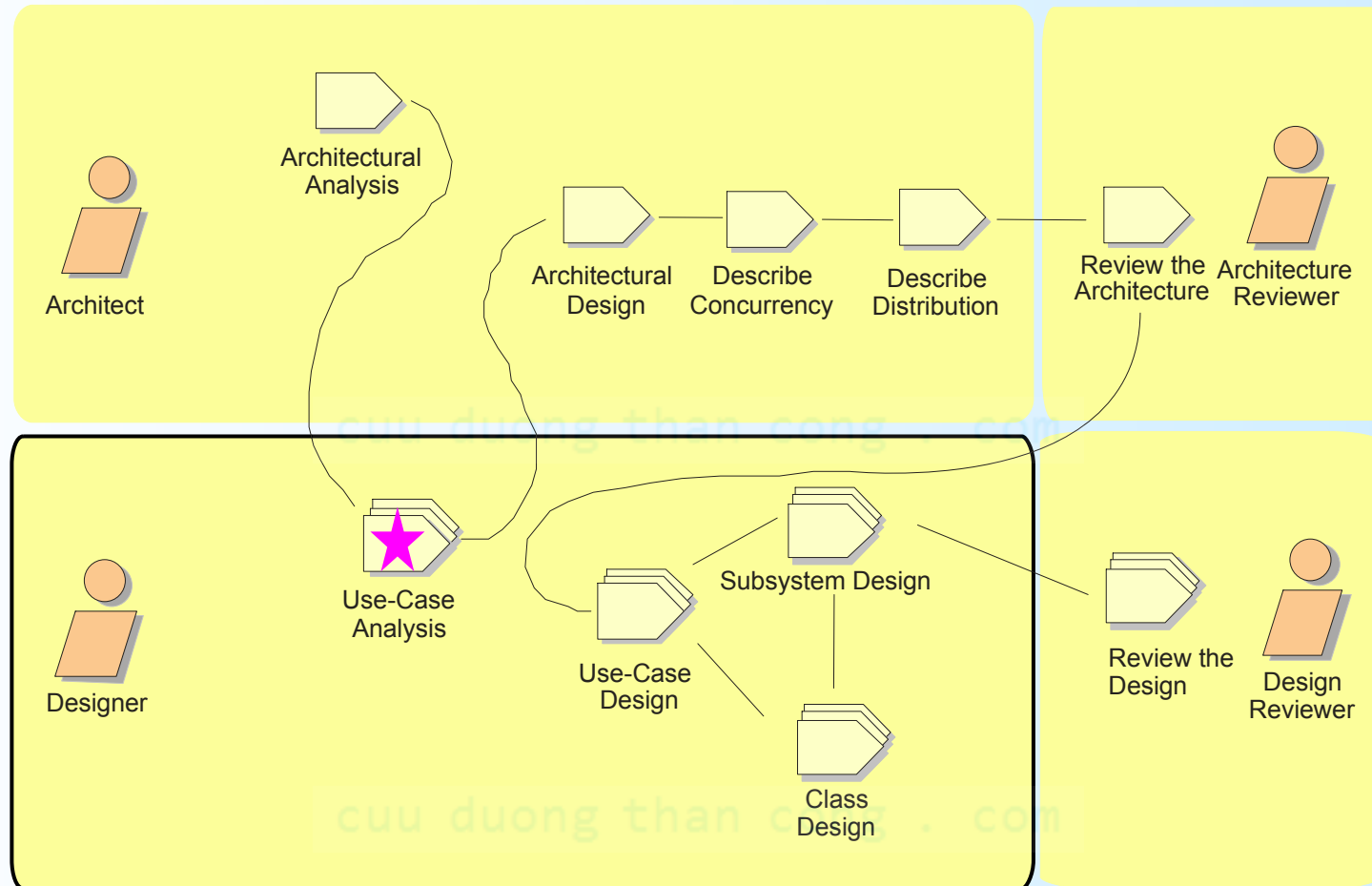
Use-Case Analysis

cuu duong than cong . com

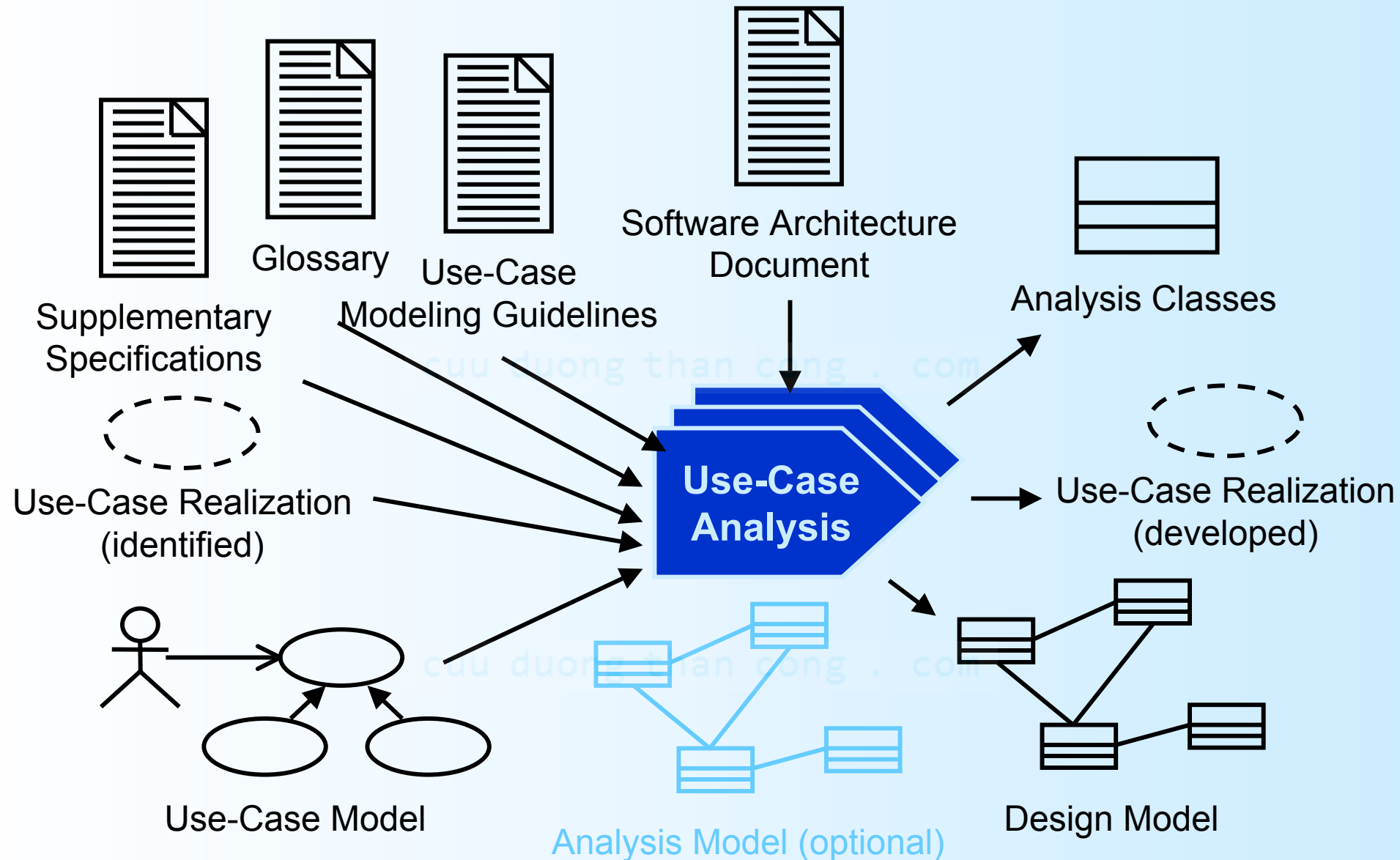
Mục tiêu: Use-Case Analysis

- Tìm hiểu mục đích của công đoạn Use-Case Analysis vị trí của nó trong chu kỳ sống của QT PTPM
- Xác định các class thực hiện một use- case flow of events
- Phân phối các hành vi (behaviour) của use-case về các class của nó, thông qua việc xác định nhiệm vụ của các class
- Phát triển các use-case realization mô hình hóa collaboration giữa các thể hiện của các class đã xác định

Use-Case Analysis trong ngữ cảnh



Tổng quan về Use-Case Analysis



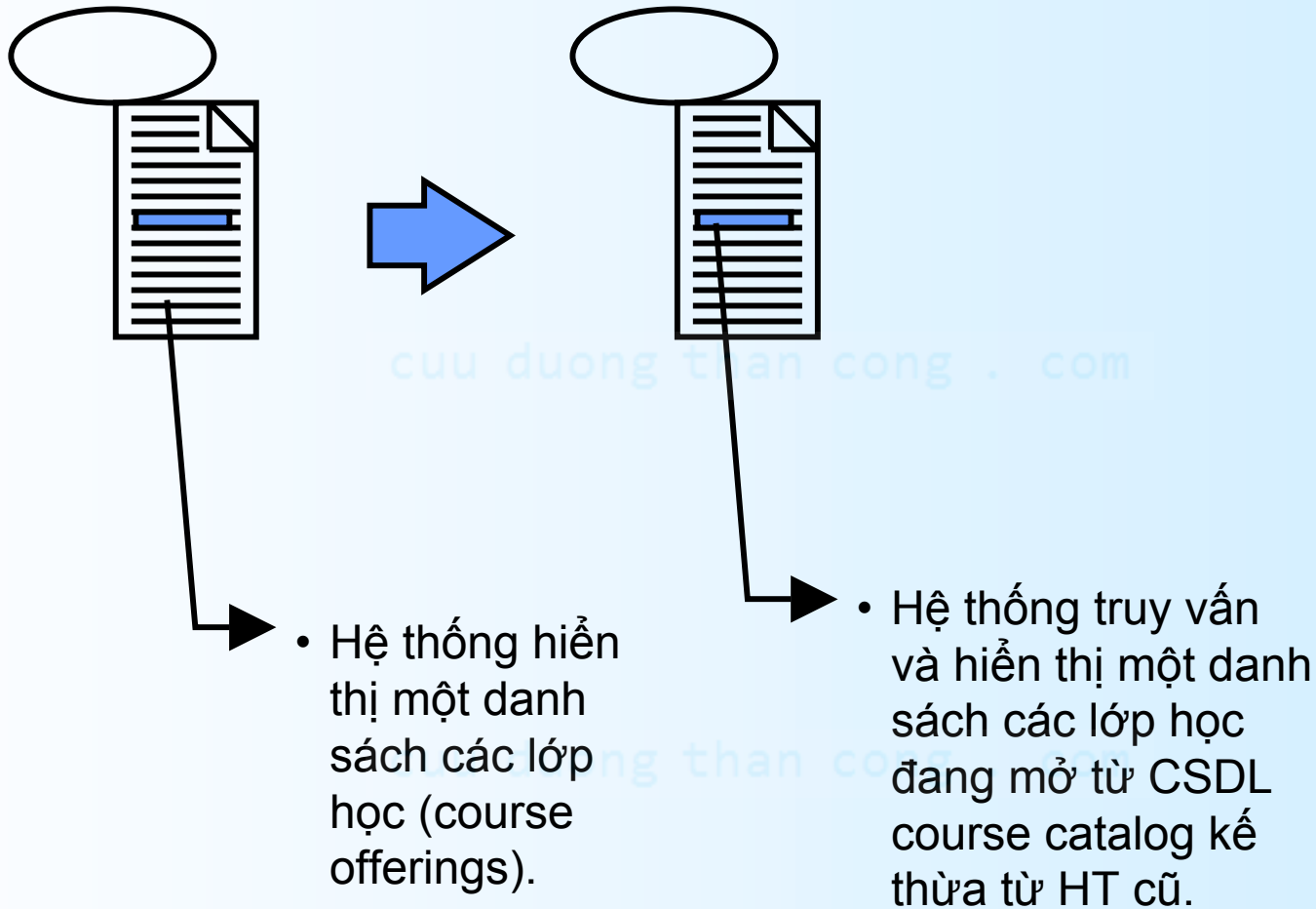
Các bước thực hiện phân tích Use-Case

- Bổ sung Use-Case Description
- Với mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior về các Class
- Với mỗi analysis class đã tìm thấy
 - Mô tả nhiệm vụ của chúng
 - Mô tả các Attribute và các Association
 - Lượng giá (qualify) các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

Các bước thực hiện phân tích Use-Case

- Bổ sung Use-Case Description
- Với mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior về các Class
- Với mỗi analysis class đã tìm thấy
 - Mô tả nhiệm vụ của chúng
 - Mô tả các Attribute và các Association
 - Lượng giá (qualify) các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

Bổ sung Use-Case Description

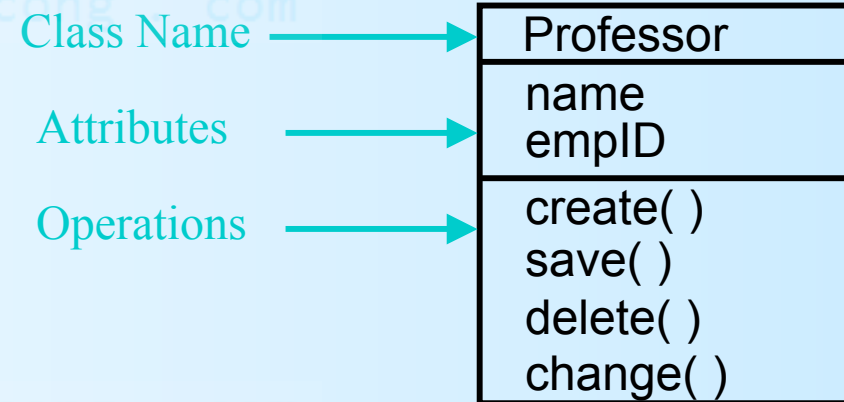


Các bước thực hiện phân tích Use-Case

- Bổ sung Use-Case Description
- Với mỗi use-case realization
 - ★ – Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior về các Class
- Với mỗi analysis class đã tìm thấy
 - Mô tả nhiệm vụ của chúng
 - Mô tả các Attribute và các Association
 - Lượng giá (qualify) các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

Review: Class

- Là một sự trừu tượng hóa
- Mô tả một nhóm các đối tượng có chung:
 - Properties (attributes)
 - Behavior (operations)
 - Relationships
 - Ngữ nghĩa (Semantics)



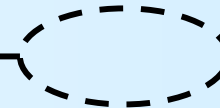
Review: Use-Case Realization

Use-Case Model

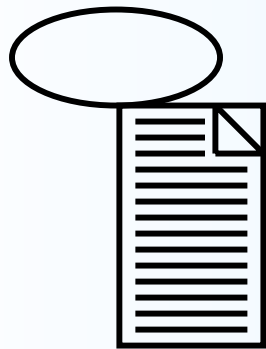


Use Case

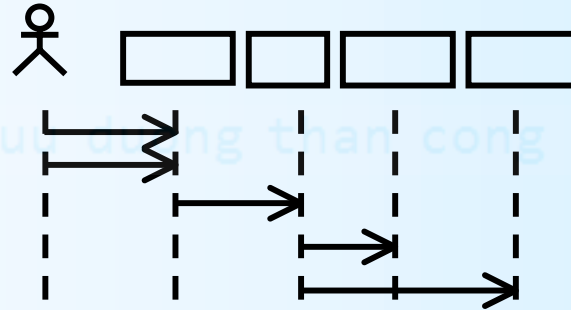
Design Model



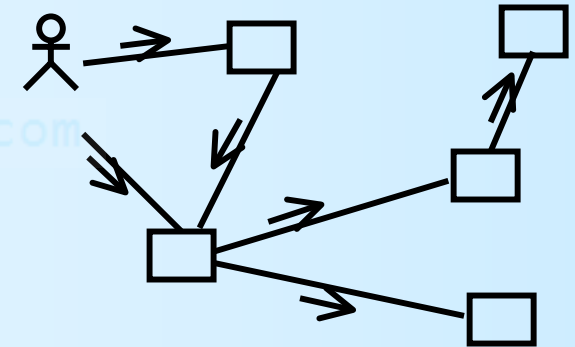
Use-Case Realization



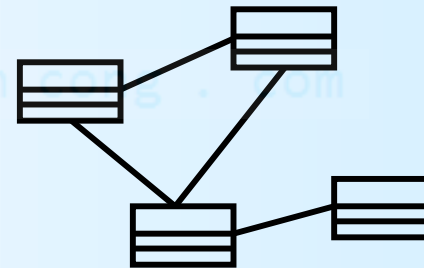
Use Case



Sequence Diagrams



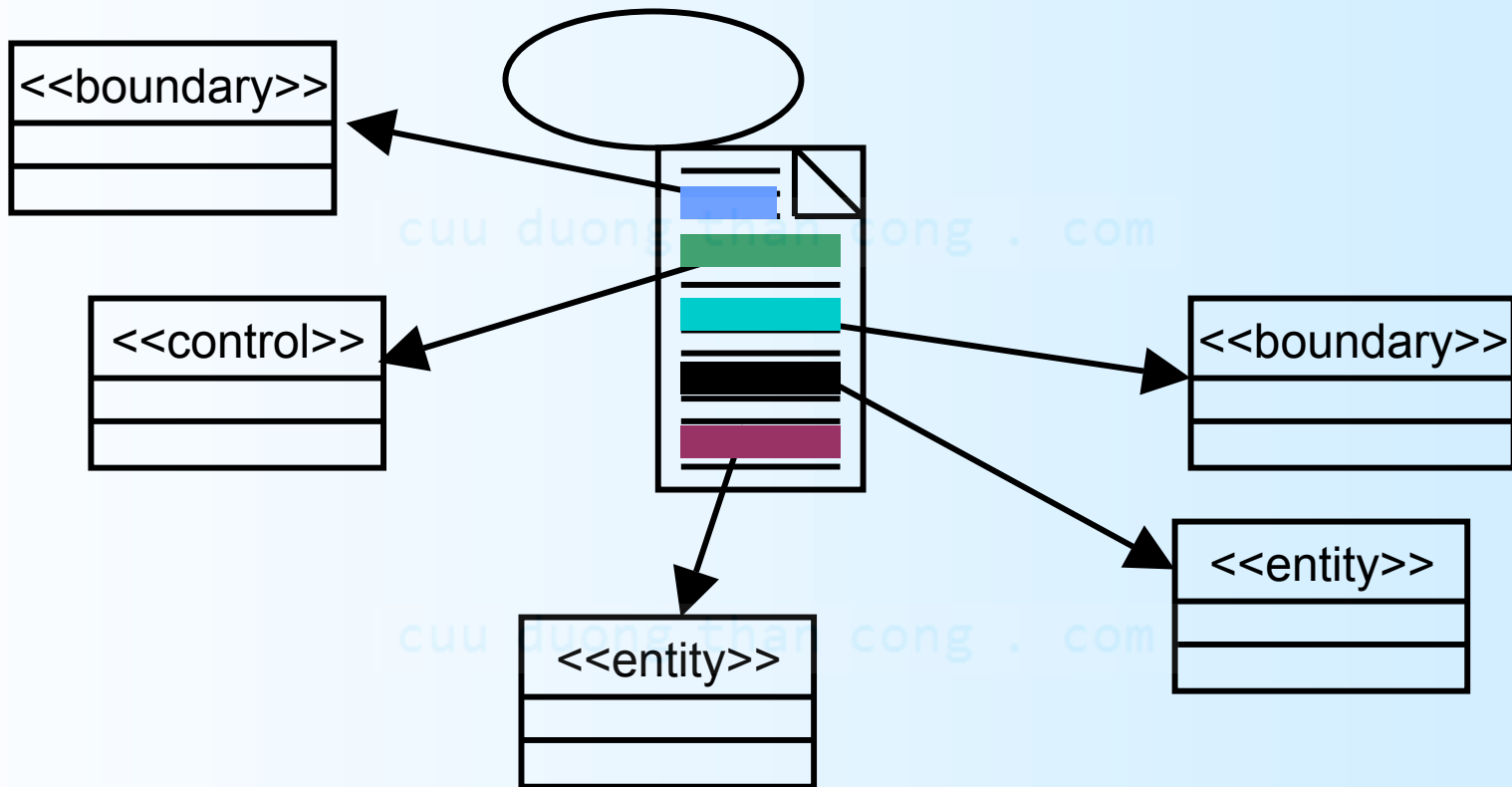
Collaboration Diagrams



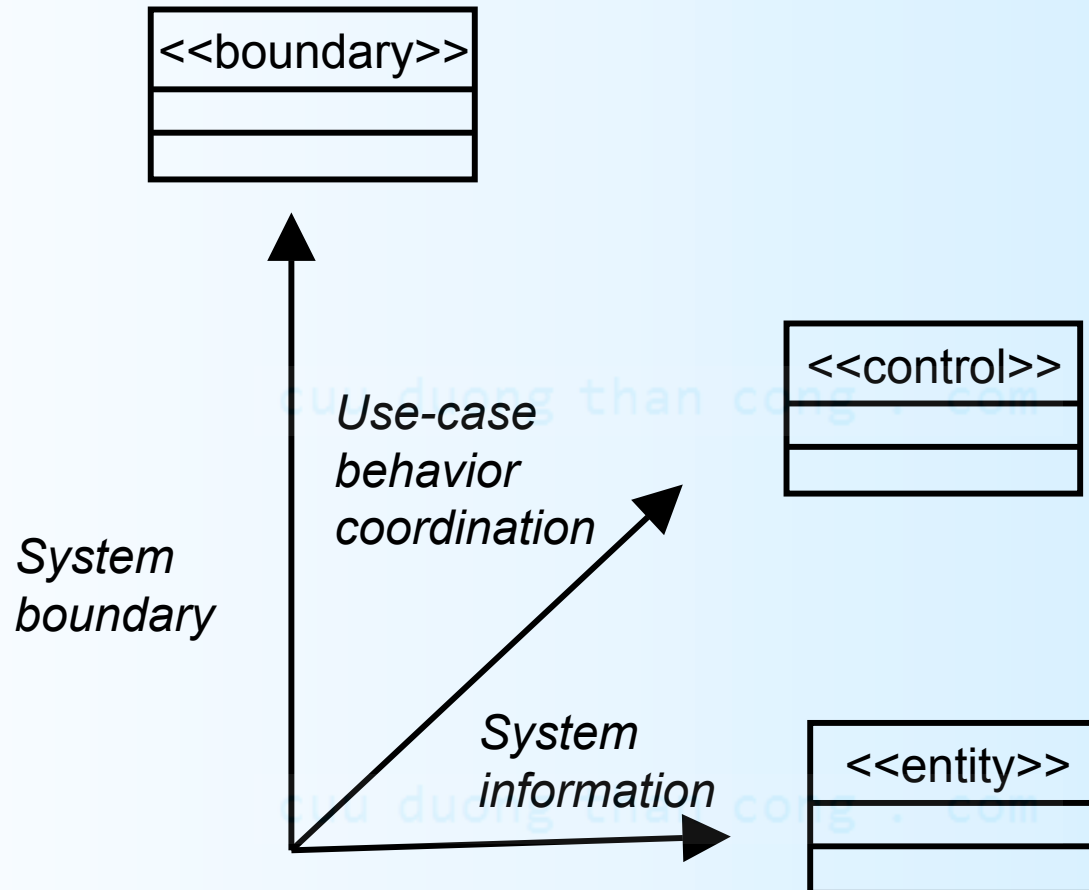
Class Diagrams

Tìm kiếm các Class từ Use-Case Behavior

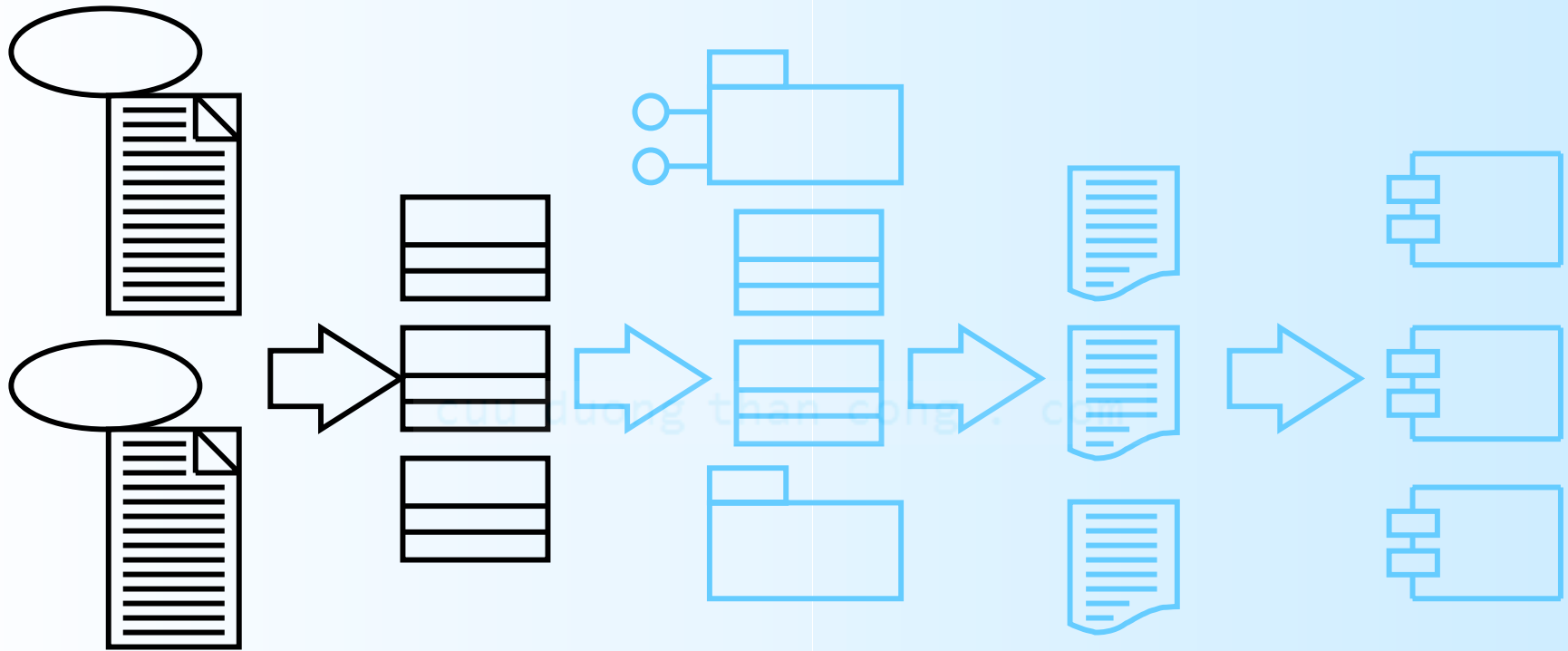
- Toàn bộ hành vi của một use case phải được phân bổ về cho các analysis class



Thế nào là một Analysis Class?



Các Analysis Class: Bước đầu tiên đến cài đặt



Use Cases

**Analysis
Classes**

**Design
Elements**

**Source
Code**

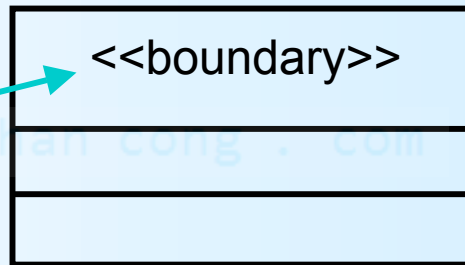
Exec

Use-Case Analysis

Thế nào là Boundary Class?

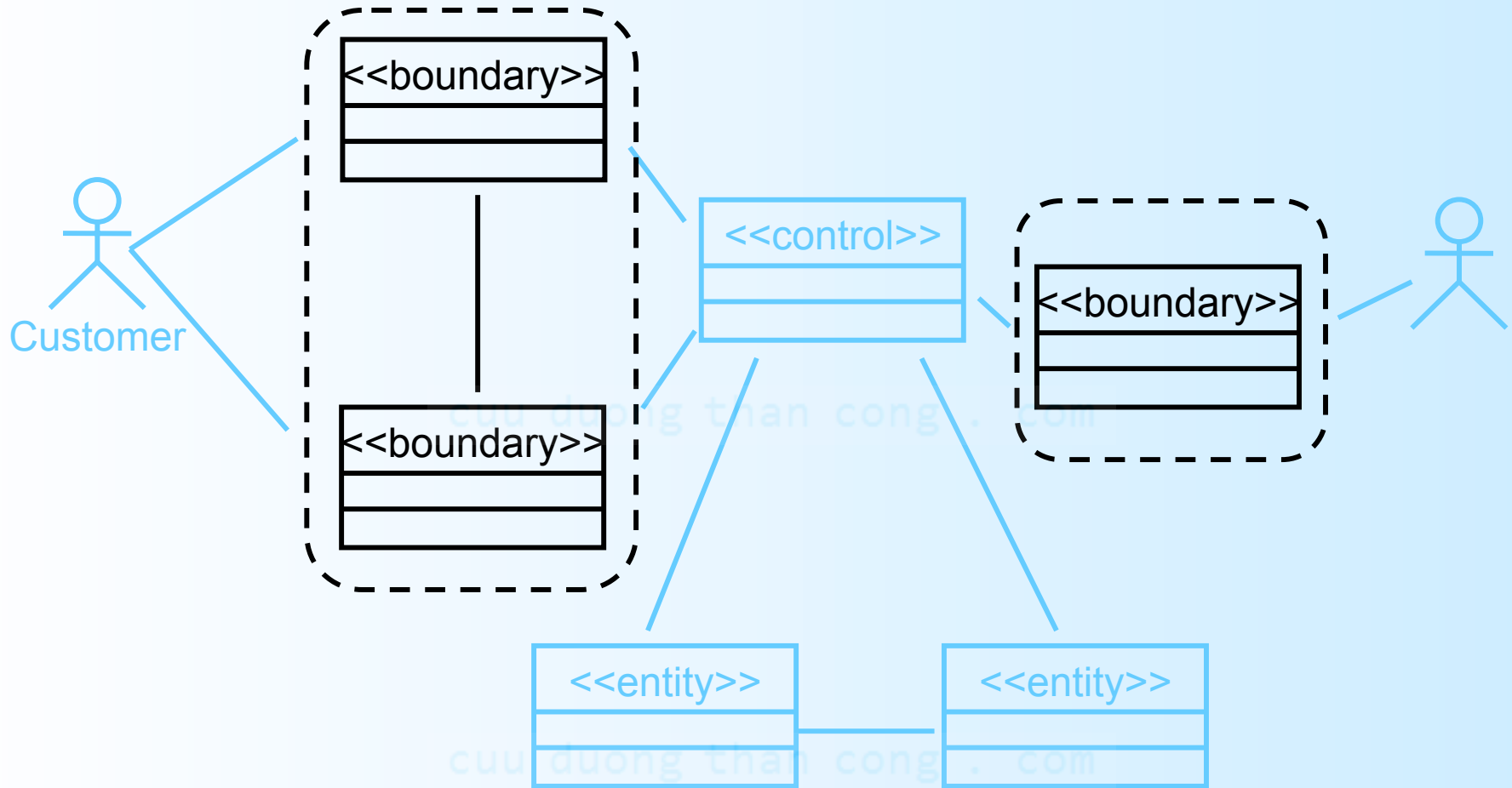
- Làm trung gian giao tiếp với những gì nằm ngoài hệ thống
- Một số kiểu
 - Các User interface class
 - Các System interface class
 - Các Device interface class
- *Một boundary class cho 1 cặp actor/use case*

*Analysis class
stereotype*



Phụ thuộc môi trường

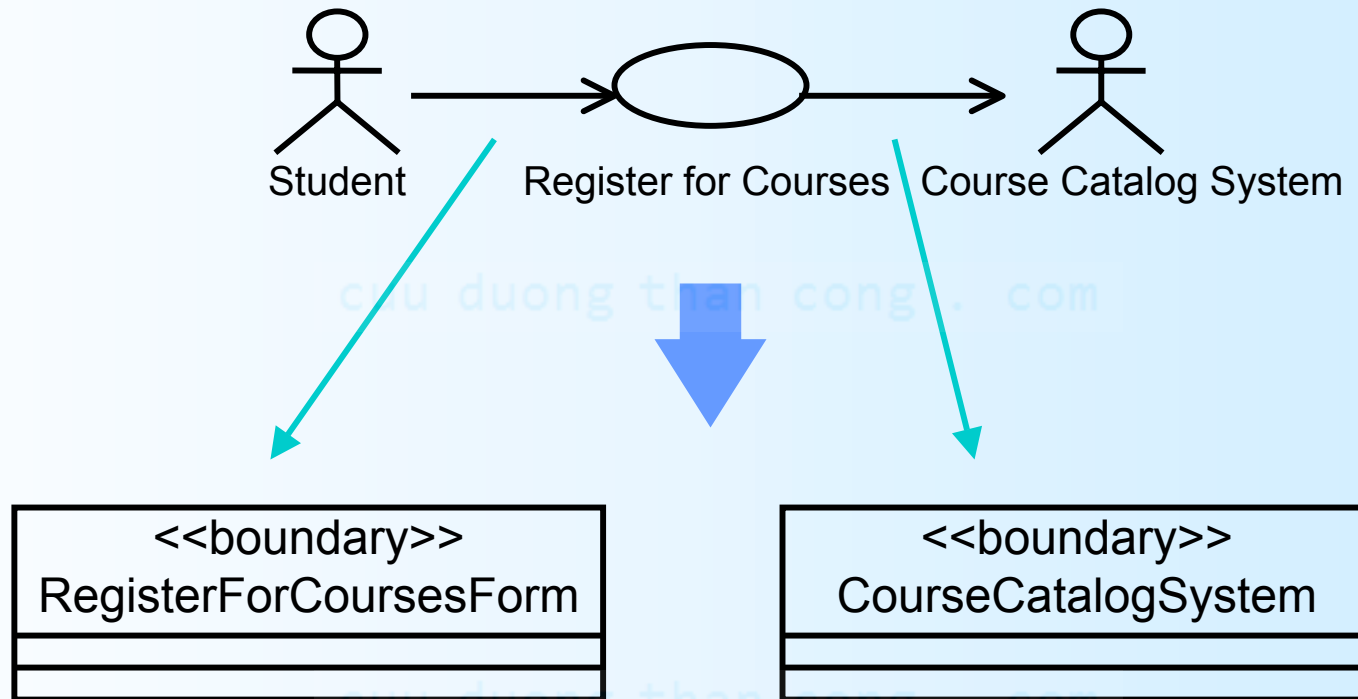
Vai trò của một Boundary Class



Mô hình hóa sự tương tác giữa system và môi trường của nó

Ví dụ: Tìm kiếm các Boundary Class

- Một boundary class cho 1 cặp actor/use case



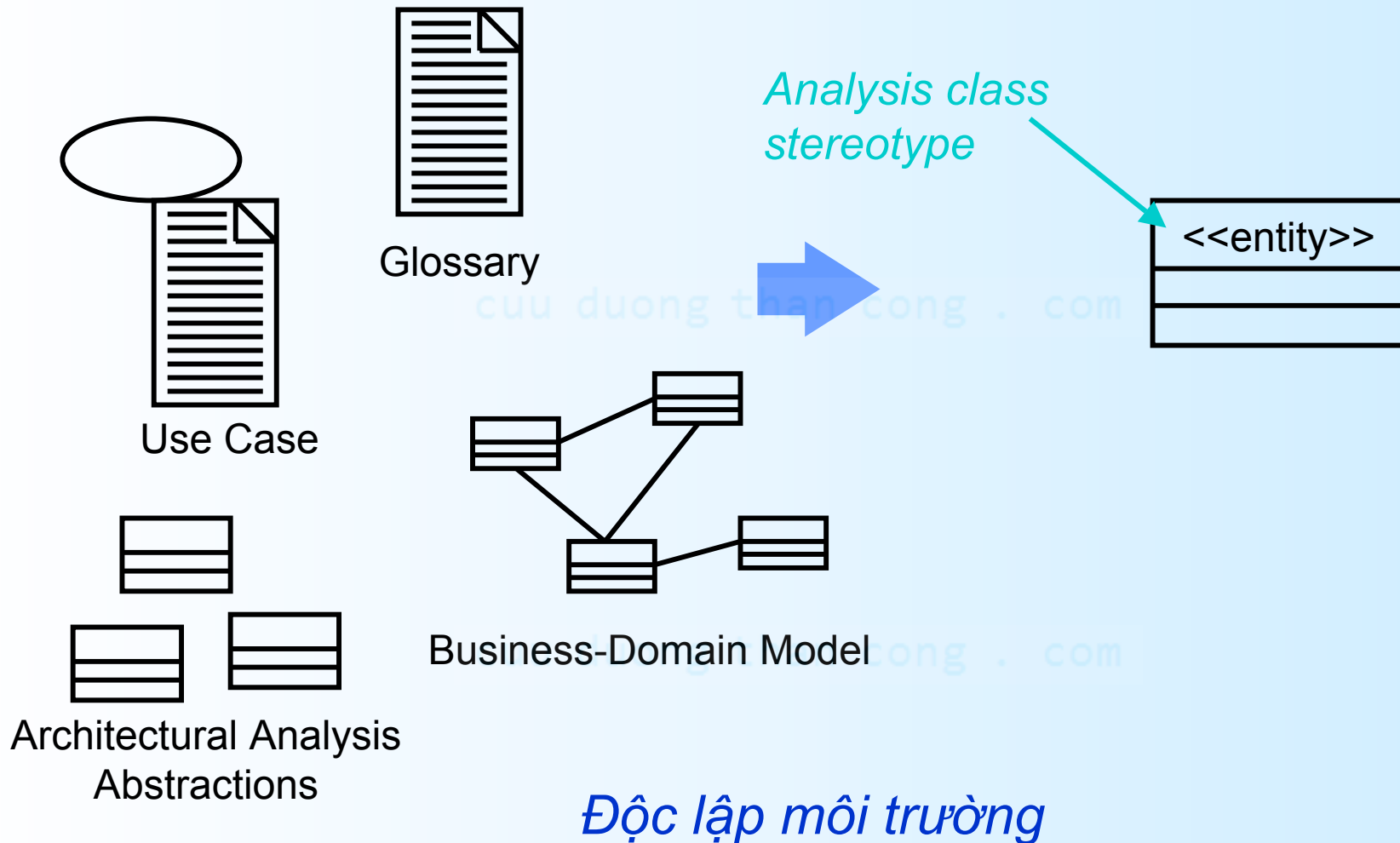
Guidelines: Boundary Class

- Các User Interface Class
 - Tập trung vào những thông tin gì được thể hiện cho người dùng
 - **KHÔNG** tập trung vào các chi tiết UI
- Các System và Device Interface Class
 - Tập trung vào những protocols nào phải đ/n
 - **KHÔNG** tập trung vào cách mà các protocol sẽ được cài đặt

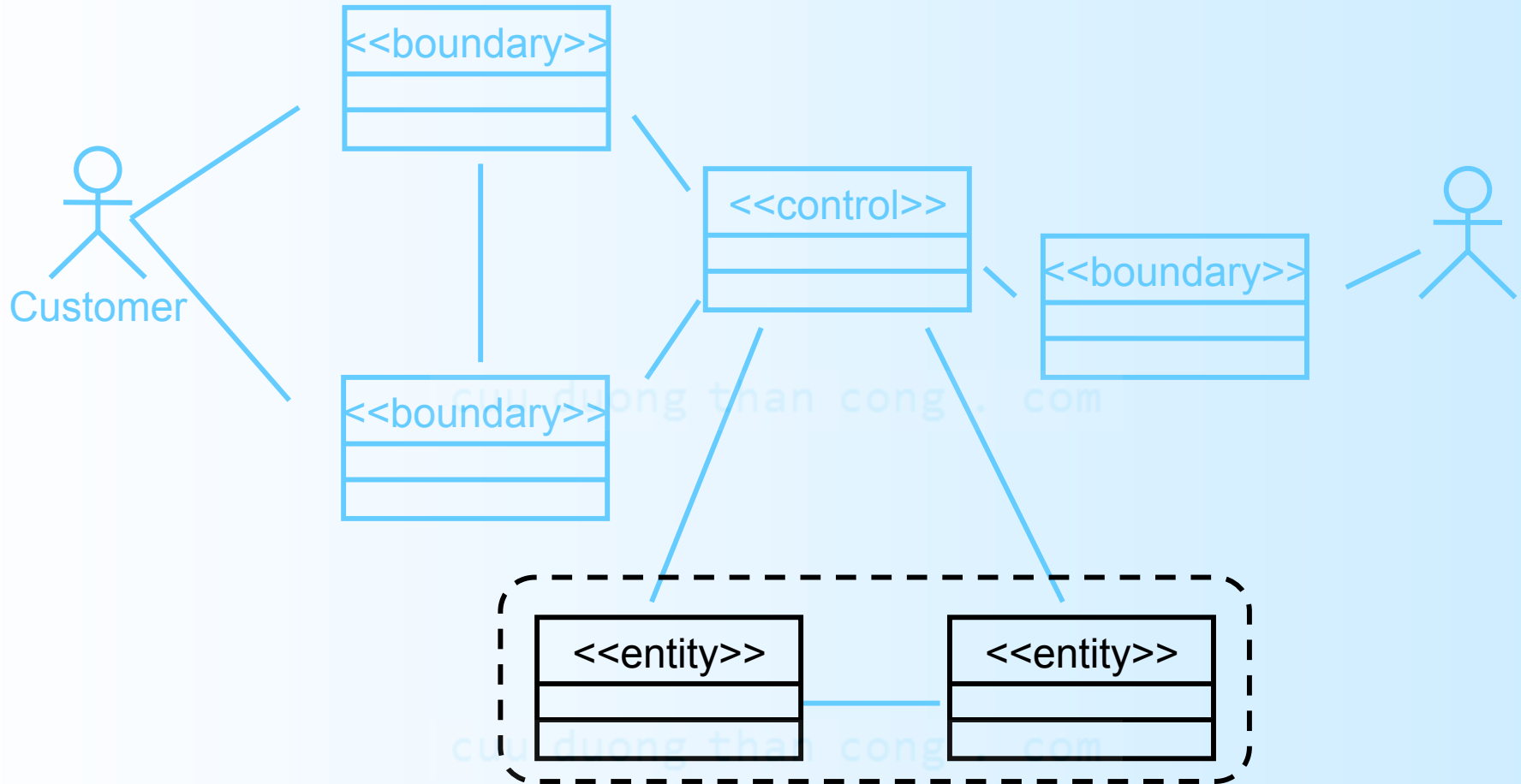
Tập trung vào các nhiệm vụ, chứ không phải chi tiết!

Thế nào là một Entity Class?

- Các trừu tượng hóa then chốt của system



Vai trò của một Entity Class



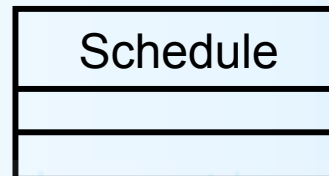
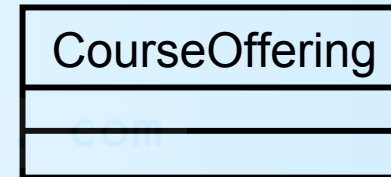
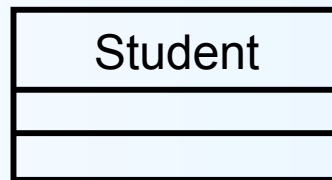
Lưu trữ và quản trị các thông tin trong system

Ví dụ: Tìm kiếm các Entity Class

- Dùng use-case flow of events như input
- Các trừu tượng hóa then chốt của use case
- Hướng tiếp cận truyền thống (nouns filtering)
 - Gạch dưới các cụm danh từ trong flow of events
 - Loại bỏ các ứng viên dư thừa
 - Loại bỏ các ứng viên mơ hồ, không rõ ràng
 - Loại bỏ các actor (ngoài phạm vi)
 - Loại bỏ các kiến trúc cài đặt
 - Loại bỏ các attribute (để lại dùng sau)
 - Loại bỏ các operation

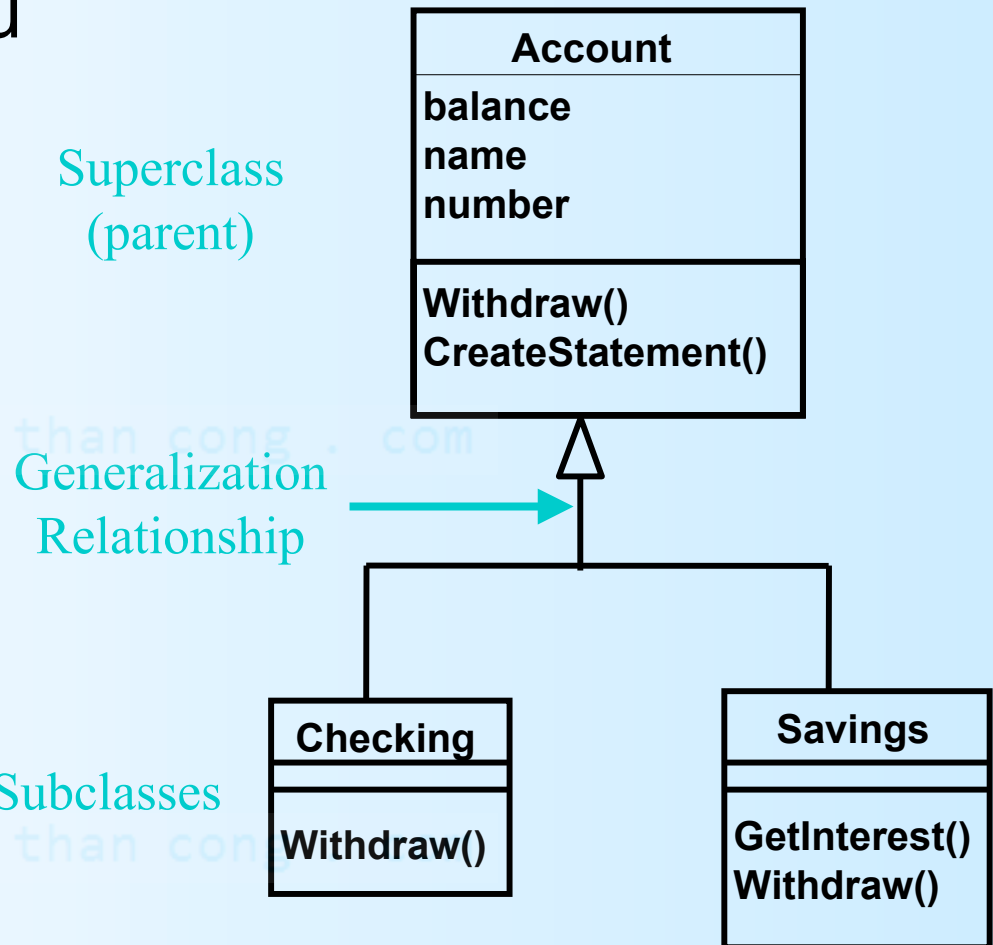
Ví dụ: Chọn ứng viên cho các Entity Class

- Đăng ký học phần (Lập lịch)

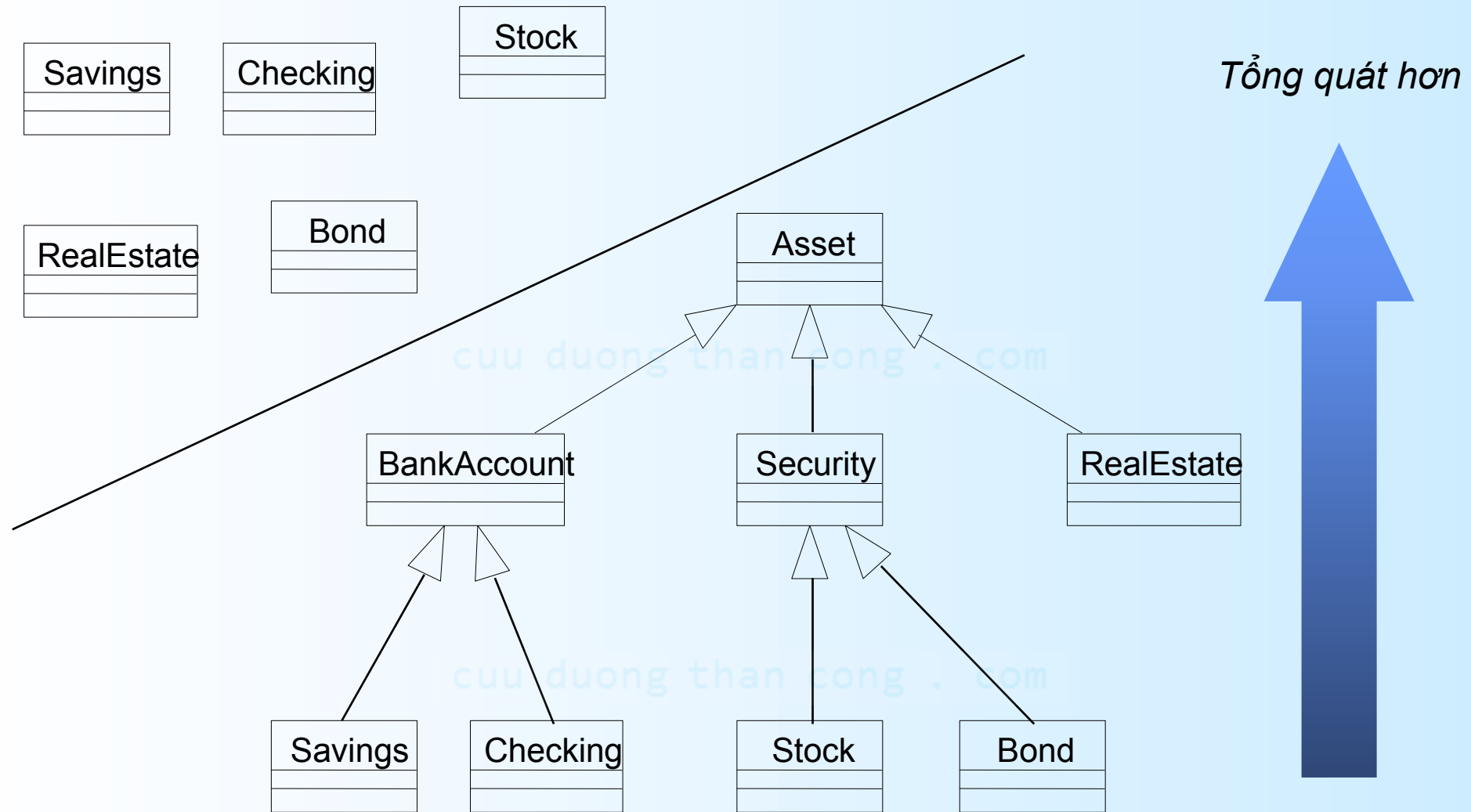


Review: Generalization

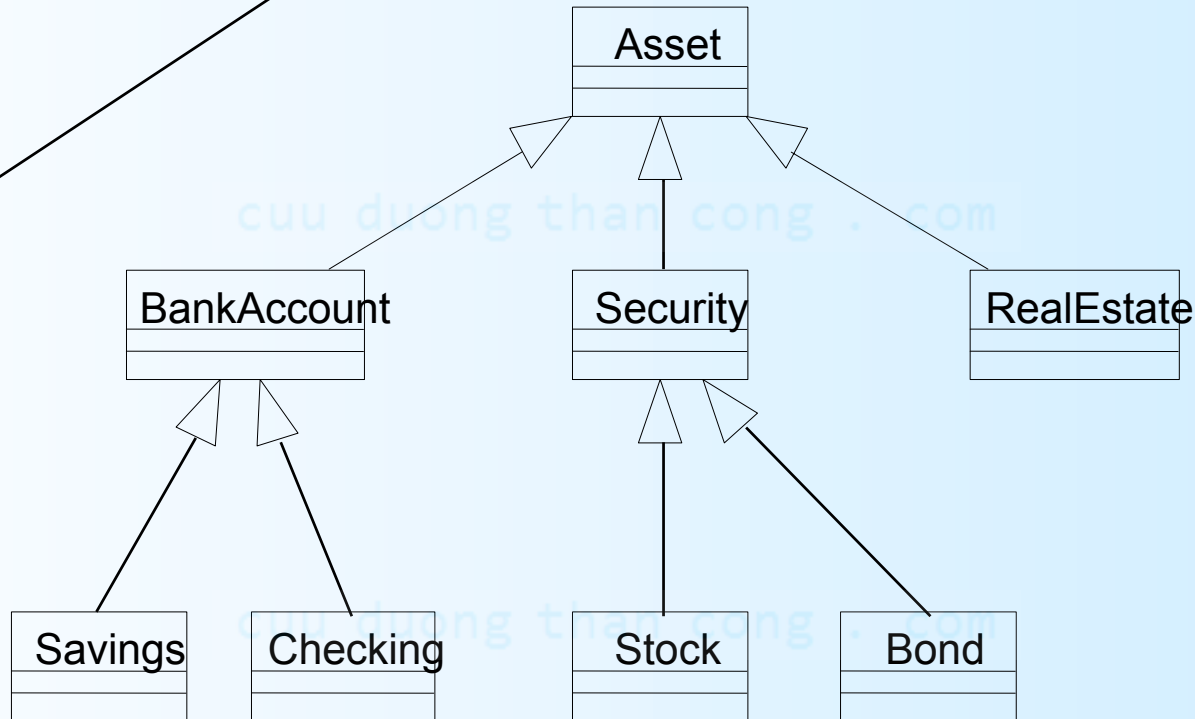
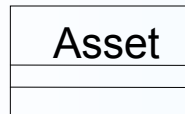
- Một class chia sẻ cấu trúc và/hoặc hành vi của một hay nhiều class
- Mỗi quan hệ “Là một dạng của”
- Trong phân tích, sử dụng ở mức độ đơn giản, sơ sài



Tìm Generalization: Generalization của các Class



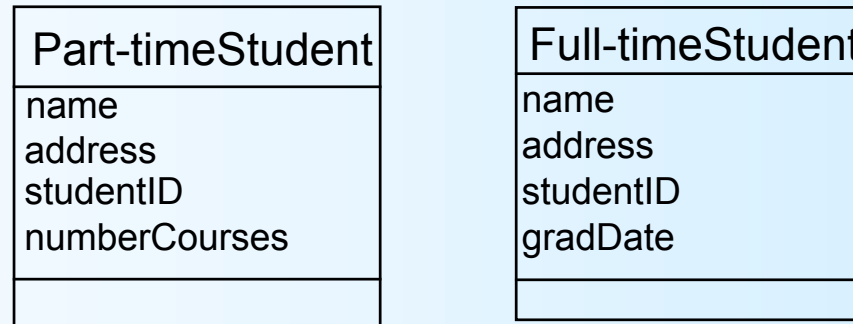
Tìm Generalization: Specialization của các Class



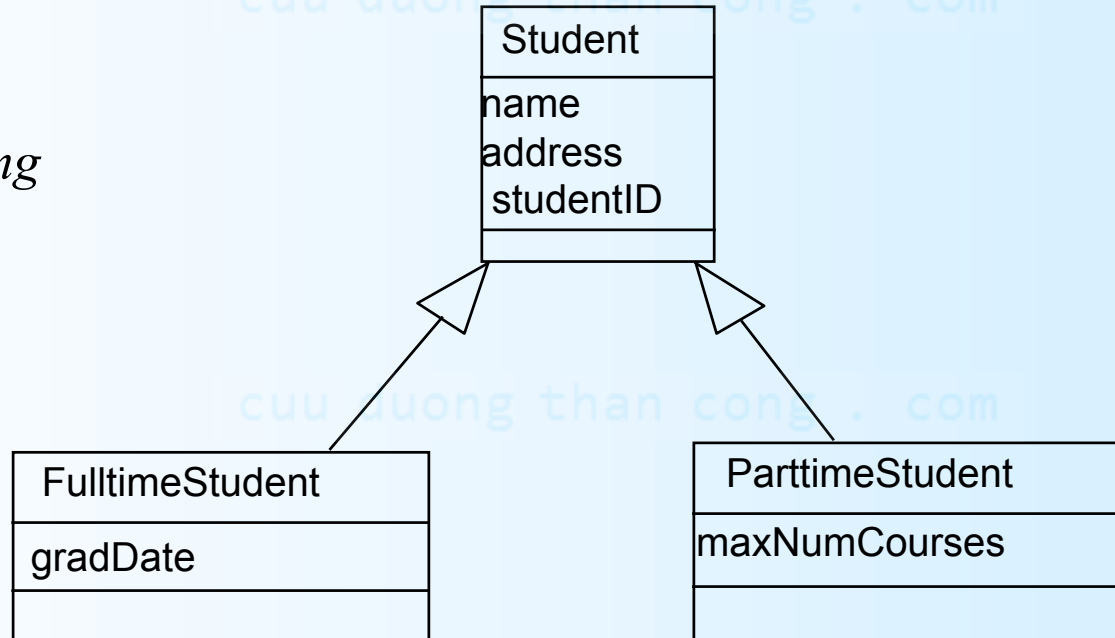
Chuyên biệt hơn

Ví dụ: Generalization (Chia sẻ ngữ nghĩa)

*Không có sự
tổng quát hóa*

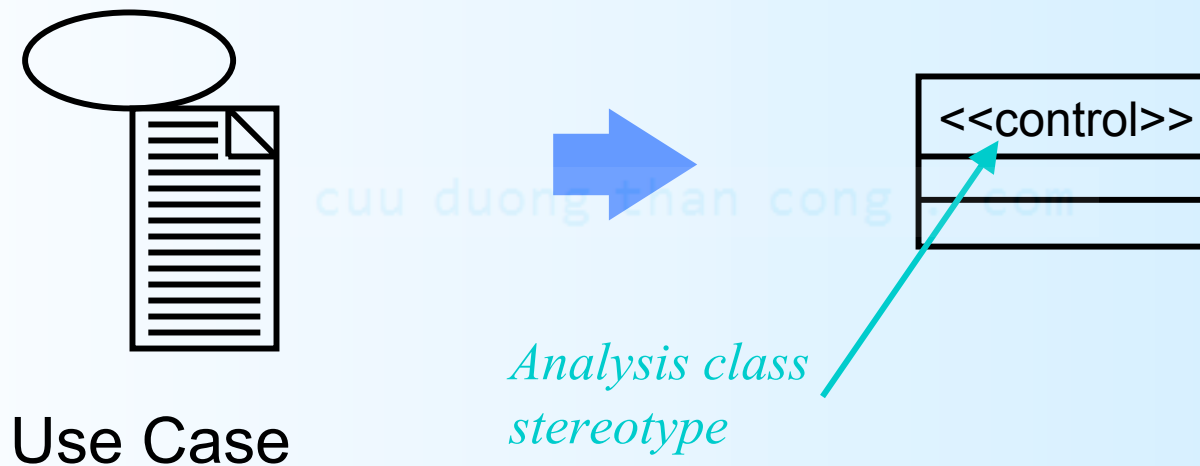


*Có sự tổng
quát hóa*



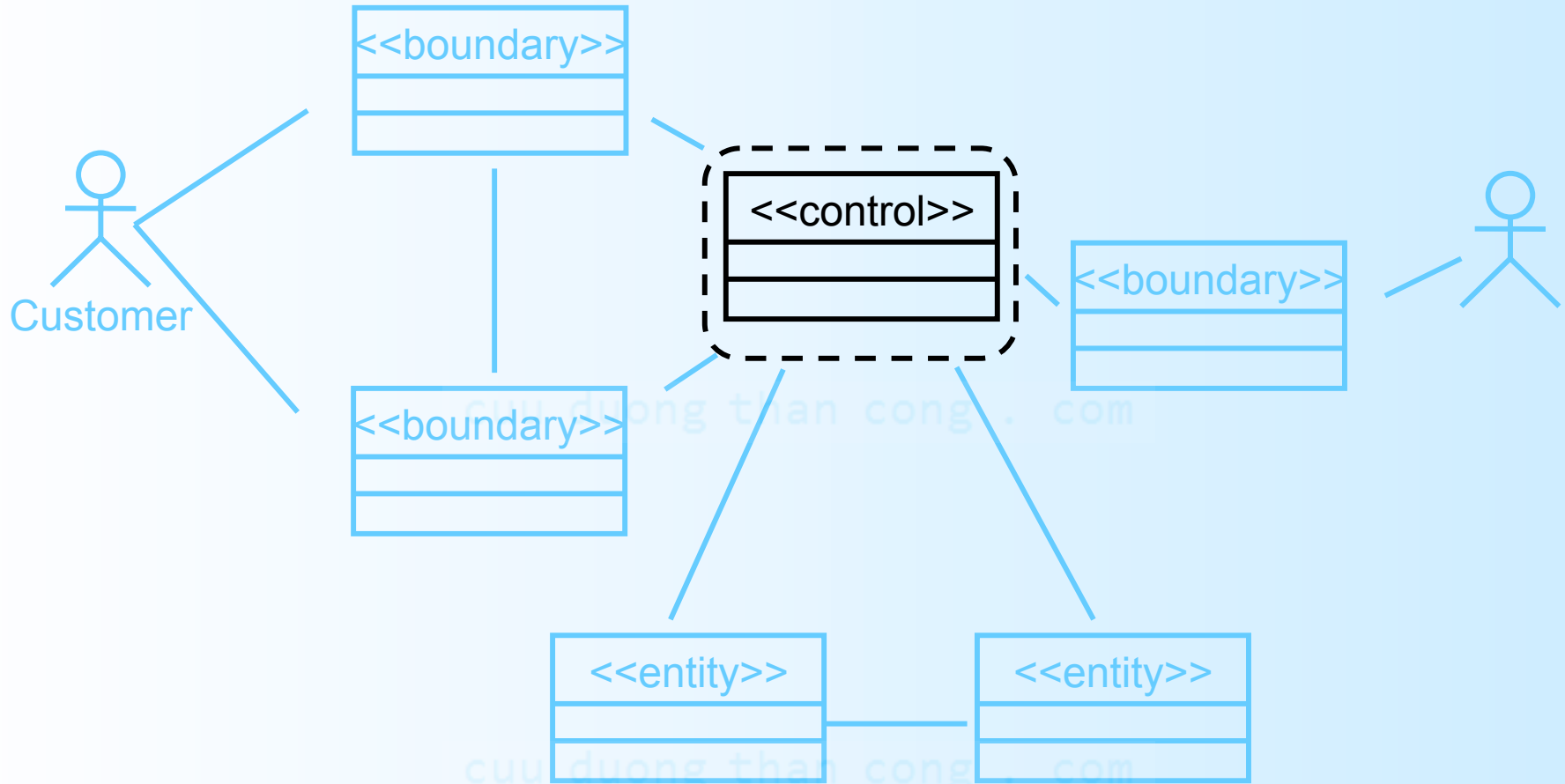
Thế nào là một Control Class?

- Nhà điều phối các hành vi của Use-case
- *Chỉ một control class cho một use case*



Phụ thuộc use-case, độc lập môi trường

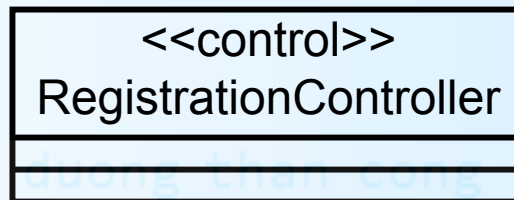
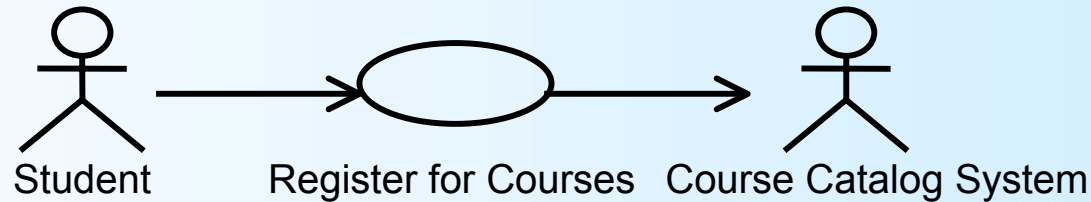
Vai trò của một Control Class



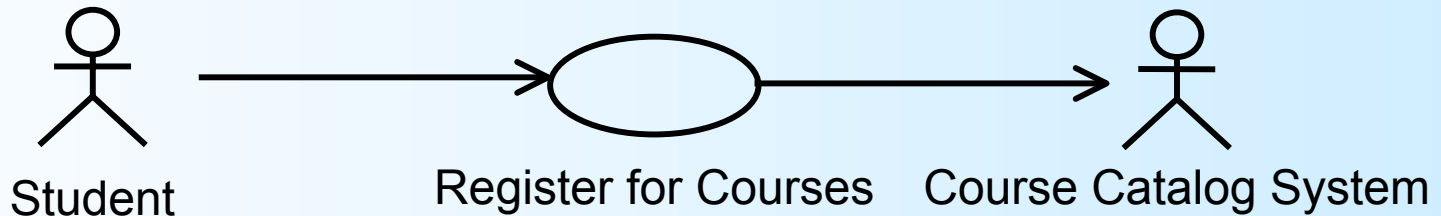
Điều phối các hành vi của use-case

Ví dụ: Tìm các Control Class

- Một control class cho một use case

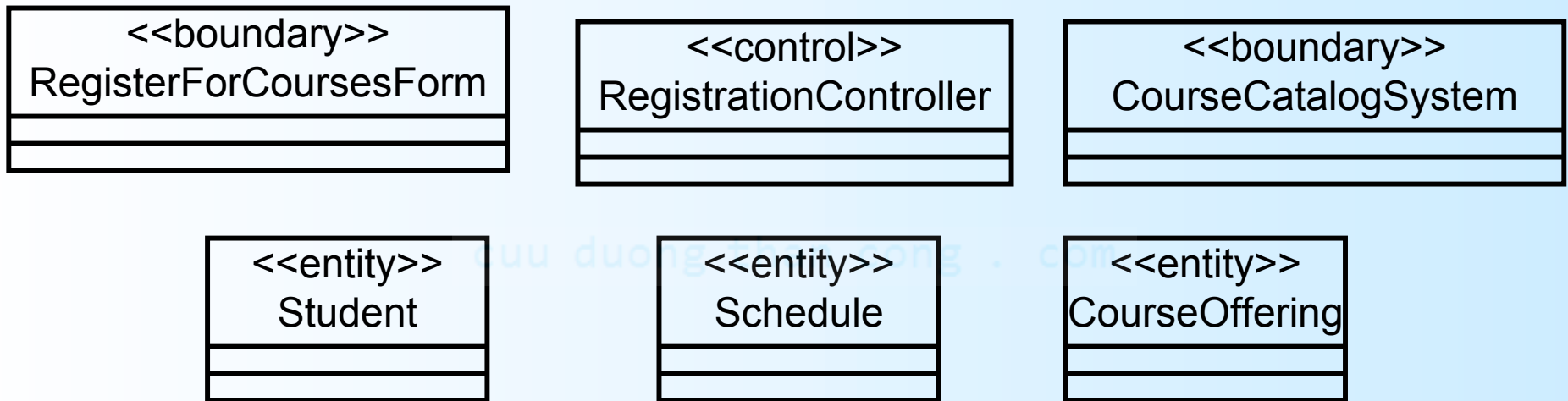


Ví dụ: Tổng kết về các Analysis Class



Use-Case Model

Design Model

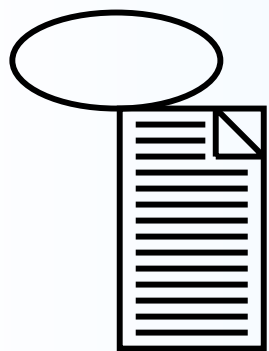


Các bước thực hiện phân tích Use-Case

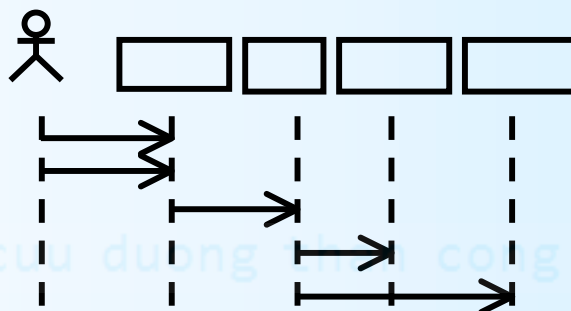
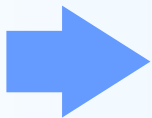
- Bổ sung các mô tả của Use-Case
- Đ/v mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
- ★ – Phân bổ Use-Case Behavior cho các Class
- Đ/v mỗi analysis class tìm được
 - Mô tả các nhiệm vụ của chúng
 - Mô tả các Attribute và Association
 - Lượng giá các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

Distribute Use-Case Behavior to Classes

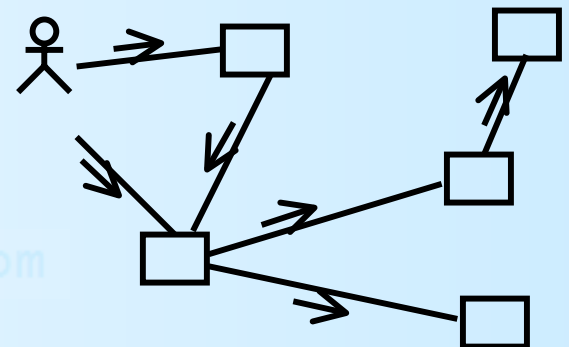
- Đ/v mỗi use-case flow of events:
 - Xác định các analysis class
 - Gắn kết các trách nhiệm của use-case cho các analysis class
 - Mô hình hóa tương tác của các analysis class trong interaction diagram



Use Case



Sequence Diagrams



Collaboration Diagrams

Use-Case Realization

Guidelines: Allocating Responsibilities to Classes

- Dùng các A.Class stereotype làm guide:
 - Các Boundary Class
 - Các hành vi liên quan đến sự giao tiếp với actor
 - Các Entity Class
 - Các hành vi liên quan đến dữ liệu được gói trong một abstraction
 - Các Control Class
 - Các hành vi đặc thù cho một use case hoặc một phần rất quan trọng của flow of events

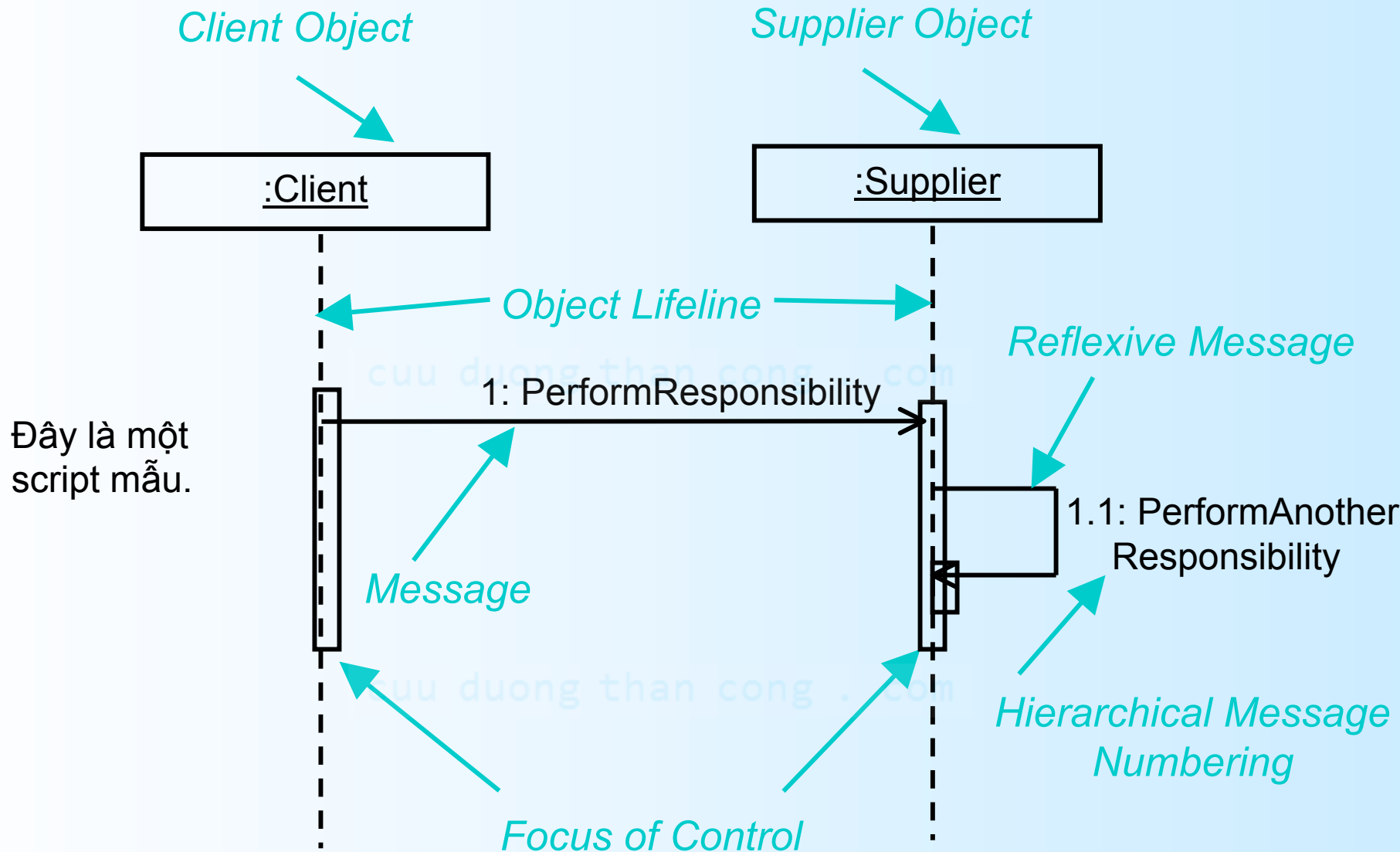
cuuduongthancong.com

(continued)

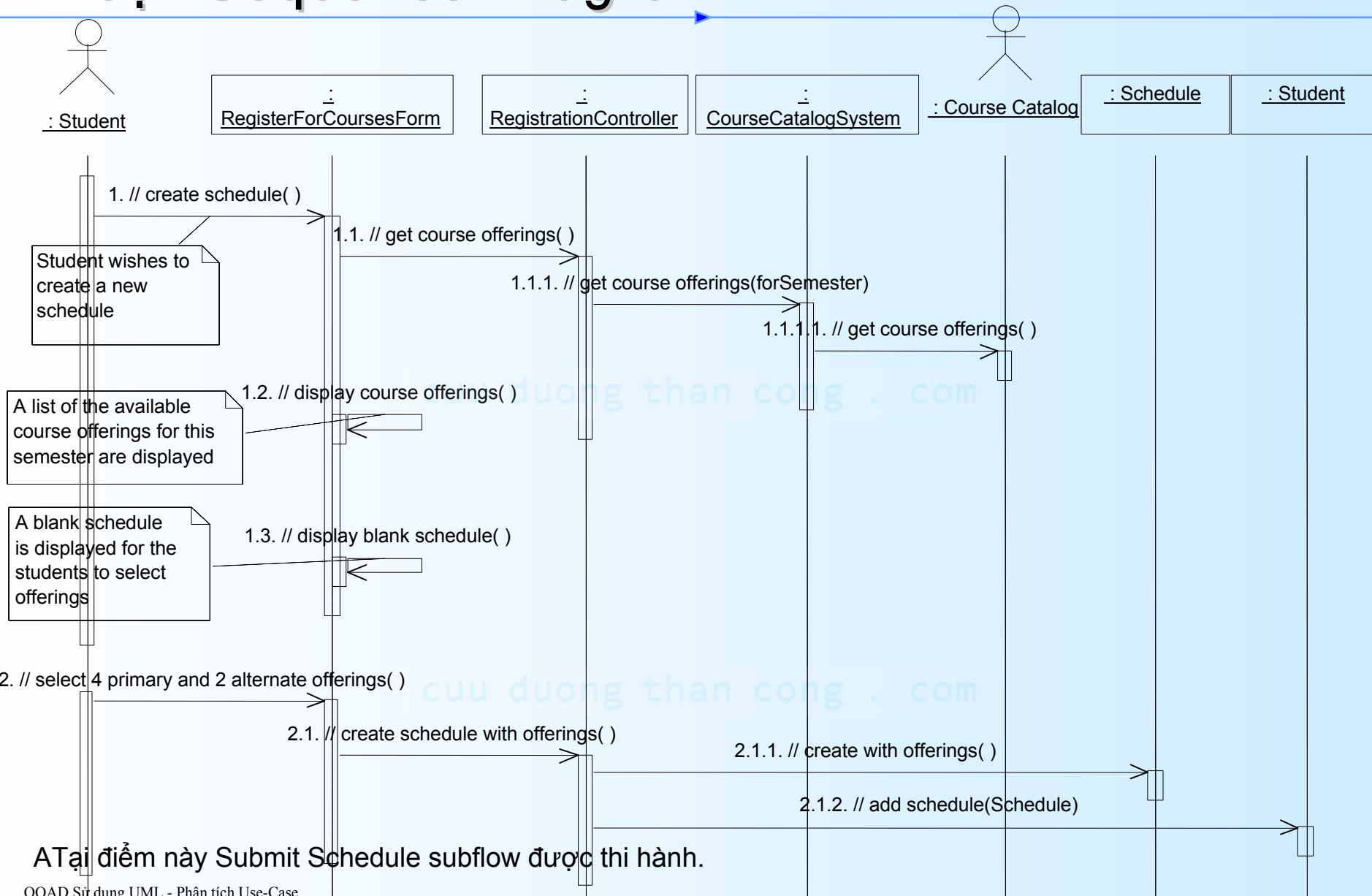
Guidelines: Allocating Responsibilities to Classes (cont.)

- Ai có dữ liệu cần cho việc thực hiện nhiệm vụ?
 - Một class có dữ liệu, hãy để n/v cùng với di
 - Nhiều class có dữ liệu :
 - Hãy để n/v trong 1 class và thêm quan hệ với các class khác.
 - Tạo một class mới, để n/v trong class mới này, và thêm quan hệ với các class cũ
 - Hãy để n/v trong control class, và thêm quan hệ với các class cần để thực hiện n/v

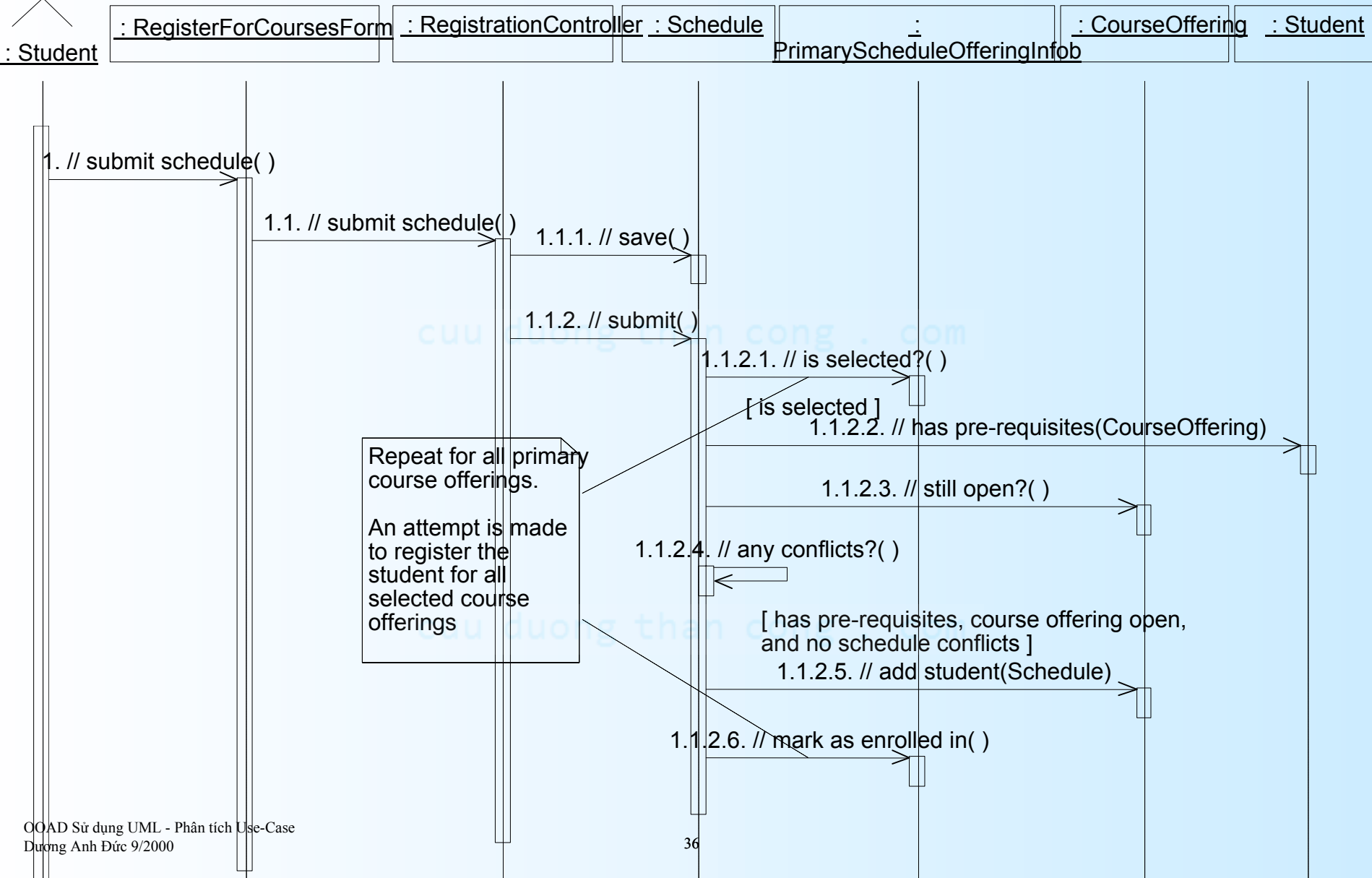
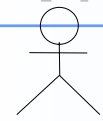
The Anatomy of Sequence Diagrams



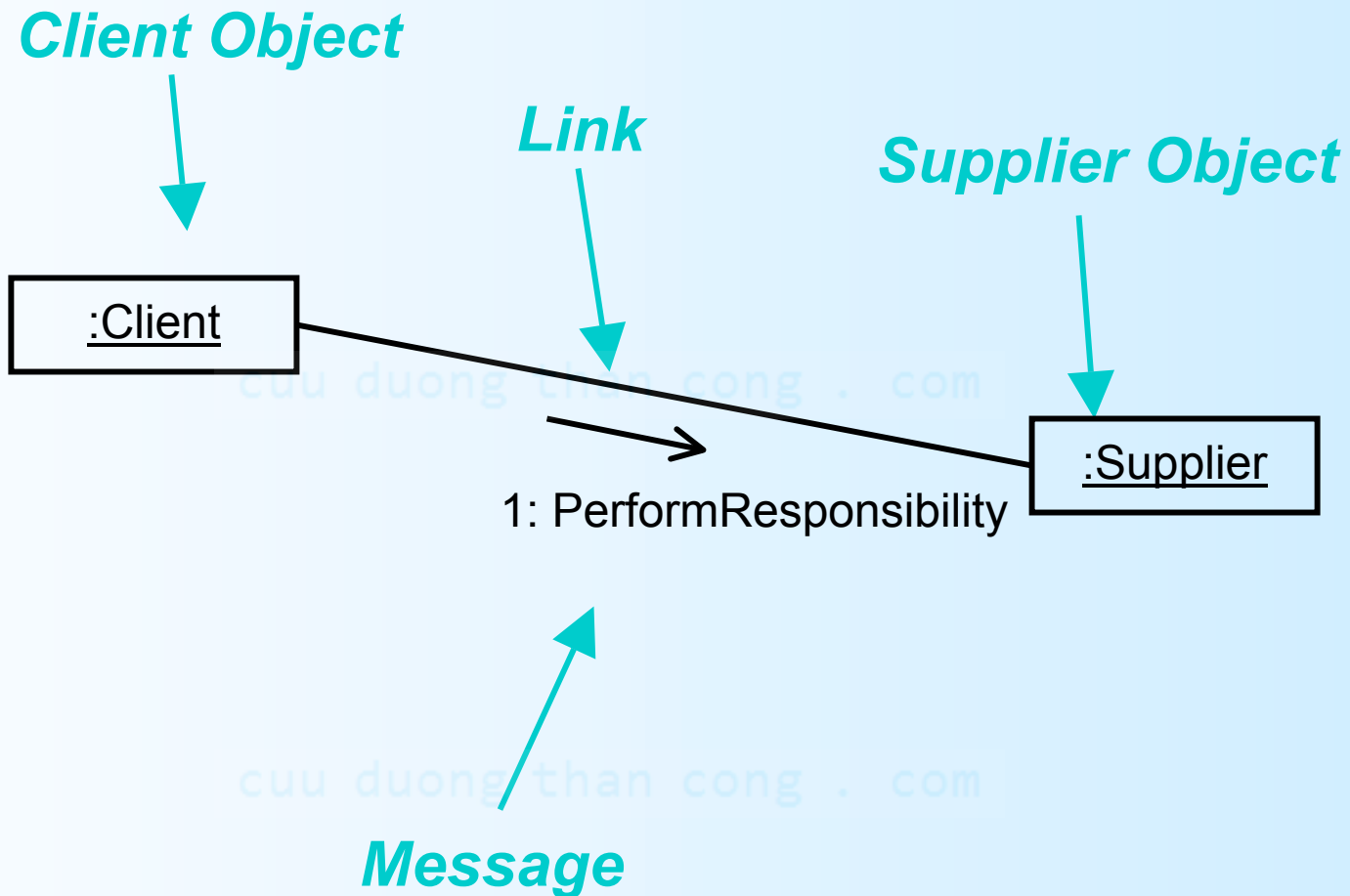
Ví dụ: Sequence Diagram



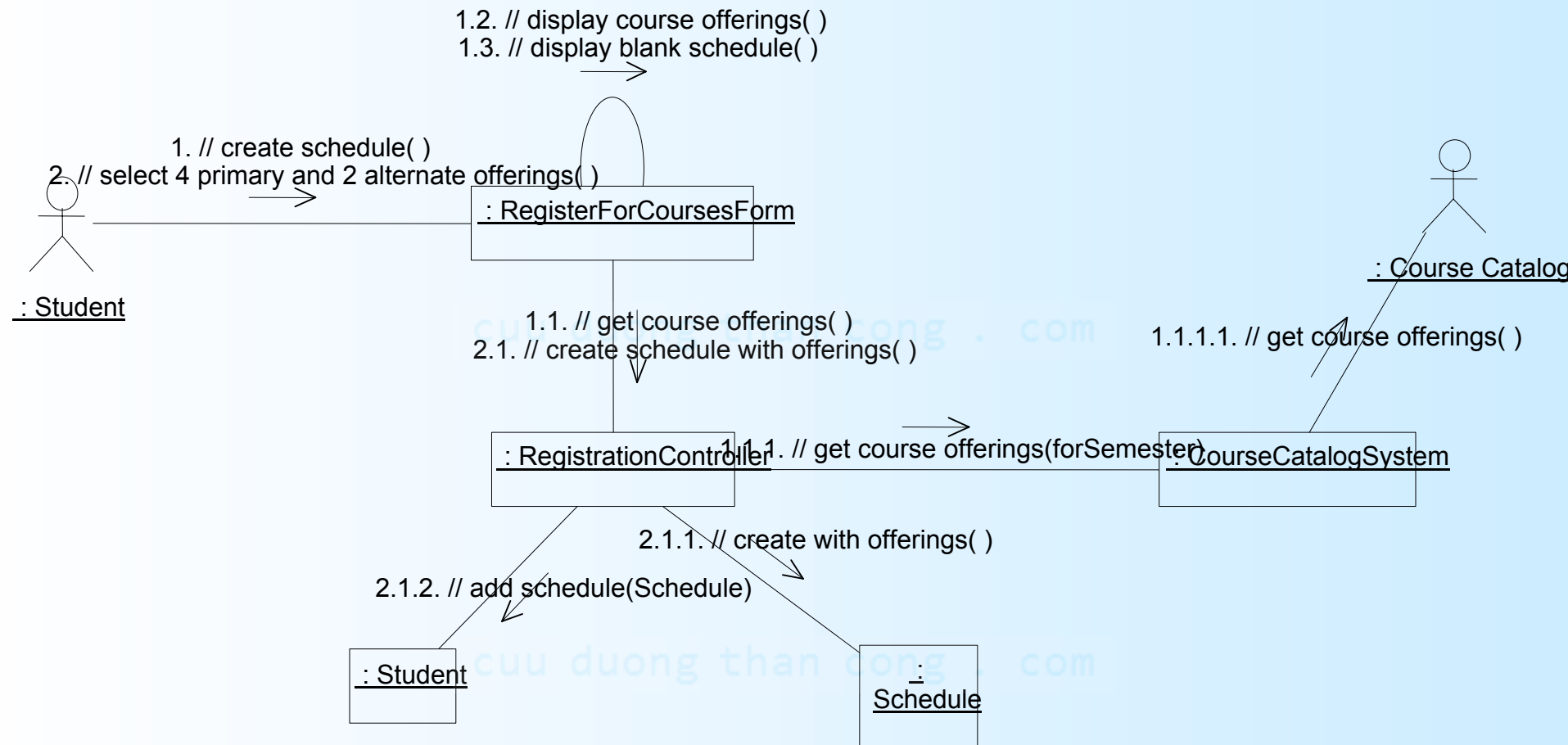
Ví dụ: Sequence Diagram (cont.)



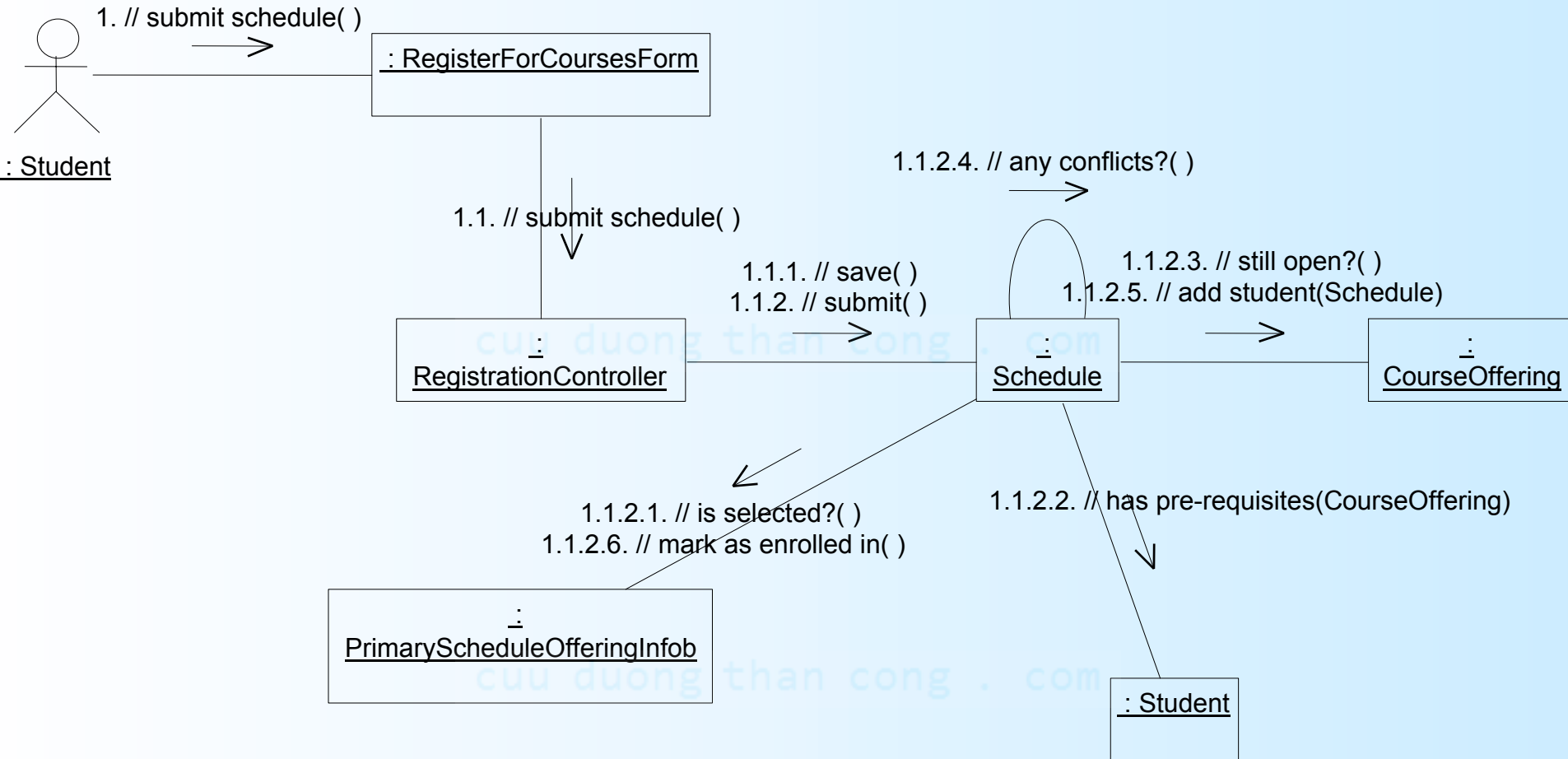
The Anatomy of Collaboration Diagrams



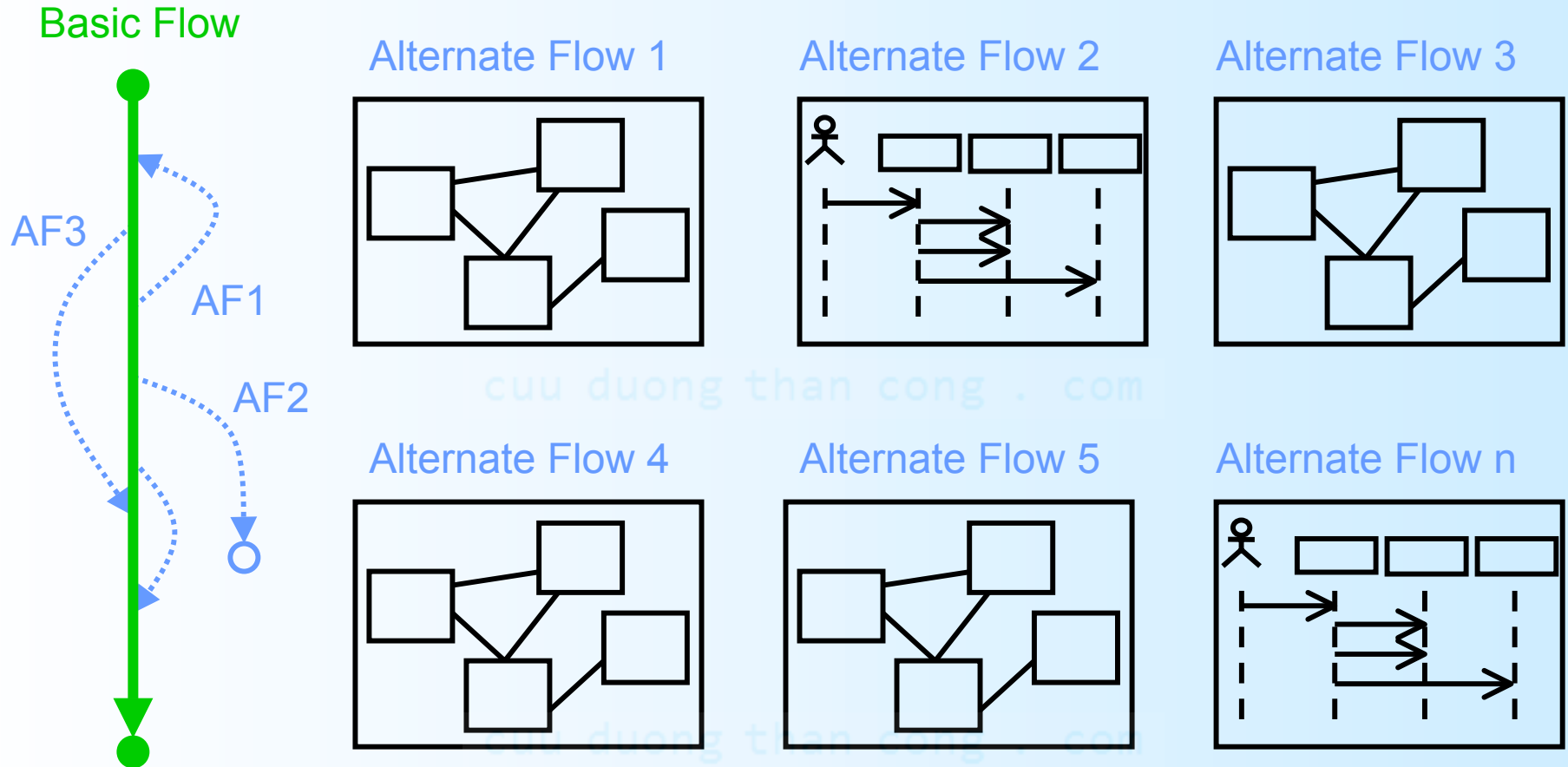
Ví dụ: Collaboration Diagram



Ví dụ: Collaboration Diagram (cont.)



One Interaction Diagram Not Good Enough



Collaboration Diagrams Vs Sequence Diagrams

- Collaboration Diagrams

- Chỉ ra các mối quan hệ bổ sung cho các tương tác
- Trực quan hóa tốt hơn các mẫu cộng tác
- Trực quan hóa tốt hơn các hiệu ứng tác động lên một đối tượng cụ
- Dễ sử dụng hơn trong các v/d cần g/q tập thể

- Sequence Diagrams

- Chỉ ra rõ ràng chuỗi các thông điệp
- Trực quan hóa tốt hơn toàn bộ luồng sự kiện
- Tốt hơn cho các đặc tả real-time và cho các scenario phức tạp

Exercise: Use-Case Analysis, Part 1

- Hãy xây dựng:
 - Use-Case Model, đặc biệt là các use-case flows of events
 - Các trừu tượng hóa/class then chốt

cuu duong than cong . com

cuu duong than cong . com

(còn tiếp)

Exercise: Use-Case Analysis, Part 1 (cont.)

- Hãy xác định các thông tin sau cho một use case cụ thể:
 - Các analysis class, cùng với chúng là:
 - Brief descriptions
 - Stereotypes
 - Responsibilities
 - Các collaboration cần thiết để cài đặt use case

(còn tiếp)

Exercise: Use-Case Analysis, Part 1 (cont.)

- Với một use case cụ thể, hãy thiết lập:
 - Use-case realization interaction diagram cho tối thiểu một trong các use-case flows of events

cuu duong than cong . com

cuu duong than cong . com

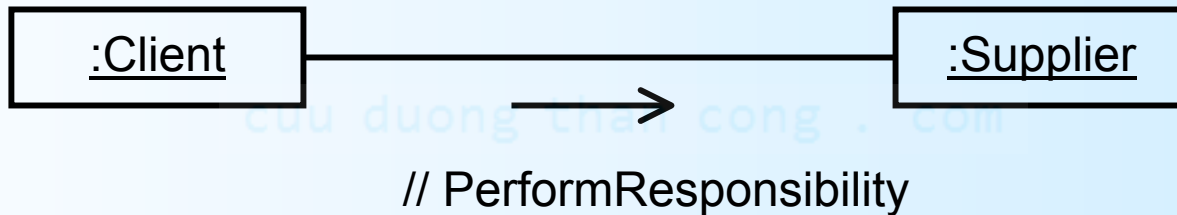
Use-Case Analysis Steps

- Bổ sung các mô tả của Use-Case
- Đ/v mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior cho các Class
- Đ/v mỗi analysis class tìm được
 - ★ – Mô tả các nhiệm vụ của chúng
 - Mô tả các Attribute và Association
 - Lượng giá các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

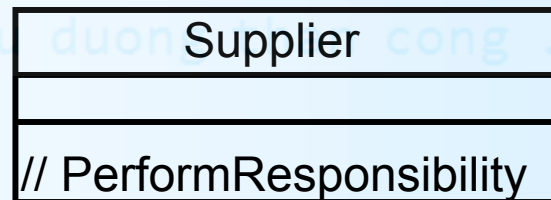
Describe Responsibilities

- Trách nhiệm (responsibilities) là gì?
- Làm thế nào để tìm ra chúng?

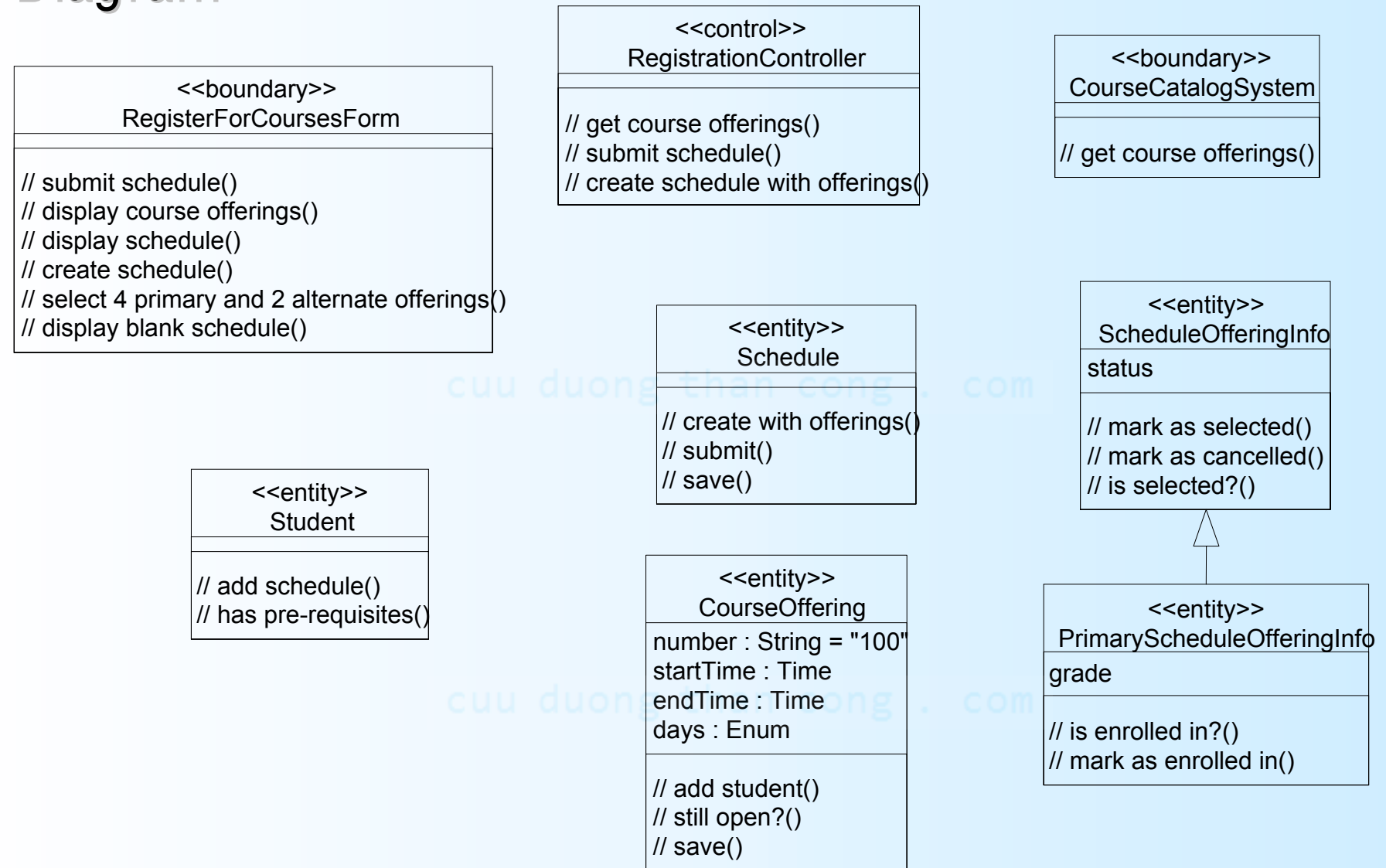
Interaction Diagram



Class Diagram



Ví dụ: View of Participating Classes (VOPC) Class Diagram



Maintaining Consistency: What to Look For

- In order of criticality
 - Redundant responsibilities across classes
 - Disjoint responsibilities within classes
 - Class with one responsibility
 - Class with no responsibilities
 - Better distribution of behavior
 - Class that interacts with many other classes

Use-Case Analysis Steps

- Bổ sung các mô tả của Use-Case
- Đ/v mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior cho các Class
- Đ/v mỗi analysis class tìm được
 - Mô tả các nhiệm vụ của chúng
 - ★ – Mô tả các Attribute và Association
 - Lượng giá các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

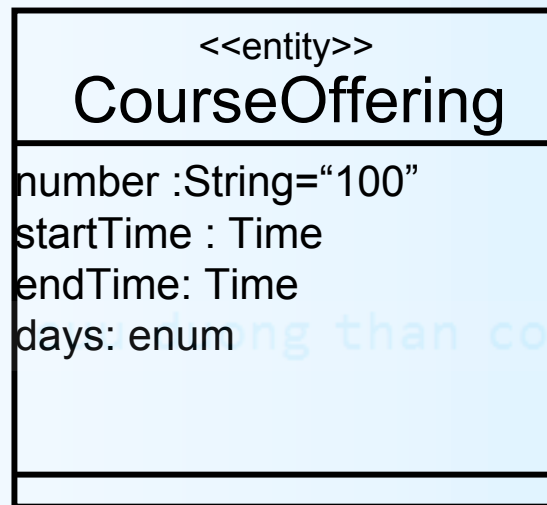
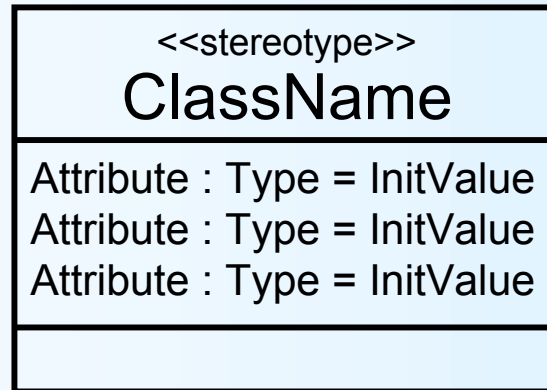
Describe Attributes and Associations

- Định nghĩa các Attribute
- Thiết lập các mối quan hệ dạng Aggregation và Association

cuu duong than cong . com

cuu duong than cong . com

Review: What is an Attribute?



attribute



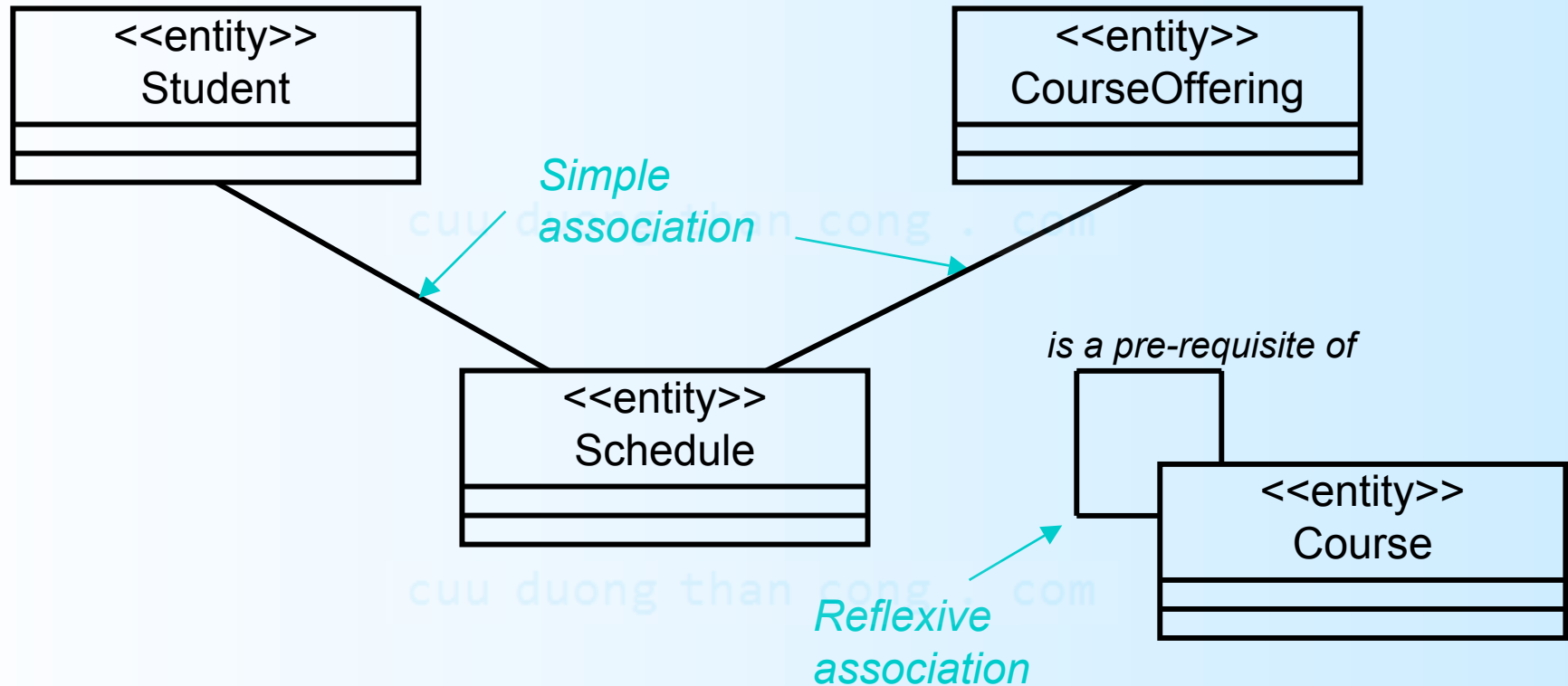
*Trong analysis, không nên
tốn nhiều thời gian cho
việc xác định các attribute
signature*

Finding Attributes

- Các thuộc tính/đặc điểm của các class
- Các thông tin đdc giữ lại bởi các class
- Các “danh từ” không biến thành class
 - Các thông tin mà giá trị của chúng là quan trọng
 - Các thông tin đdc sở hữu bởi 1 object duy nhất
 - Các thông tin không có hành vi

Review: What is an Association?

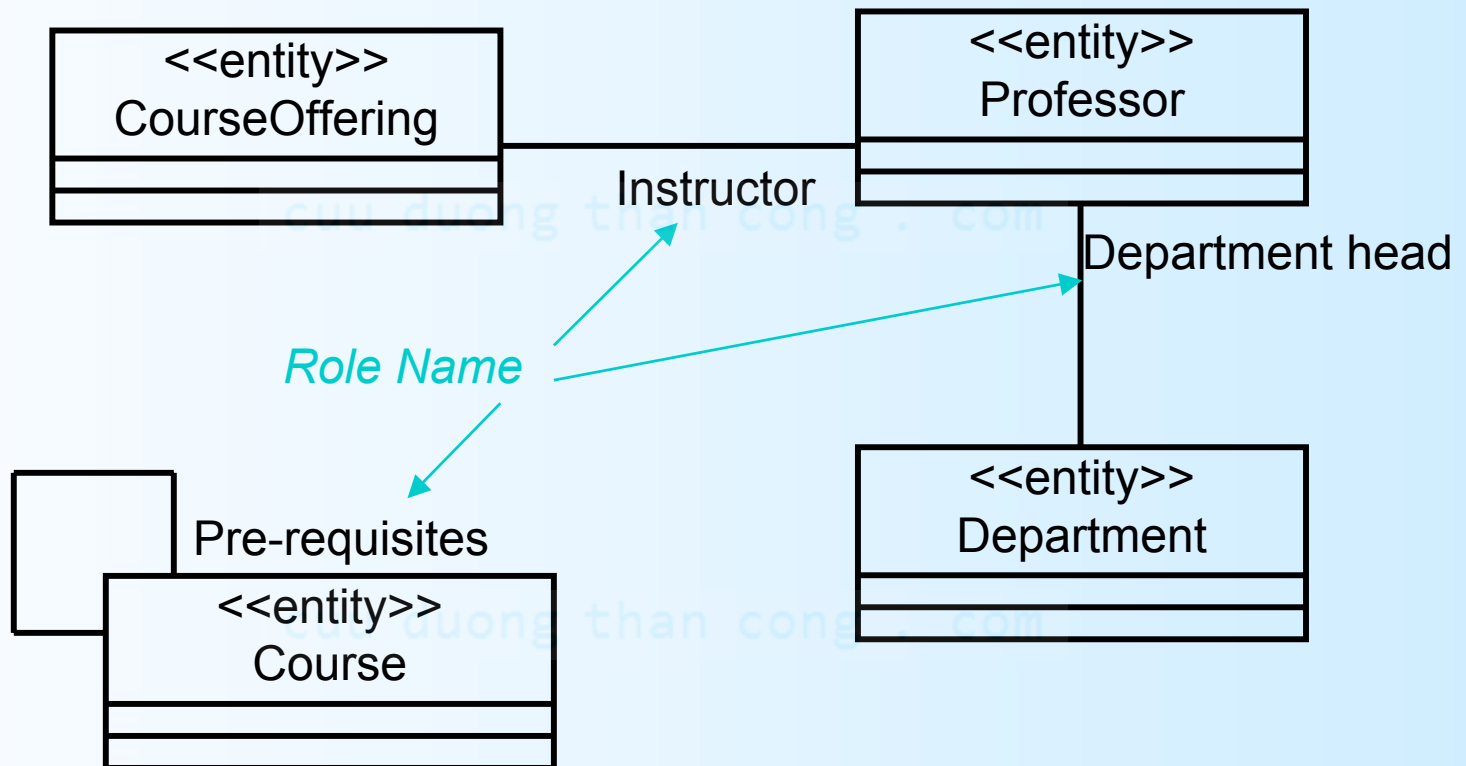
- Mô hình hóa một liên hệ ngữ nghĩa giữa các thể hiện (instances) của các class



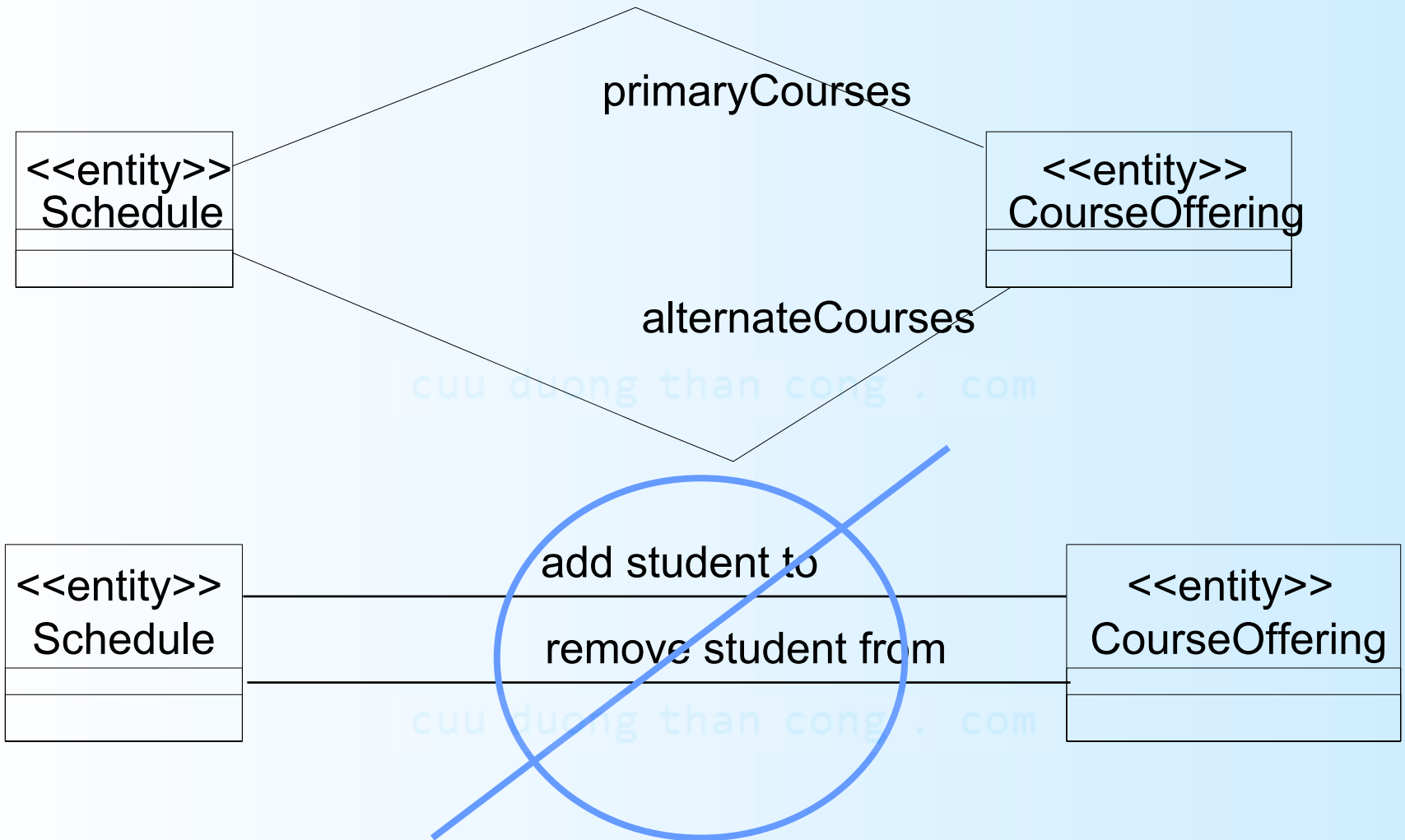
Association là một quan hệ cấu trúc

Review: What are Roles?

- “Nhân vật” mà một class “đóng vai” trong association



Ví dụ: Multiple Associations



Các Multiple association phải phản ánh multiple roles

Review: Multiplicity

- Không mô tả
- Duy nhất một
- Zero hoặc nhiều
(many, unlimited)

1

0..*

*

- Một hoặc nhiều
- Zero hoặc 1
- Một đoạn
- Nhiều đoạn rời nhau

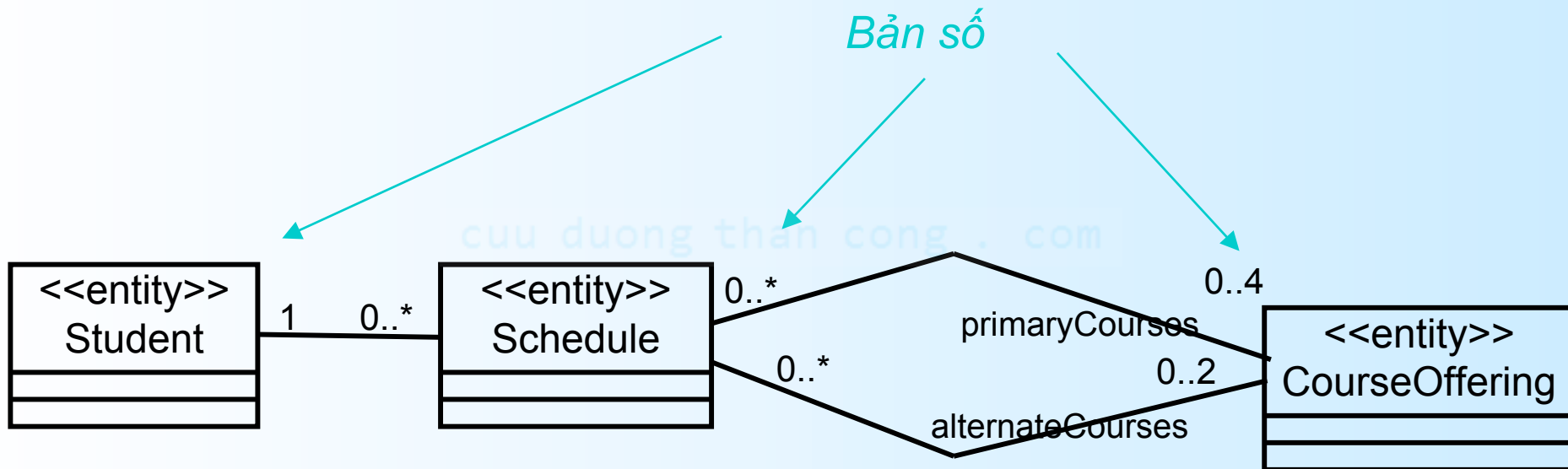
1..*

0..1

2..4

2, 4..6

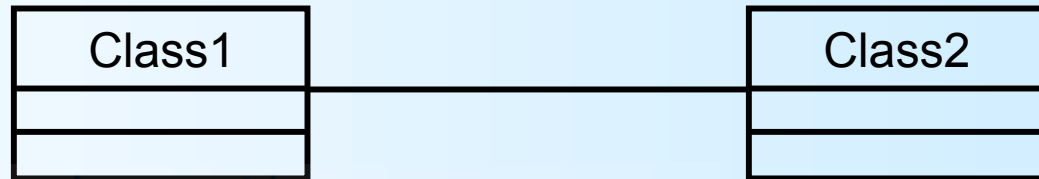
Ví dụ: Multiplicity



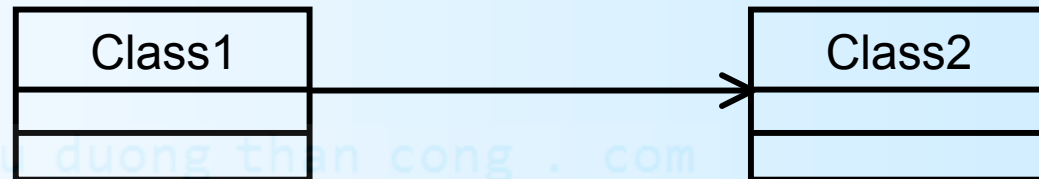
Review: Navigability

- Khả năng định hướng từ associating class đến target class

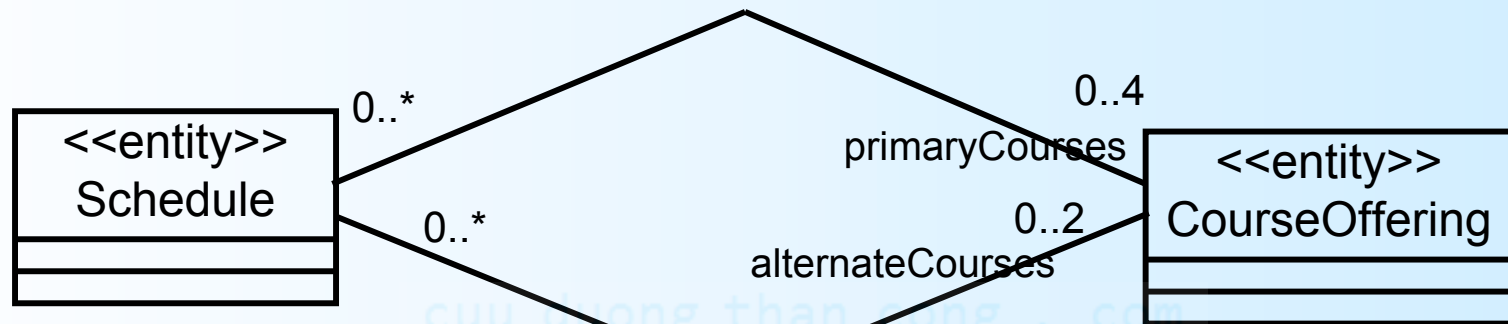
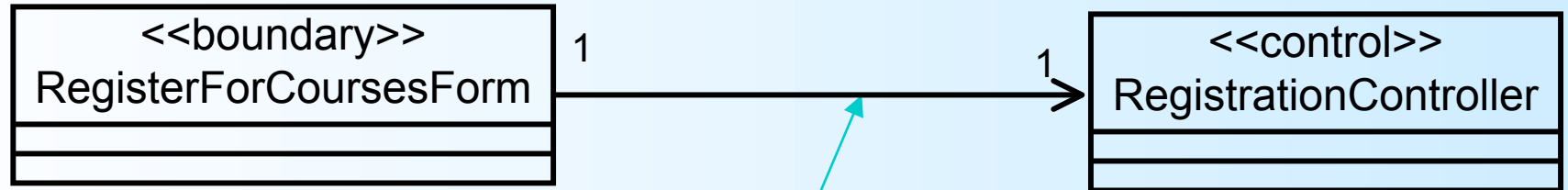
Hai chiều



Một chiều



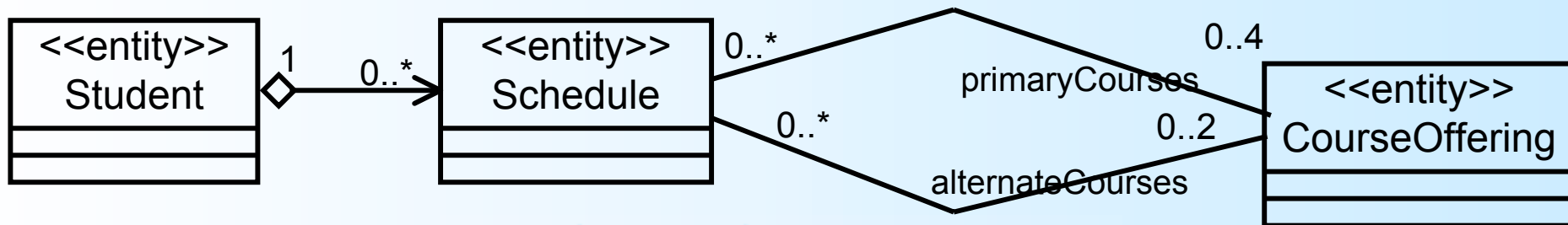
Ví dụ: Navigability



Review: What is Aggregation?

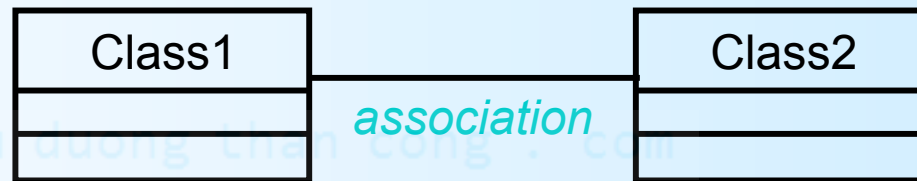
- Một dạng đặc biệt của association dùng để mô hình hóa một mối quan hệ toàn thể-bộ phận giữa toàn thể và các phần của nó

Whole/aggregate *part*



Association or Aggregation?

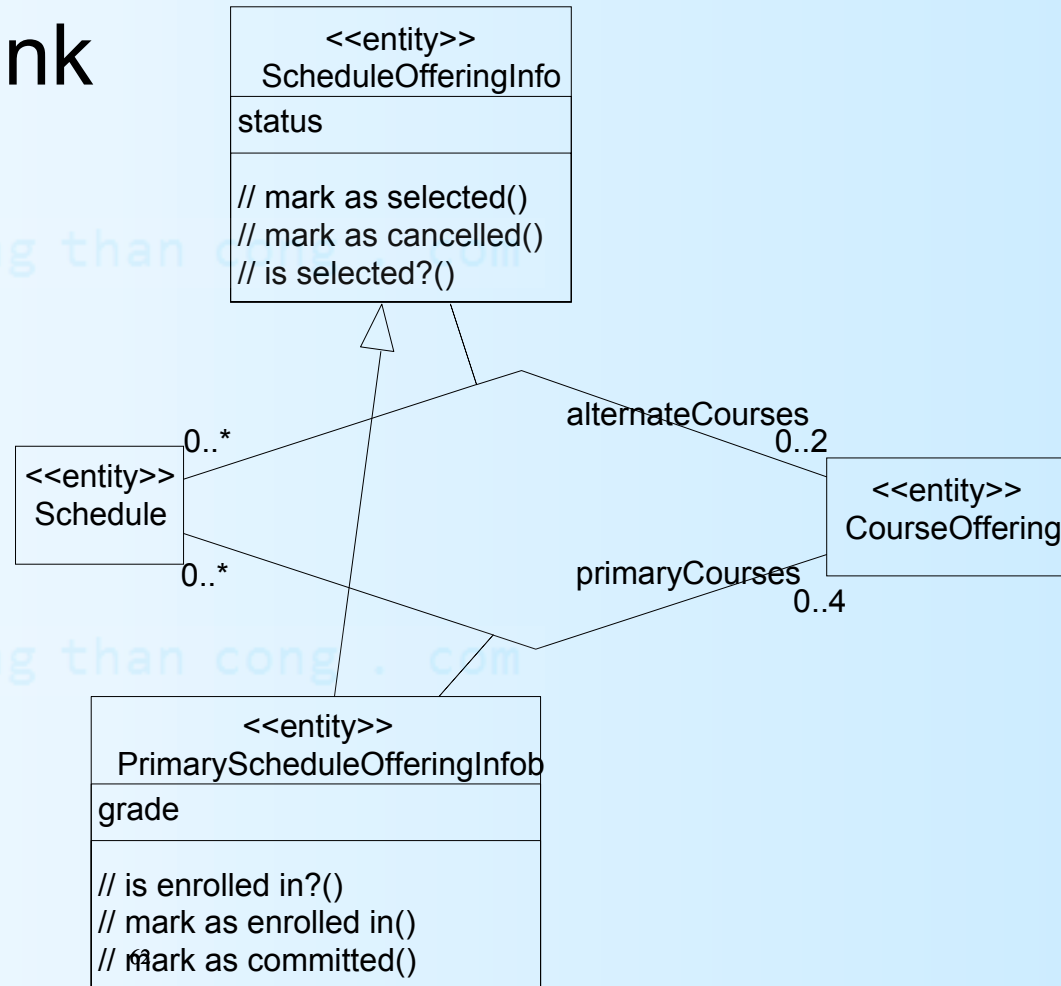
- Xem xét
 - Ngữ cảnh, các đặc trưng độc lập của Class2



Khi cảm thấy nghi ngờ hãy dùng association

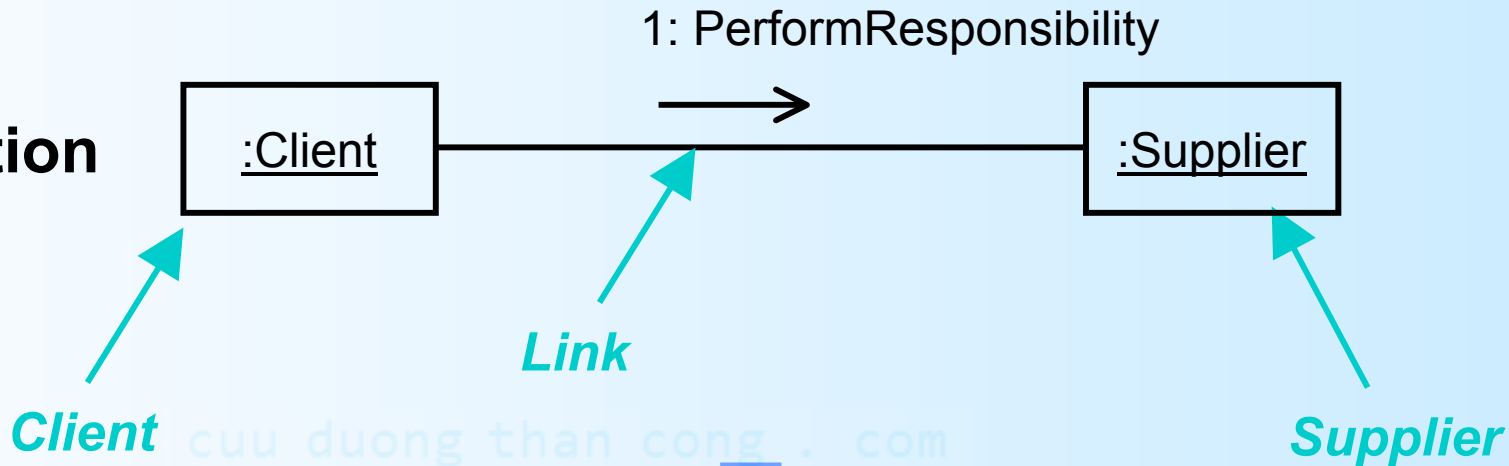
Association Class

- Một class “được gắn” vào một association
- Chứa các thuộc tính của relationship
- Một thể hiện / 1 link

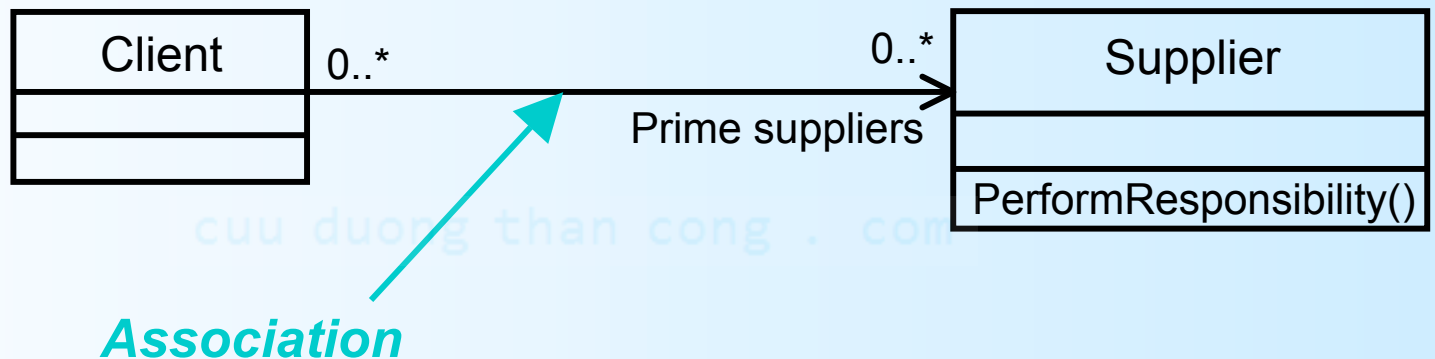


Finding Relationships

Collaboration Diagram

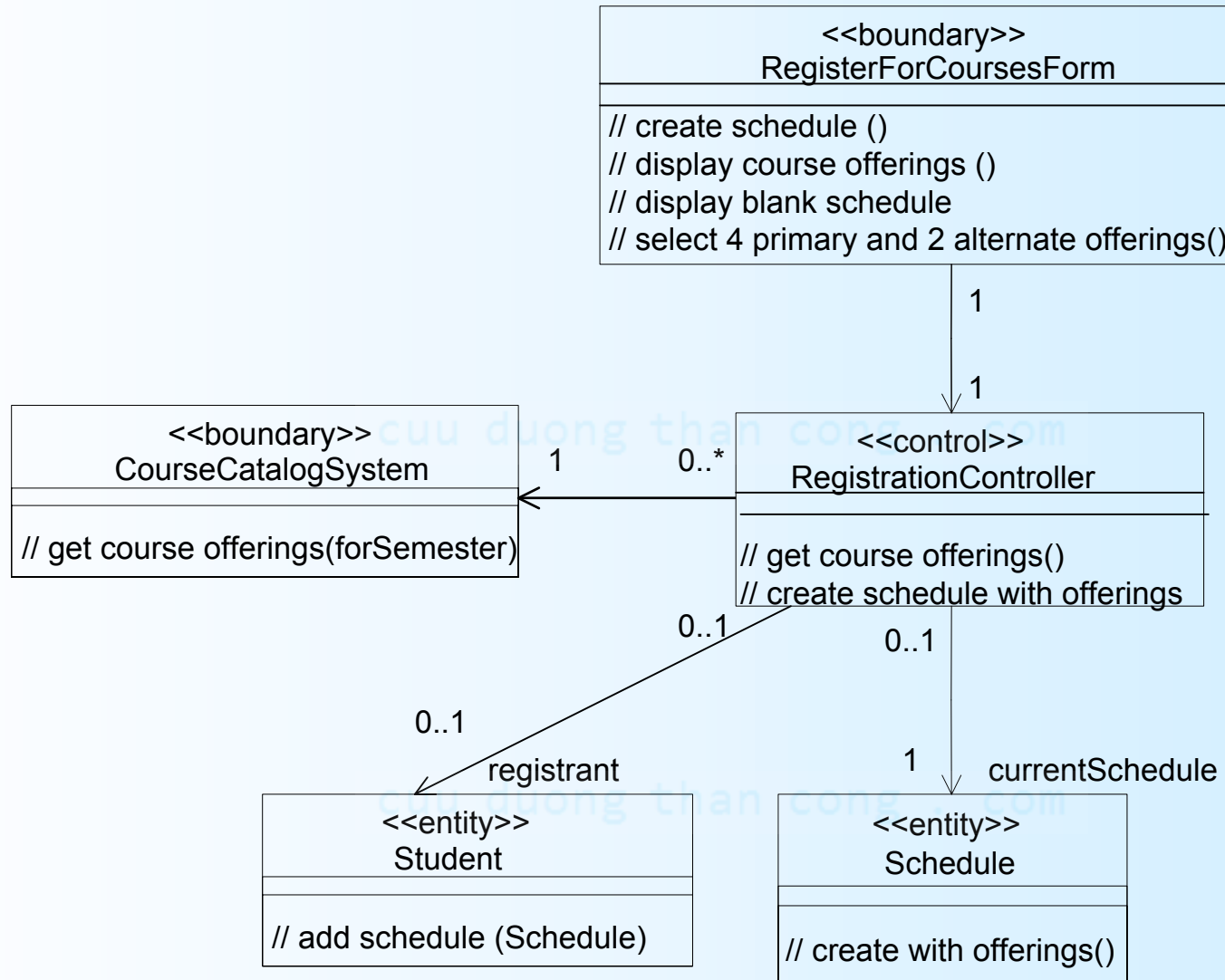


Class Diagram

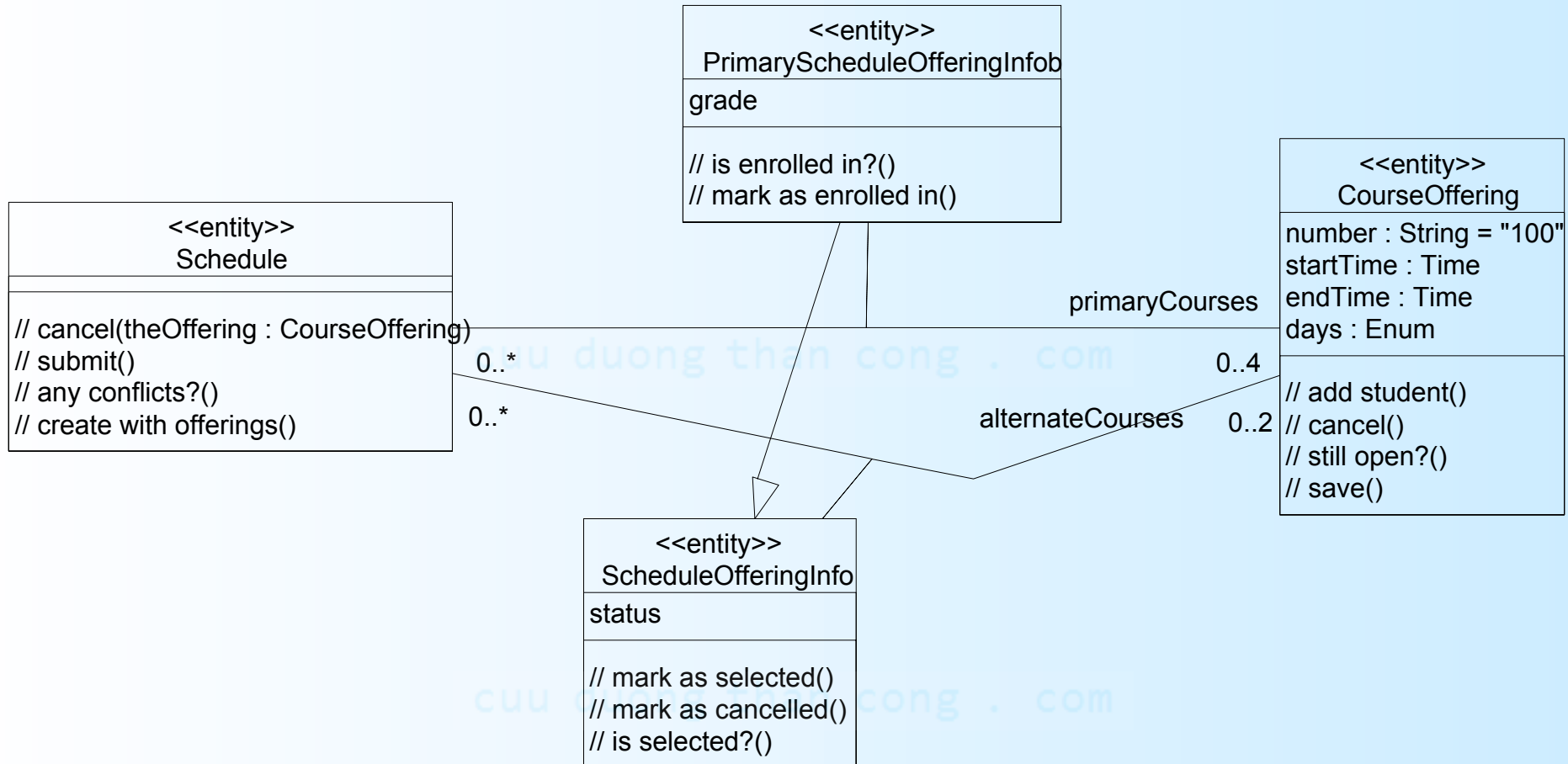


Tạo relationship cho mọi link!

Ví dụ: VOPC: Finding Relationships



Ví dụ: VOPC: Finding Relationships (contd)



Use-Case Analysis Steps

- Bổ sung các mô tả của Use-Case
- Đ/v mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior cho các Class
- Đ/v mỗi analysis class tìm được
 - Mô tả các nhiệm vụ của chúng
 - Mô tả các Attribute và Association
- ★ – Lượng giá các Analysis Mechanism
- Hợp nhất các Analysis Class
- Checkpoints

Describing Analysis Mechanisms

- Tập hợp tất cả analysis mechanisms trong 1 danh sách
- Vẽ một ánh xạ các client classes đến các analysis mechanism

Analysis Class	Analysis Mechanism(s)

- Xác định các đặc trưng của Analysis Mechanisms

Ví dụ: Describing Analysis Mechanisms

- Ánh xạ giữa Analysis class và analysis mechanism

Analysis Class	Analysis Mechanism(s)
Student	Persistency, Security
Schedule	Persistency, Security
CourseOffering	Persistency, Legacy Interface
Course	Persistency, Legacy Interface
RegistrationController	Distribution

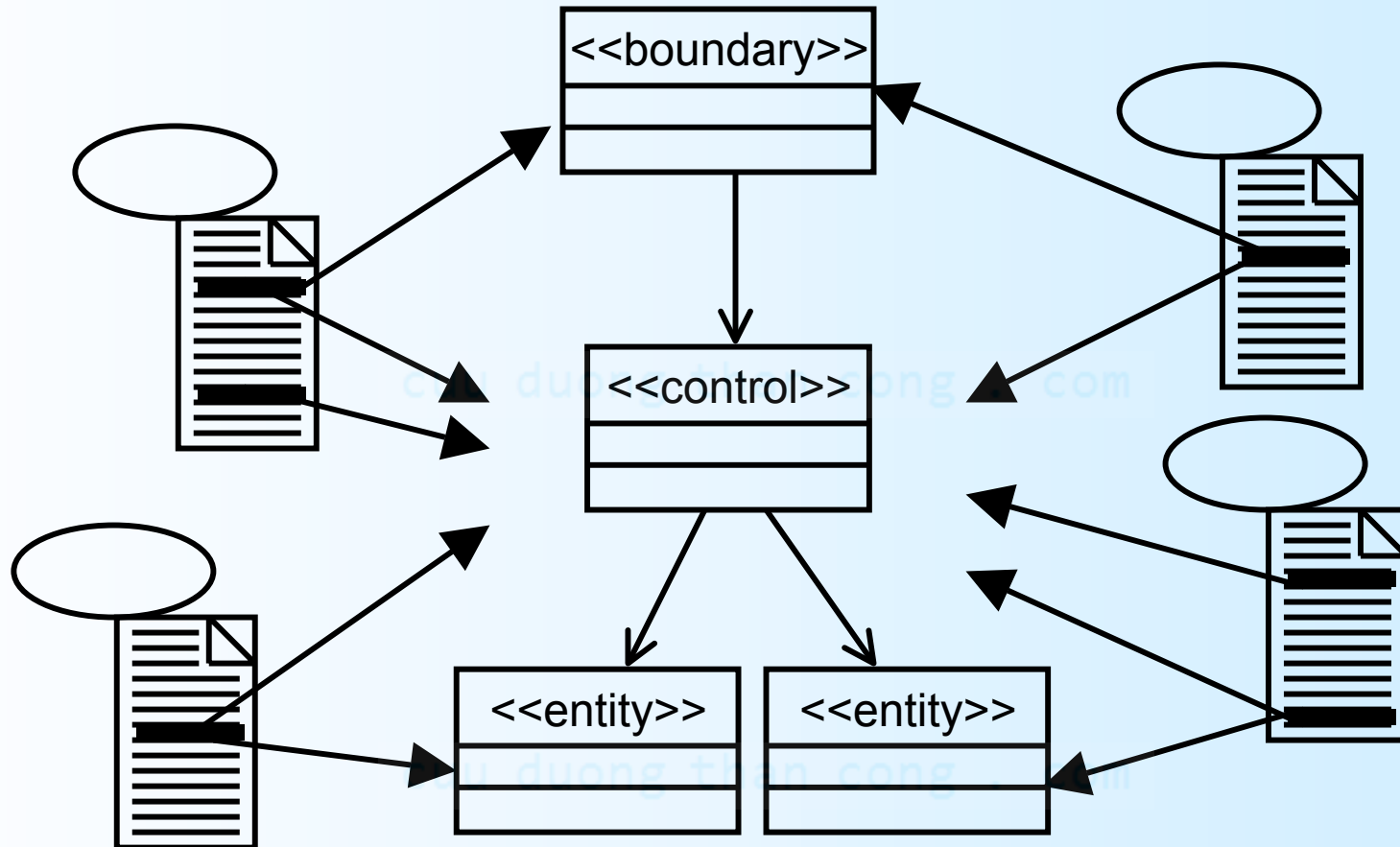
Ví dụ: Describing Analysis Mechanisms (cont.)

- Các đặc trưng của Analysis mechanism
- Persistency của Schedule class:
 - Granularity: 1 đến 10 Kbytes / sản phẩm
 - Volume: tối đa 2,000 schedule
 - Access frequency
 - Create: 500 lần / ngày
 - Read: 2,000 lần / giờ
 - Update: 1,000 lần / ngày
 - Delete: 50 lần / ngày
 - Etc.

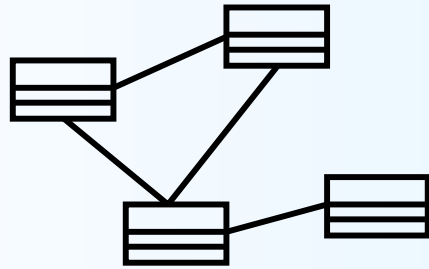
Use-Case Analysis Steps

- Bổ sung các mô tả của Use-Case
- Đ/v mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior cho các Class
- Đ/v mỗi analysis class tìm được
 - Mô tả các nhiệm vụ của chúng
 - Mô tả các Attribute và Association
 - Lượng giá các Analysis Mechanism
- ★ ● Hợp nhất các Analysis Class
- Checkpoints

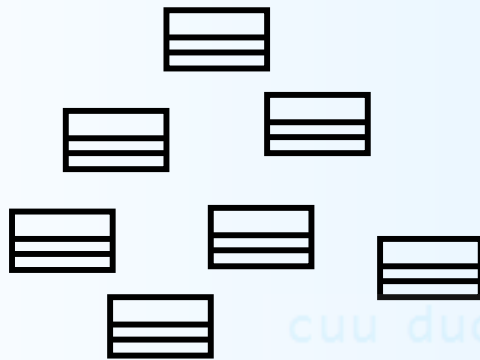
Unify Analysis Classes



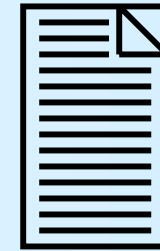
Evaluate Your Results



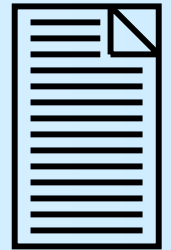
Design Model



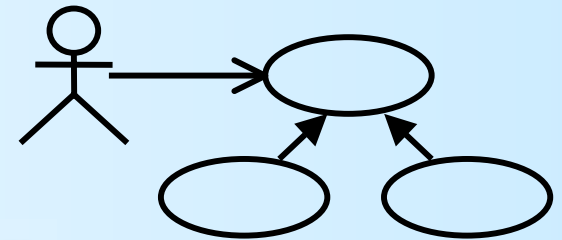
Analysis Classes



Supplementary
Specification



Glossary



Use-Case Model

Use-Case Analysis Steps

- Bổ sung các mô tả của Use-Case
- Đ/v mỗi use-case realization
 - Tìm các Class từ Use-Case Behavior
 - Phân bổ Use-Case Behavior cho các Class
- Đ/v mỗi analysis class tìm được
 - Mô tả các nhiệm vụ của chúng
 - Mô tả các Attribute và Association
 - Lượng giá các Analysis Mechanism
- Hợp nhất các Analysis Class
- ★ ● Checkpoints

Checkpoints: Analysis Classes

- Các class có hợp lý không?
- Tên của các class có phản ánh đúng vai trò của chúng?
- Class có biểu diễn 1 single well-defined abstraction?
- Tất cả các attribute và responsibility có gắn kết với nhau về mặt chức năng không?
- Class có cung cấp các hành vi được y/c?
- Tất cả các y/c cụ thể đã được thể hiện trên class chưa?

Checkpoints: Use-Case Realizations

- Tất cả các luồng chính và luồng con đã được điều khiển chưa, bao gồm cả các t/h ngoài lệ?
- Đã tìm thấy tất cả các đối tượng cần thiết?
- Đã phân phối một cách rõ ràng tất cả các hành vi về các đối tượng chưa?
- Các hành vi có được phân phối về đúng đối tượng không?
- Các interaction diagrams nằm ở đâu, mối quan hệ giữa chúng có rõ ràng và phù hợp không?

Review: Use-Case Analysis

- Mục tiêu của Use-Case Analysis là gì?
- Một analysis class là gì? Cho biết tên và mô tả về 3 analysis stereotype.
- Use-case realization là gì?
- Mô tả một vài hoạt động khảo sát when đặt các trách nhiệm cho các analysis class.
- Bao nhiêu interaction diagram phải được xây dựng trong giai đoạn Use-Case Analysis?

Exercise: Use-Case Analysis, Part 2

- Hãy cho biết các khái niệm sau:
 - Các Requirements artifact, đặc biệt là đặc tả bổ sung
 - Các cơ chế phân tích có thể
 - Các flow of events interaction diagram cho một use case cụ thể

cuu duong than cong . com

(continued)

Exercise: Use-Case Analysis, Part 2 (cont.)

- Với mỗi use case hãy xác định các dữ kiện sau:
 - Các thuộc tính và các mối quan hệ của Analysis class
 - Các cơ chế phân tích Analysis class

cuu duong than cong . com

(continued)

Exercise: Use-Case Analysis, Part 2 (cont.)

- Xây dựng các lược đồ sau:
 - VOPC class diagram, chứa các analysis class, stereotype của chúng, nhiệm vụ, các attribute, và relationship.
 - Ánh xạ Analysis class với các cơ chế phân tích

cuu duong than cong . com