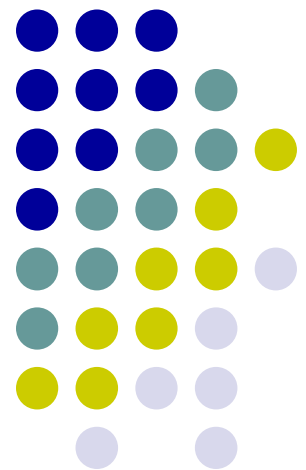


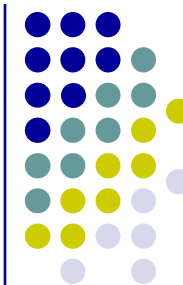
Phân tích hướng đối tượng UML

Giáo viên: Đỗ Thị Mai Hương

Bộ môn : Các hệ thống thông tin

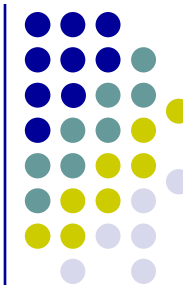
Khoa : CNTT - Học viện kỹ thuật quân sự





Bài 2

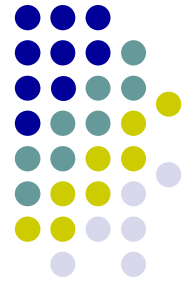
Ngôn ngữ mô hình hóa thống nhất UML



Mô hình là gì?

- Mô hình
 - là bức tranh hay mô tả vấn đề đang cố gắng giải quyết hay mô tả chính giải pháp vấn đề
 - là ngôn ngữ của người thiết kế (trong nhiều lĩnh vực)
 - là trình diễn hệ thống sẽ xây dựng
 - là phương tiện giao tiếp giữa người phân tích, thiết kế và người dùng
 - là kế hoạch chi tiết (blueprints)
- Mô hình cho khả năng suy diễn một số đặc tính của hệ thống thực
- Mô hình hóa trực quan
 - Bằng các phần tử đồ họa
- Ngôn ngữ mô hình hóa là ngôn ngữ mô tả hệ thống hay tác nghiệp

Thí dụ mô hình



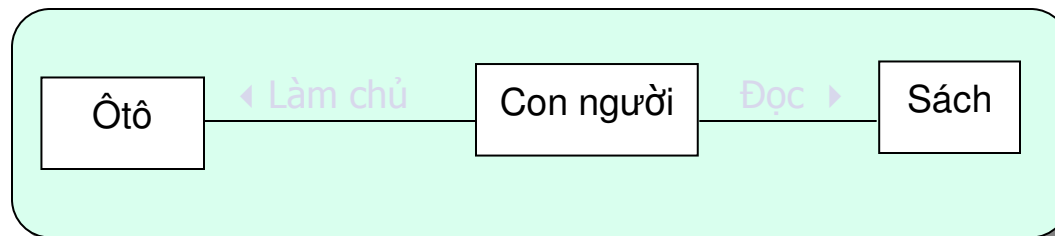
Thế giới thực



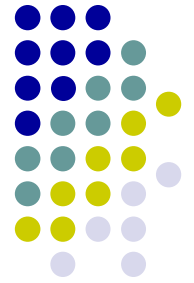
Mô hình: Quả địa cầu học sinh



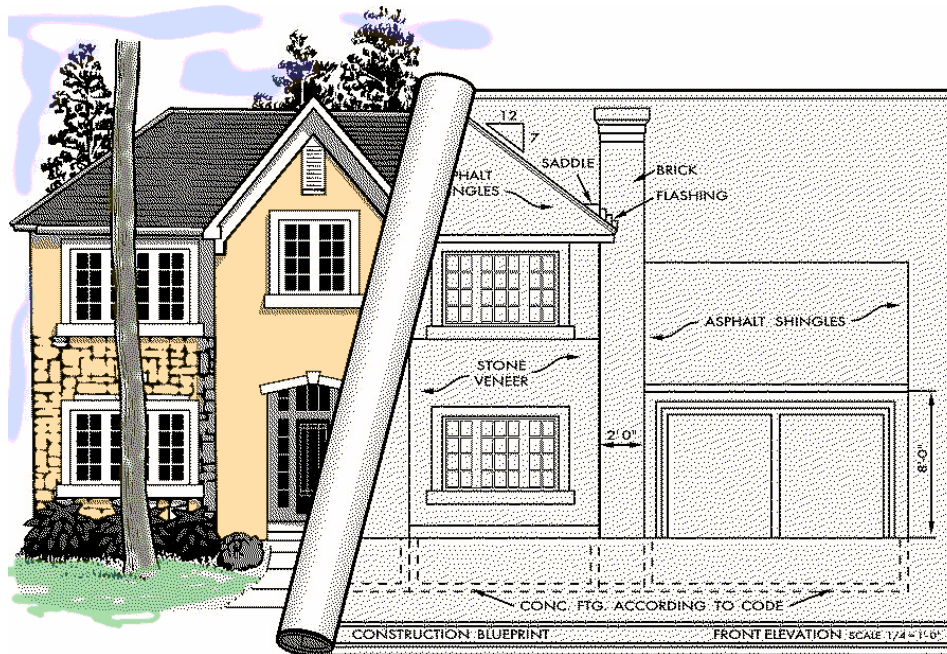
Thế giới thực



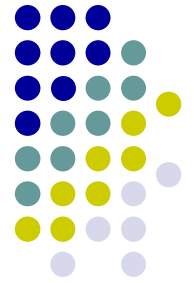
Mô hình



Thí dụ mô hình

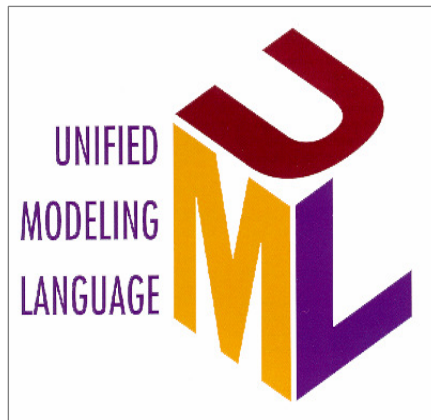


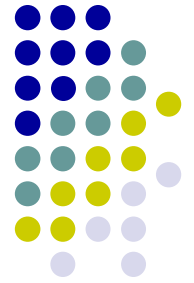
A **model** is a complete description of a system from a particular perspective



UML là gì?

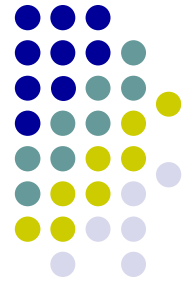
- UML là một ngôn ngữ dùng cho
 - Mô hình hóa trực quan (Visualizing)
 - Đặc tả (Specifying)
 - Xây dựng (Constructing)
 - Tài liệu hóa (Documenting)
- các nhân tố của hệ thống phần mềm





Mô hình hóa trực quan

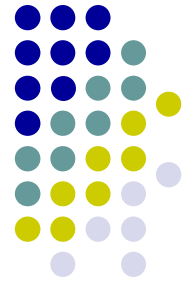
- Giúp cho việc suy nghĩ về cài đặt và bản cài đặt thực tế được nhất quán
- Giúp lập trình viên, phân tích viên:
 - Dễ dàng trao đổi những mô hình trong phần mềm với nhau
 - Hiểu rõ toàn bộ hệ thống và vai trò của mình trong hệ thống
 - Thông tin được lưu trữ rõ ràng, dễ tra cứu quản lý



Đặc tả

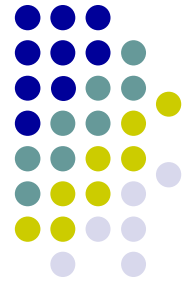
- Đặc tả giúp cho xây dựng các mô hình
 - Chính xác
 - Rõ ràng, không nhập nhằng
 - Đầy đủ
- UML giúp giải quyết các vấn đề
 - Phân tích
 - Thiết kế
 - Cài đặt

Được rõ ràng, thông suốt trong quá trình phát triển phần mềm



Xây dựng

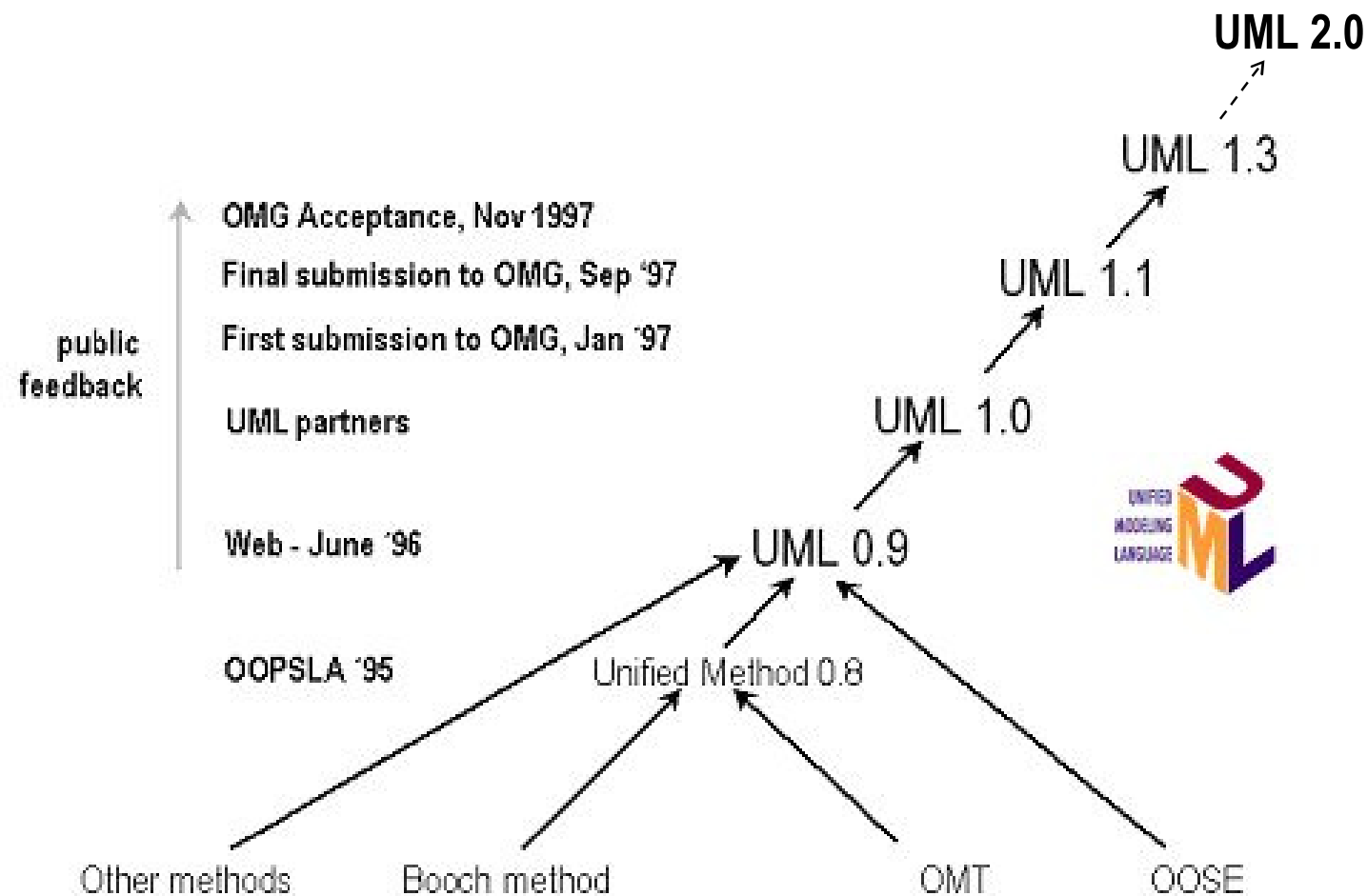
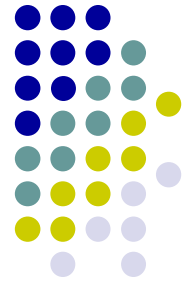
- Các mô hình UML có thể ánh xạ sang
 - Một ngôn ngữ lập trình
 - Java
 - C++/C#
 - Visual Basic/ .NET
 - Một bảng trong CSDL quan hệ (R-DBMS)
 - Một lưu trữ bền vững của CSDL HĐT (OO-DBMS)

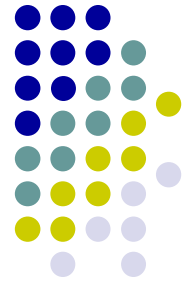


Tài liệu hóa

- UML giúp ghi chép, tài liệu hóa lại kiến trúc của hệ thống
 - Mô tả các yêu cầu (Requirements)
 - Các bộ kiểm thử (Tests)
 - Mô hình hóa các hoạt động, lập lịch của project, giảm khó khăn cho công việc quản lý

Sơ lược về lịch sử UML

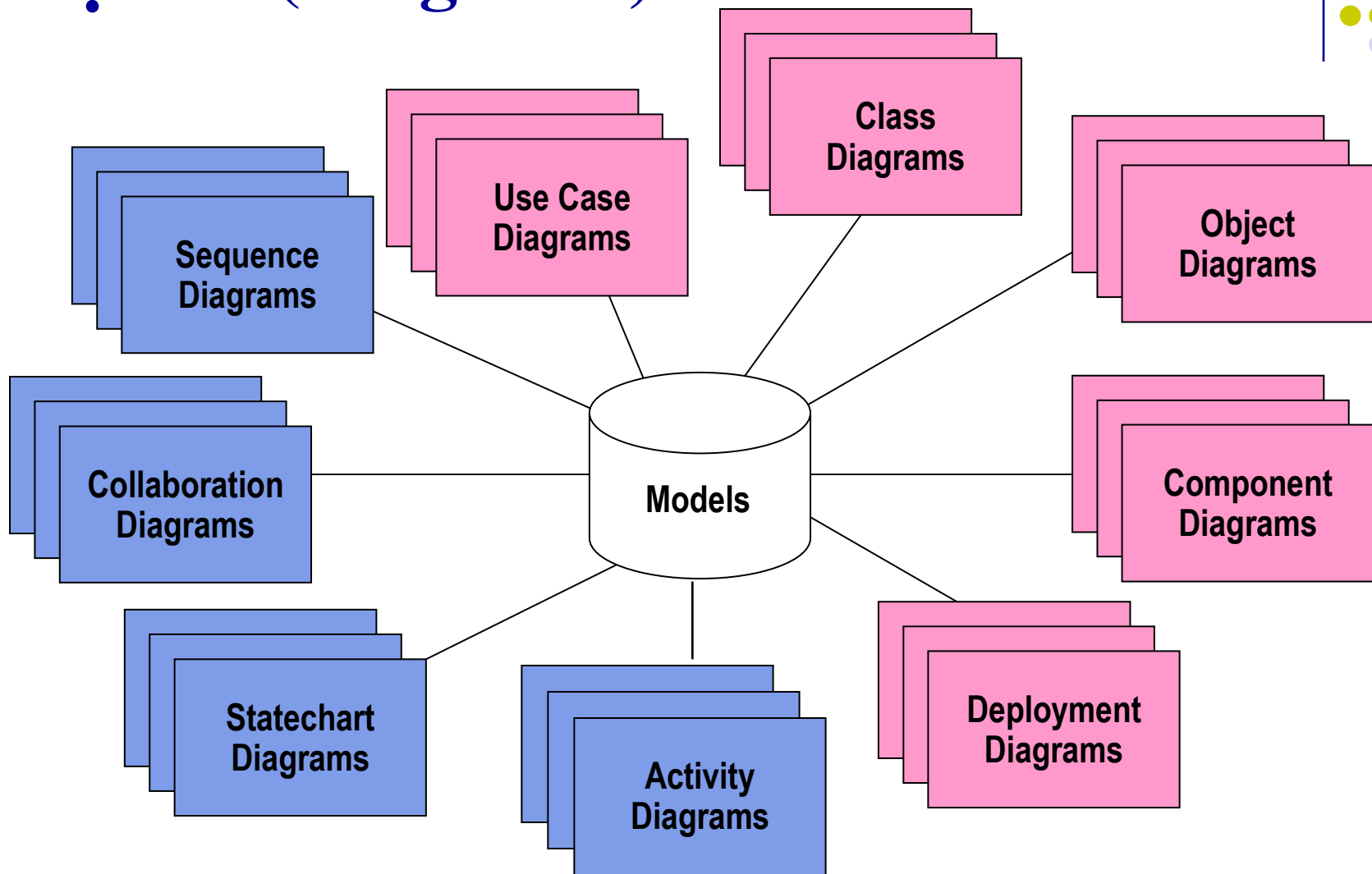
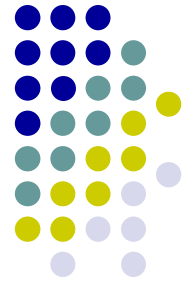


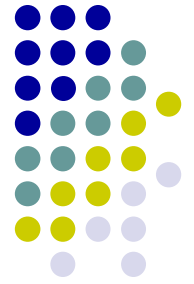


Tại sao chúng ta cần UML?

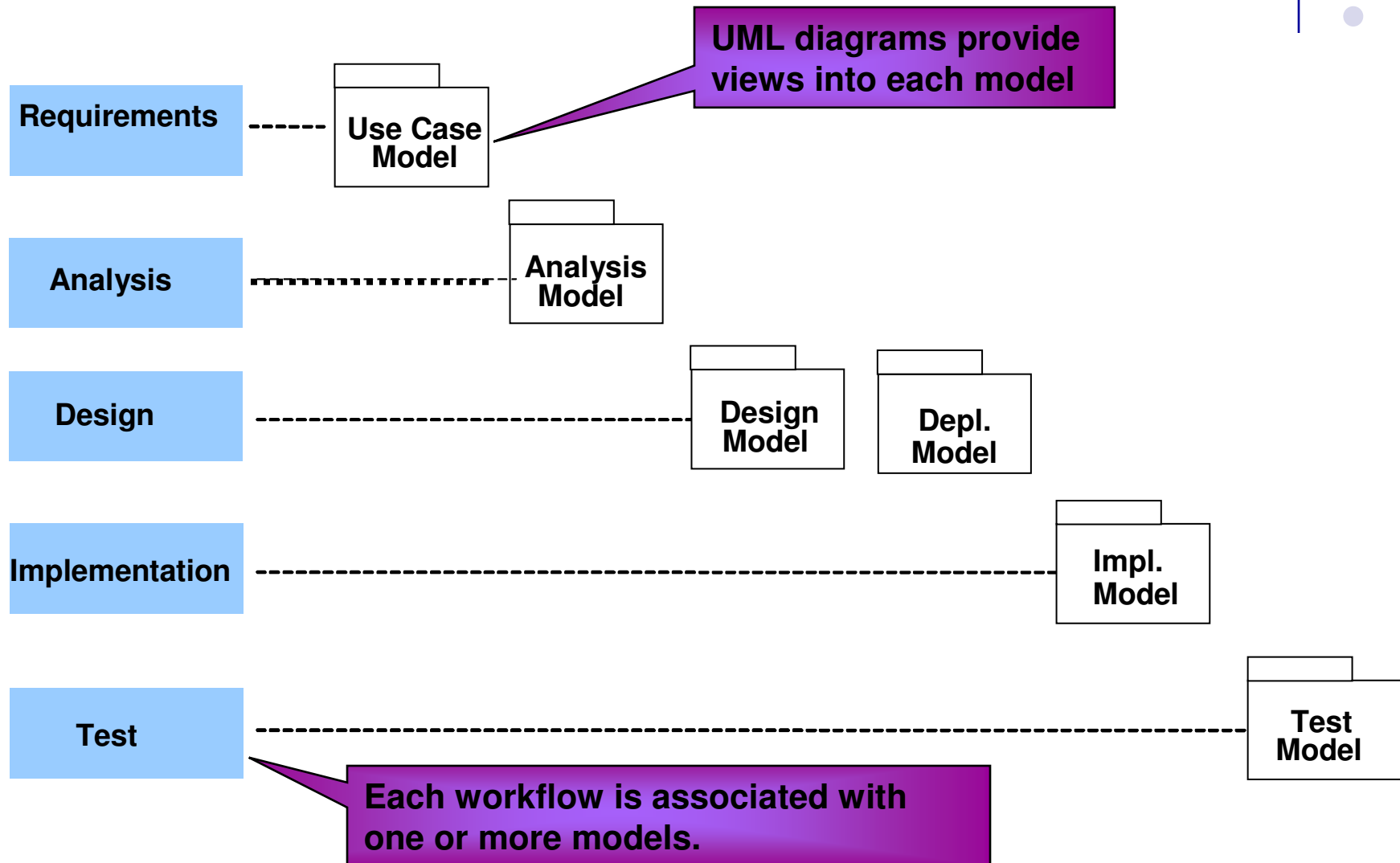
- Một ngôn ngữ mô hình hóa chuẩn, trực quan bằng hình ảnh
- Phát triển hệ thống hiệu quả, thiết kế chính xác và hiệu quả
- Dễ dàng giao tiếp giữa các nhóm trong cùng project
- Giao tiếp dễ dàng giữa các người liên quan đến project (khách hàng, nhà phát triển,...)
- Cho “cái nhìn tổng thể” về project

Lược đồ (Diagrams)

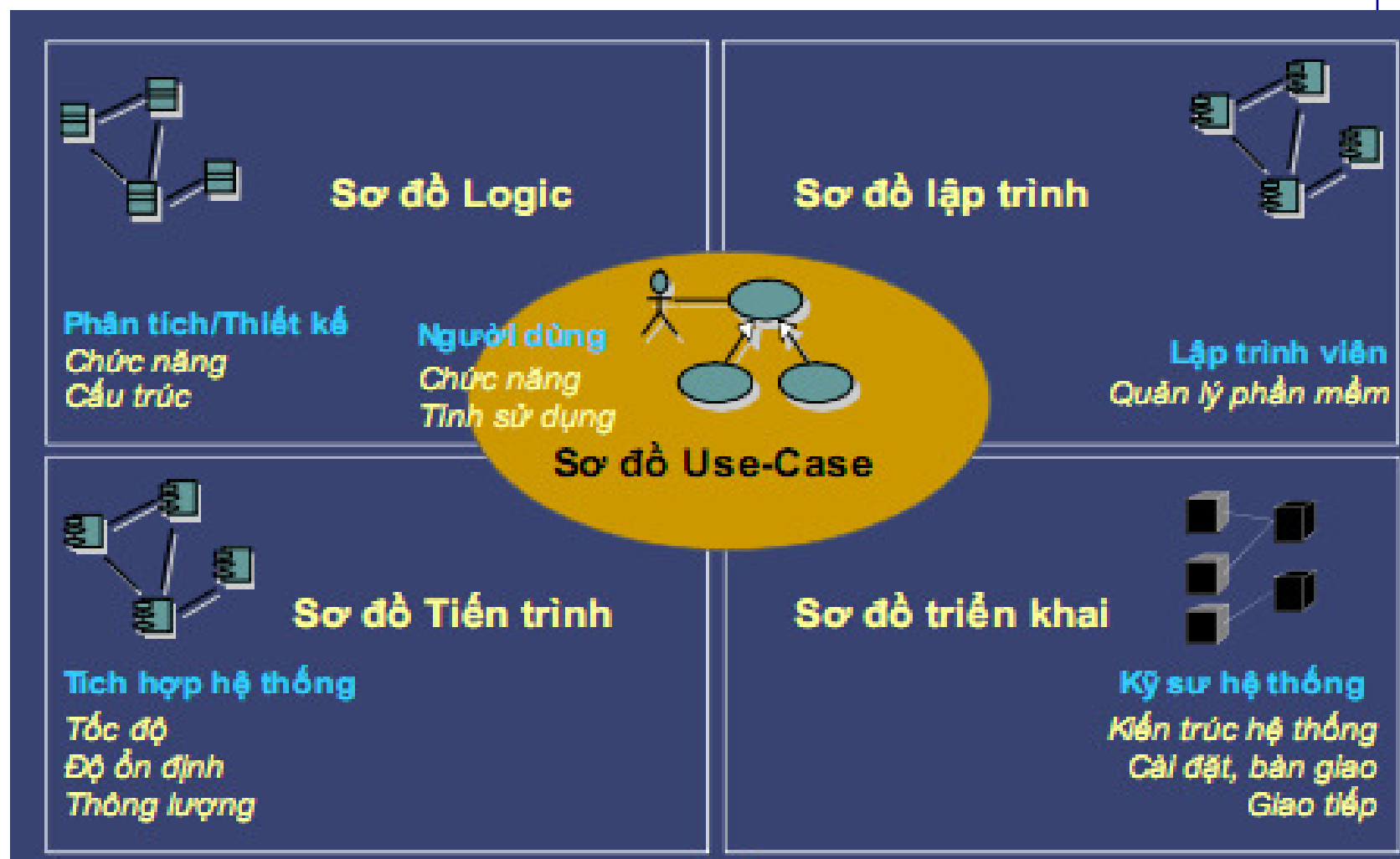
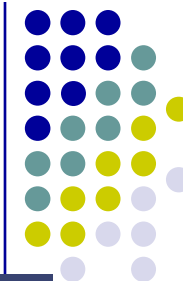


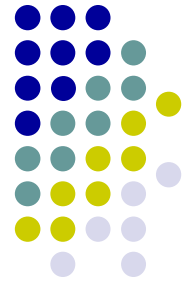


Workflows and Models



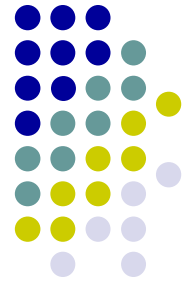
Representing System Architecture





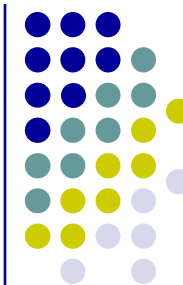
Cần bao nhiêu khung nhìn?

- Mô hình phù hợp với ngữ cảnh phát triển hệ thống
- Không phải tất cả các mô hình đòi hỏi đầy đủ khung nhìn
 - Đơn xử lý: Bỏ qua khung nhìn triển khai
 - Đơn tiến trình: Bỏ qua khung nhìn tiến trình
 - Chương trình rất nhỏ: Bỏ qua khung nhìn cài đặt
- Bổ sung các khung nhìn
 - Data view
 - Security view



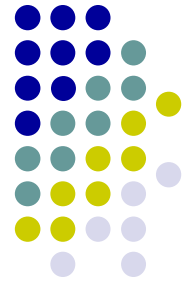
UML Concepts

- UML được sử dụng để:
 - Hiện thị biên hệ thống và các chức năng chính của nó bằng use cases và actors
 - Mô tả hiện thực use case bằng interaction diagrams
 - Biểu diễn các cấu trúc tĩnh của hệ thống bằng class diagrams
 - Mô hình hóa hành vi đối tượng bằng state transition diagrams
 - Biểu thị kiến trúc cài đặt vật lý bằng component & deployment diagrams
 - Mở rộng các chức năng bằng stereotypes



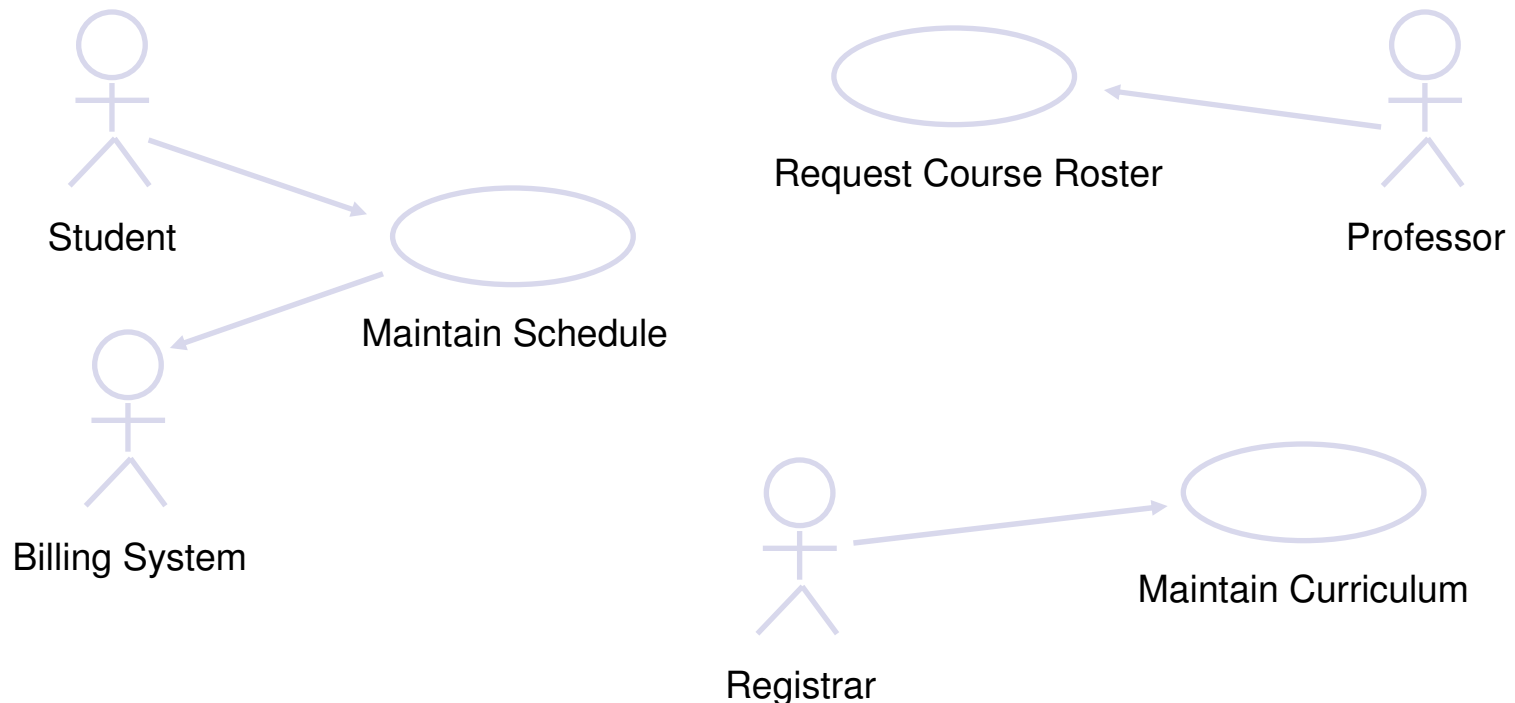
Thí dụ ứng dụng UML

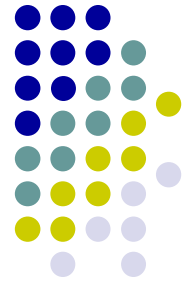
- Một trường đại học thực hiện tin học hóa hệ thống đăng ký học và dạy học:
 - Giáo vụ (Registrar) lập chương trình giảng dạy (curriculum) cho một học kỳ
 - Sinh viên (Student) chọn 4 môn học chính và 2 môn dự bị
 - Khi sinh viên đăng ký học thì hệ thống thanh toán (billing system) in hóa đơn học phí cho sinh viên
 - Sinh viên có thể sử dụng hệ thống để bổ sung/loại bỏ môn học sau khi đã đăng ký (trong khoảng thời gian cố định)
 - Giáo sư (Professors) sử dụng hệ thống để xem bảng phân công dạy học (course rosters)
 - Người sử dụng hệ thống đăng ký được cấp passwords để vào máy



Use case Diagram

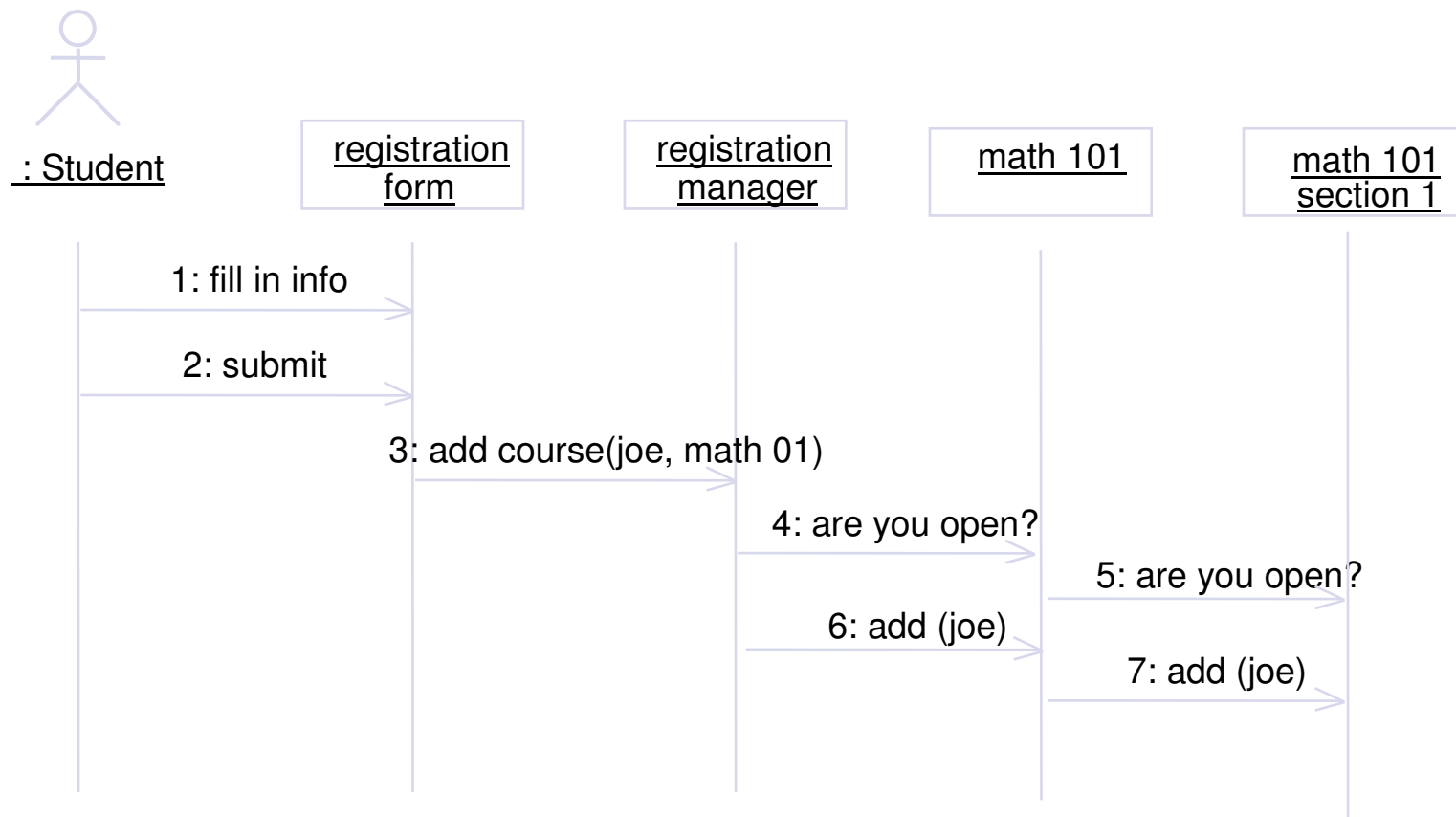
- Biểu diễn sơ đồ chức năng của hệ thống. Từ tập yêu cầu của hệ thống, biểu đồ use case sẽ phải chỉ ra hệ thống cần thực hiện điều gì để thoả mãn các yêu cầu của người dùng hệ thống đó.
- Đi kèm với biểu đồ use case là các kịch bản.



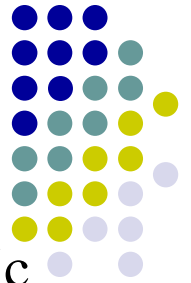


Sequence Diagram

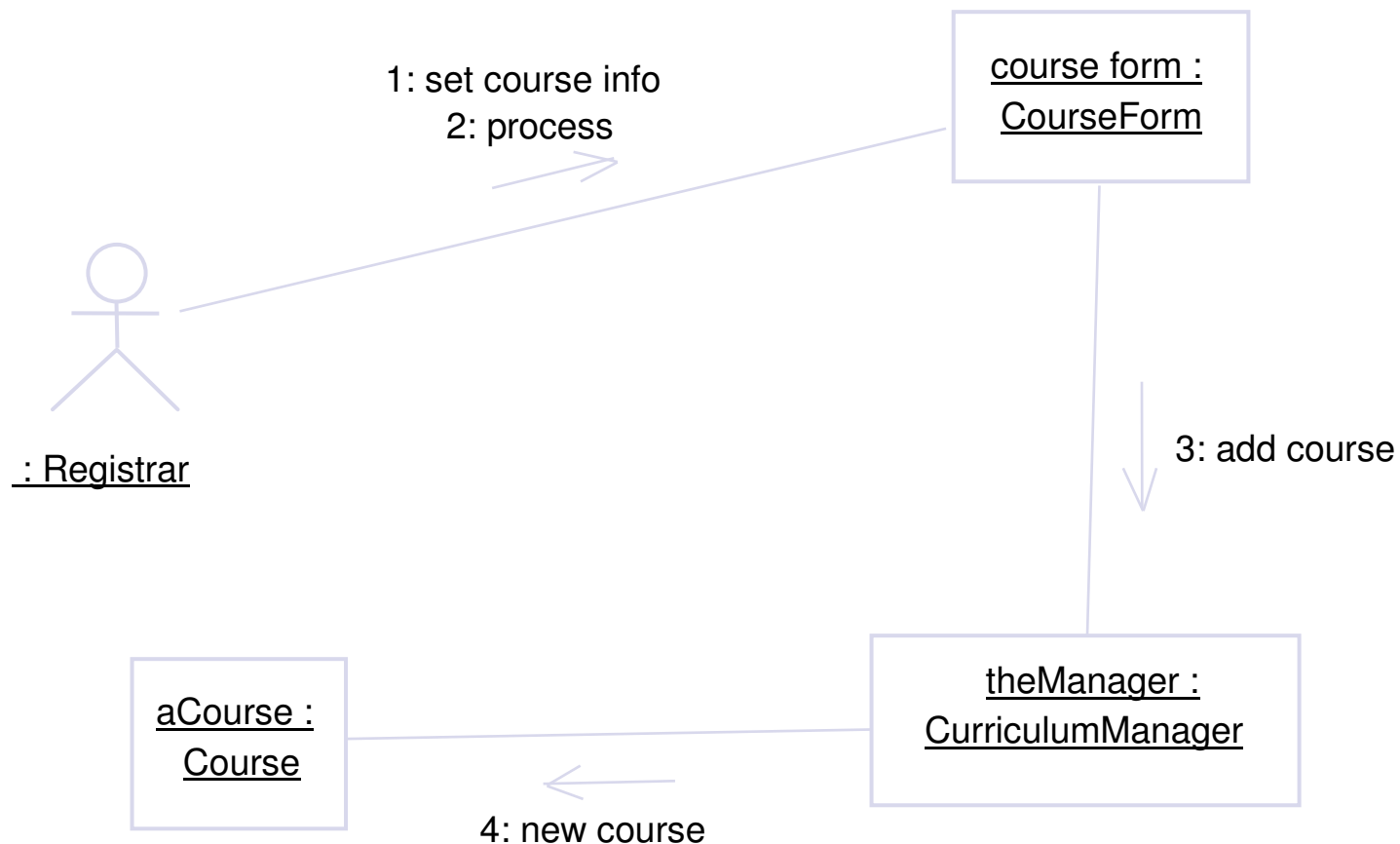
- Biểu diễn mối quan hệ giữa các đối tượng và giữa các đối tượng và tác nhân theo thứ tự thời gian.

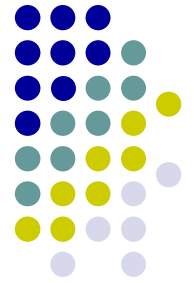


Collaboration Diagram



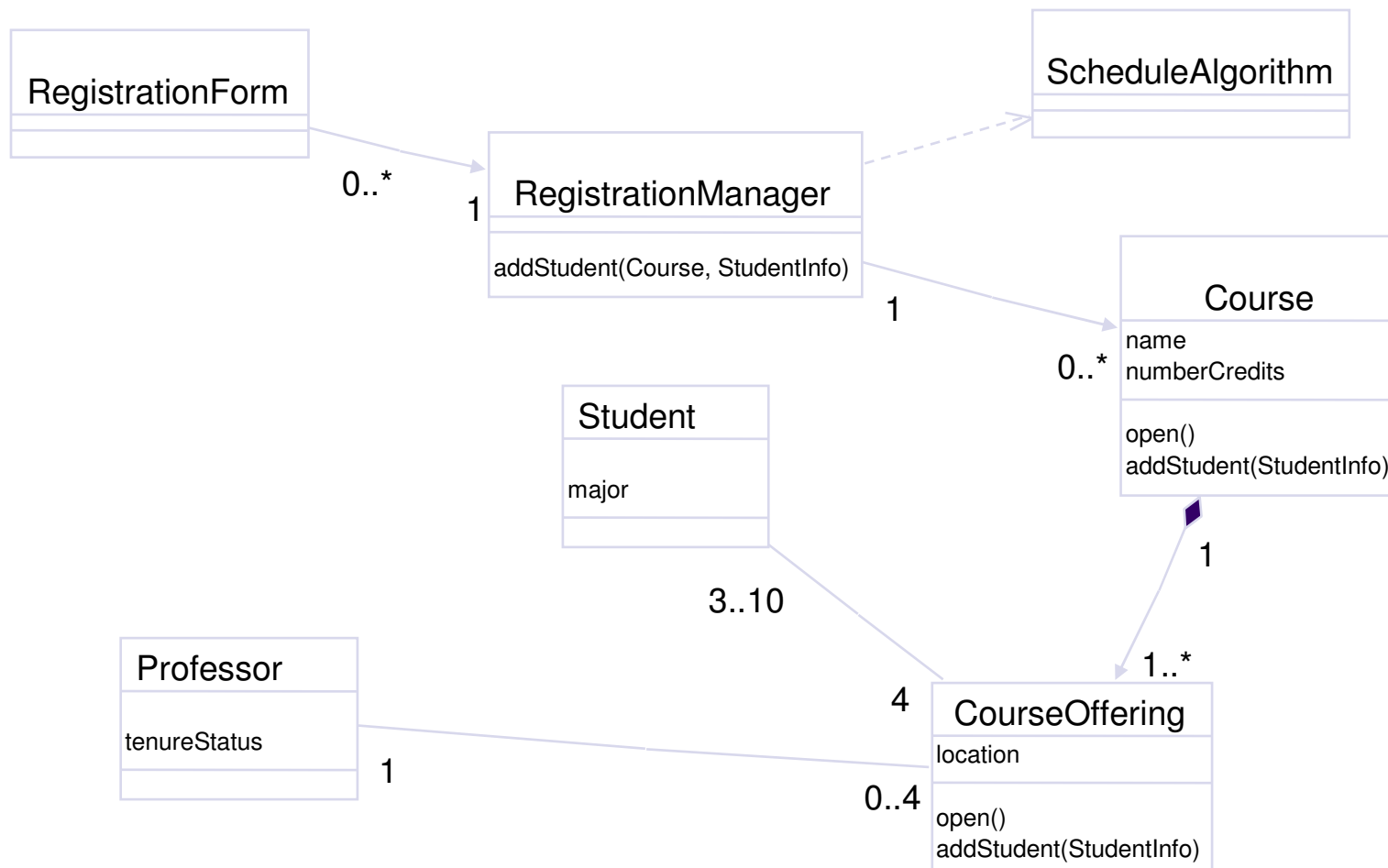
- Biểu diễn mối quan hệ giữa các đối tượng và giữa các đối tượng và tác nhân nhưng nhấn mạnh đến vai trò của các đối tượng trong tương tác.

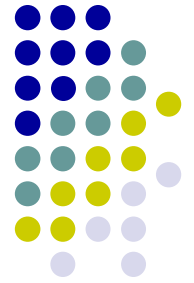




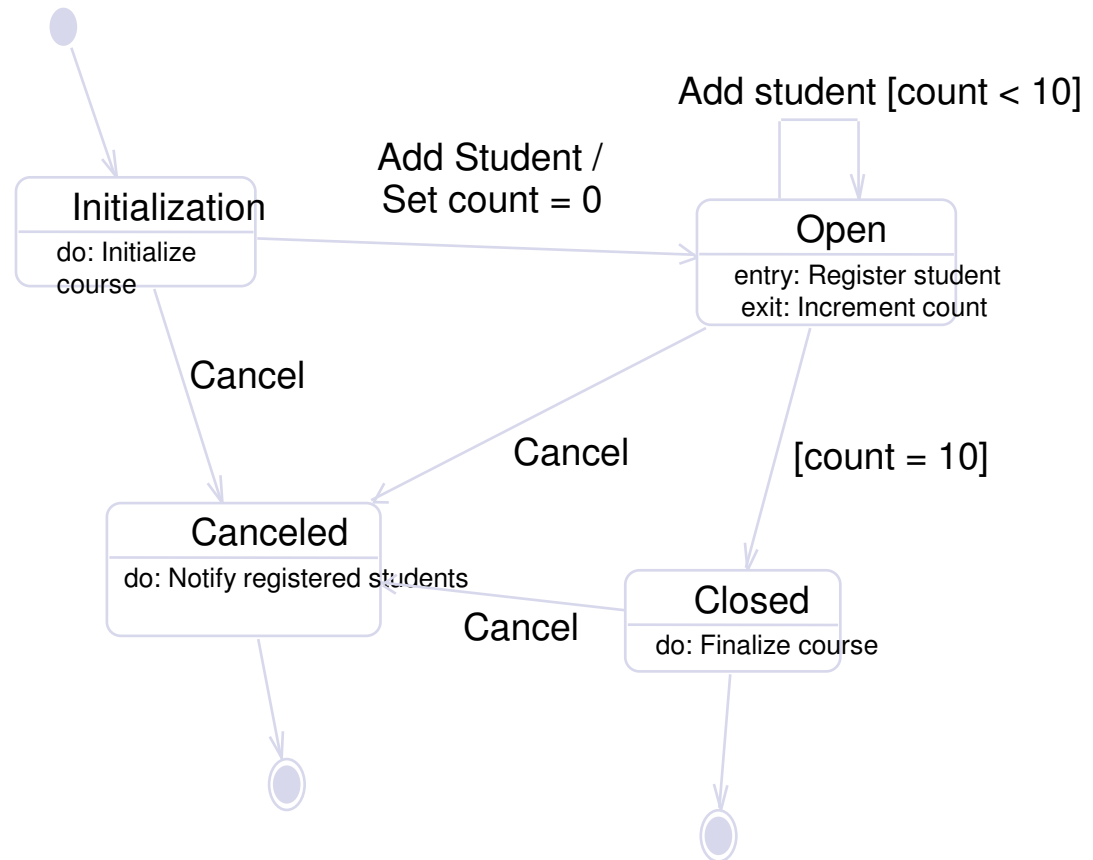
Class Diagram

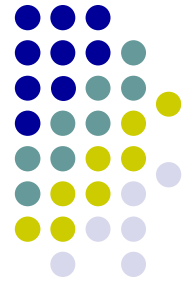
- Chỉ ra các lớp đối tượng trong hệ thống, các thuộc tính và phương thức của từng lớp và các mối quan hệ giữa những lớp đó.





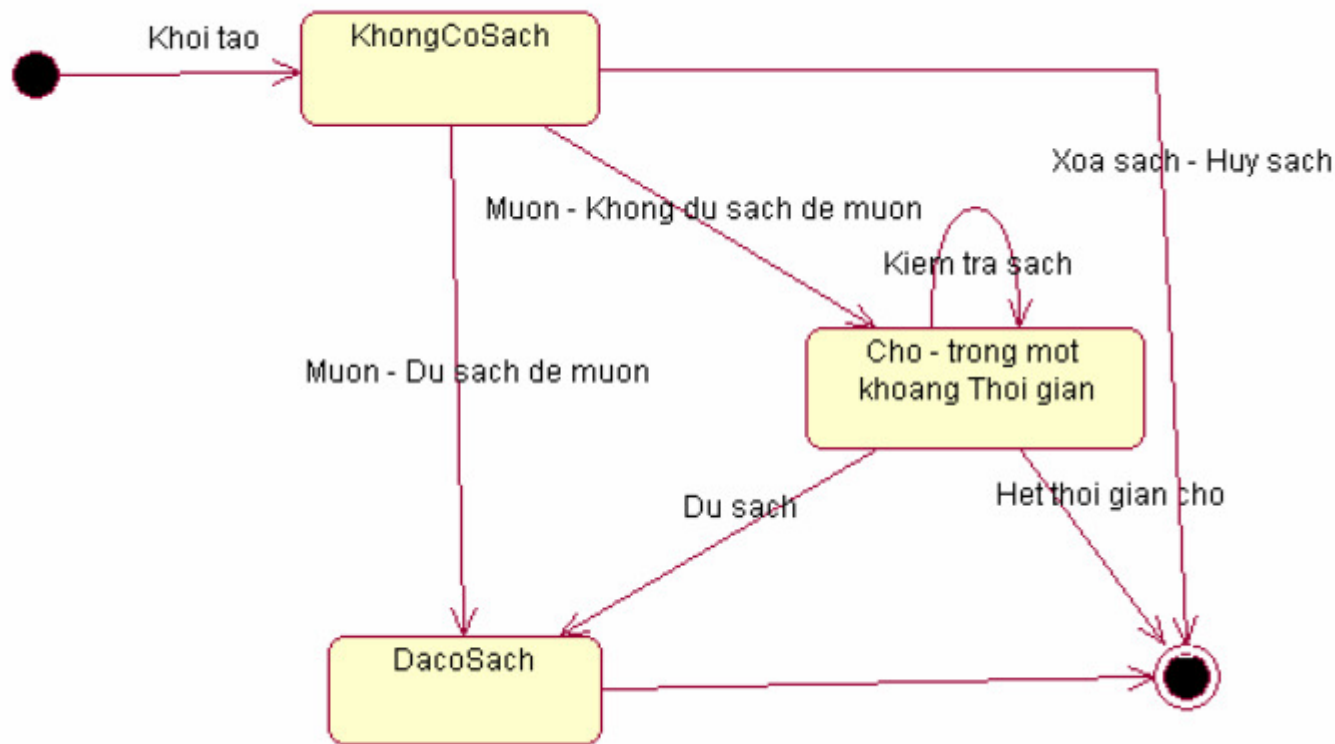
State Transition Diagram

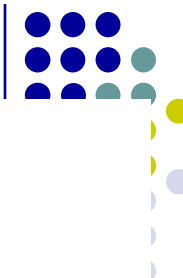




State Transition Diagram

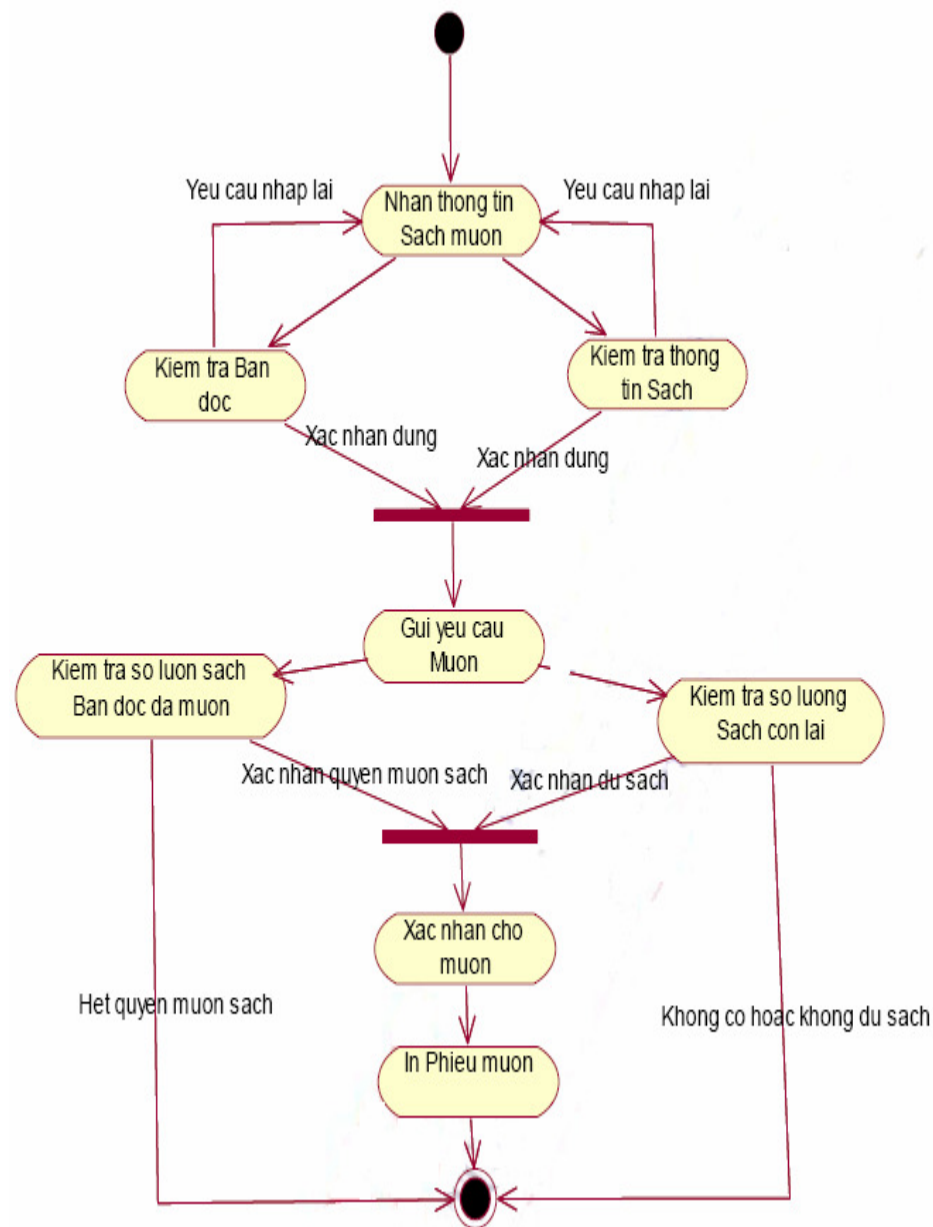
- Tương ứng với mỗi lớp sẽ chỉ ra các trạng thái mà đối tượng của lớp đó có thể có và sự chuyển tiếp giữa những trạng thái đó.



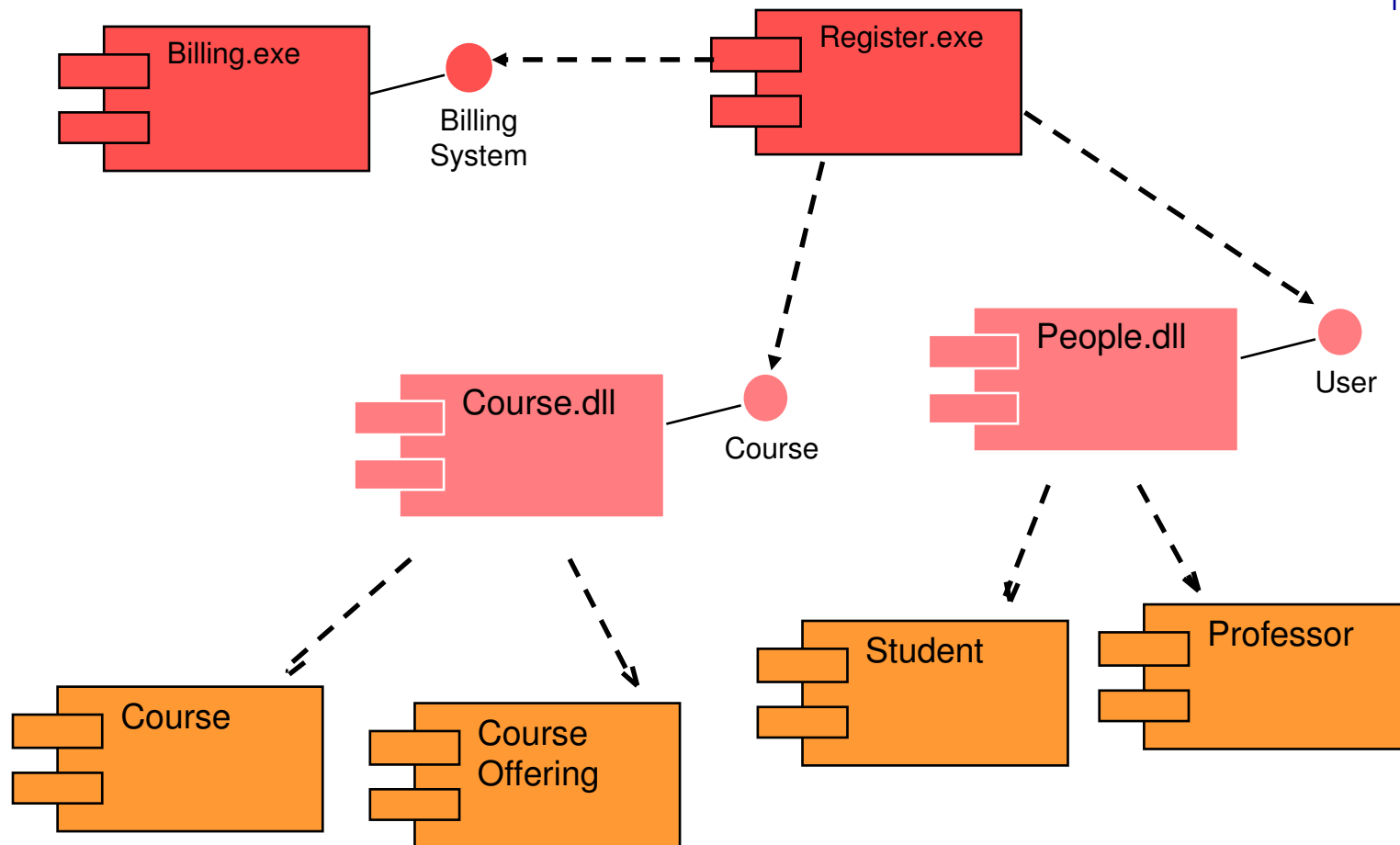
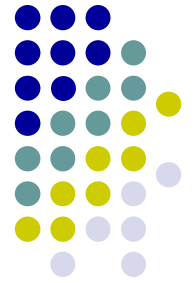


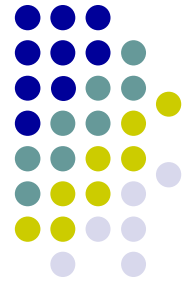
Activity Diagram

- Biểu diễn các hoạt động và sự đồng bộ, chuyển tiếp các hoạt động, thường được sử dụng để mô tả ca sử dụng hoặc biểu diễn các phương thức phức tạp của các lớp.



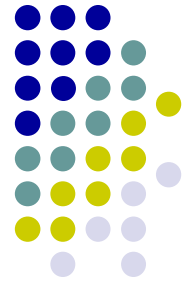
Component Diagram





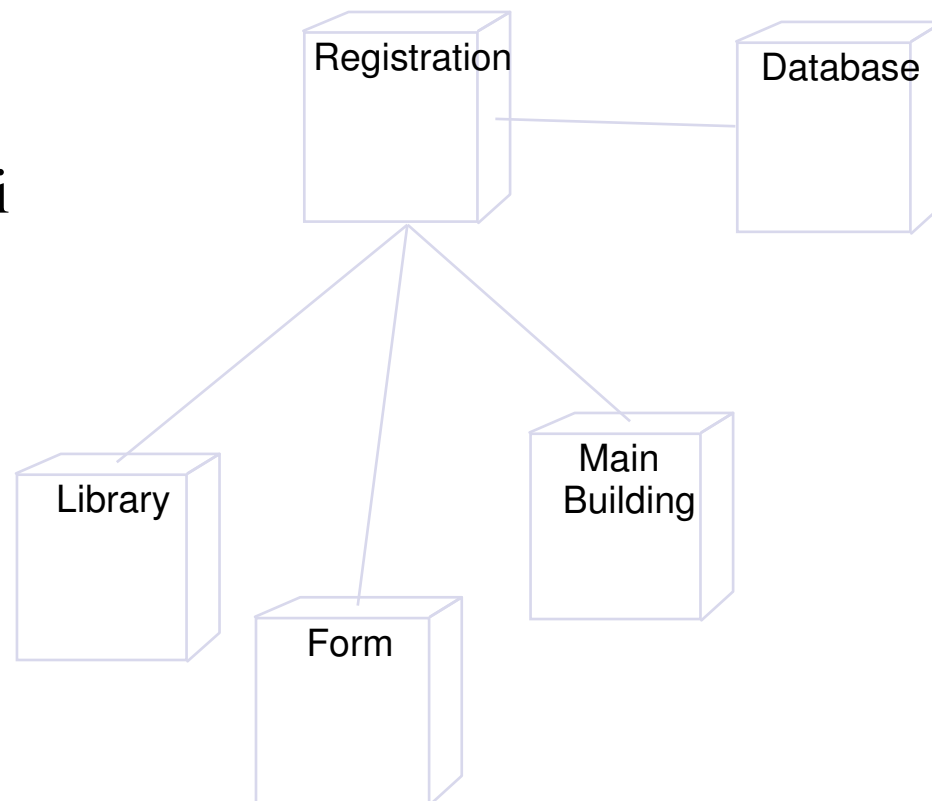
Component Diagram

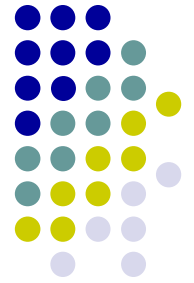
- Component diagrams định nghĩa các thành phần của hệ thống và mối liên hệ giữa các thành phần đó.
- Nắm bắt kiến trúc vật lý trong thực thi
- Xây dựng như bộ phận của sự đặc tả kiến trúc
- Mục đích
 - Tổ chức mã nguồn
 - Xây dựng các modul thi hành
 - Chỉ định cơ sở dữ liệu vật lý
- Được phát triển bởi người thiết kế và lập trình



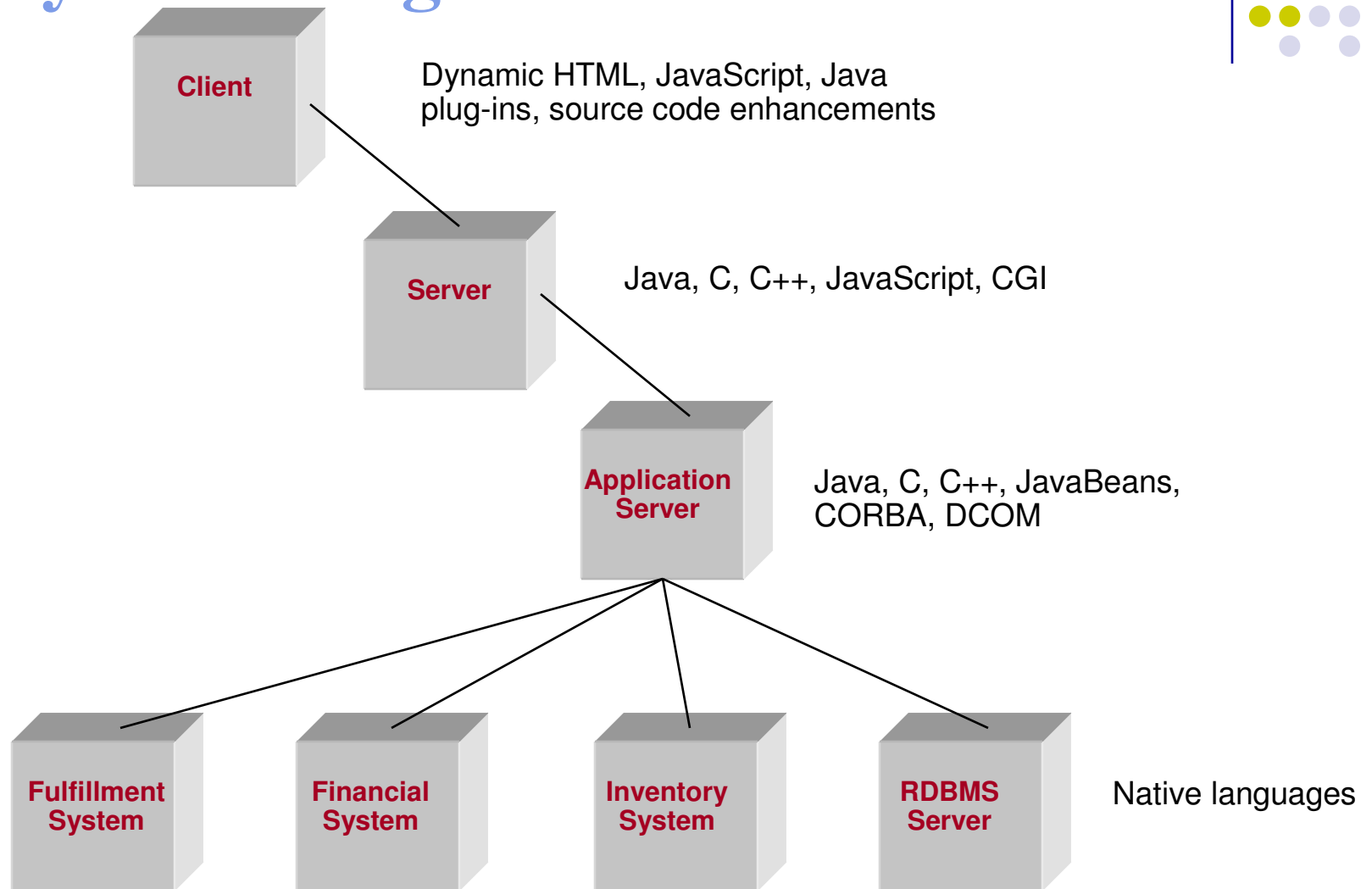
Deployment Diagram

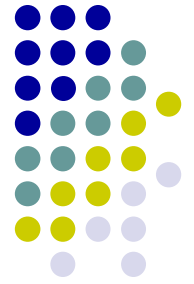
Biểu đồ triển khai biểu diễn kiến trúc cài đặt và triển khai hệ thống dưới dạng các nodes và các mối quan hệ giữa các node đó. Thông thường, các nodes được kết nối với nhau thông qua các liên kết truyền thông như các kết nối mạng, liên kết TCPIP, ...





Deployment Diagram

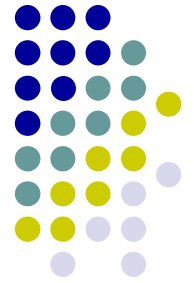




Một số phần mềm hỗ trợ UML

- Rational Rose
 - Là ngôn ngữ chuẩn phù hợp với các lược đồ trong UML
 - Phù hợp với sự kế thừa của các hệ thống hướng đối tượng
 - Các mô hình duy trì sự đồng bộ hóa trong suốt vòng đời phát triển
 - ...
- Paradigm
- Altova UModel
- StartUML
- ...

Tóm tắt



- Các vấn đề đã nghiên cứu
 - Khái niệm mô hình, mô hình hóa trực quan
 - Khái quát về Ngôn ngữ mô hình hóa thống nhất
 - Thí dụ sử dụng các biểu đồ của UML

