



ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN
Vietnam - Korea University of Information and Communication Technology

PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG PHẦN MỀM

Lê Viết Trương
Khoa Khoa học máy tính

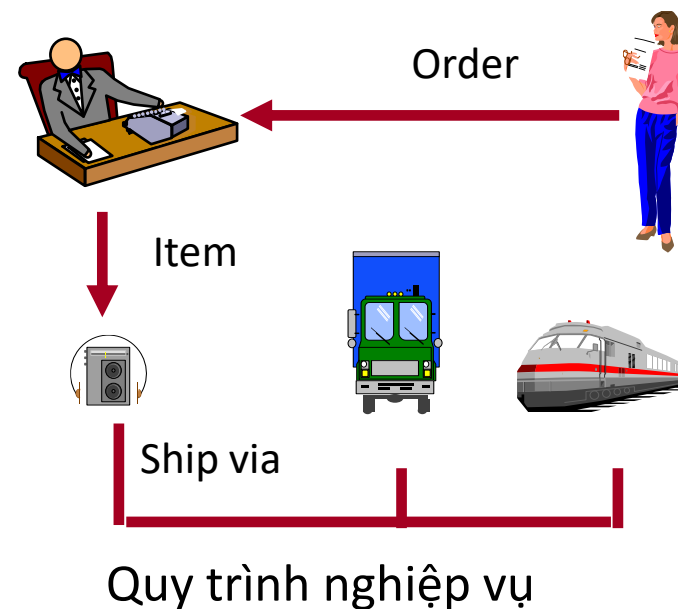
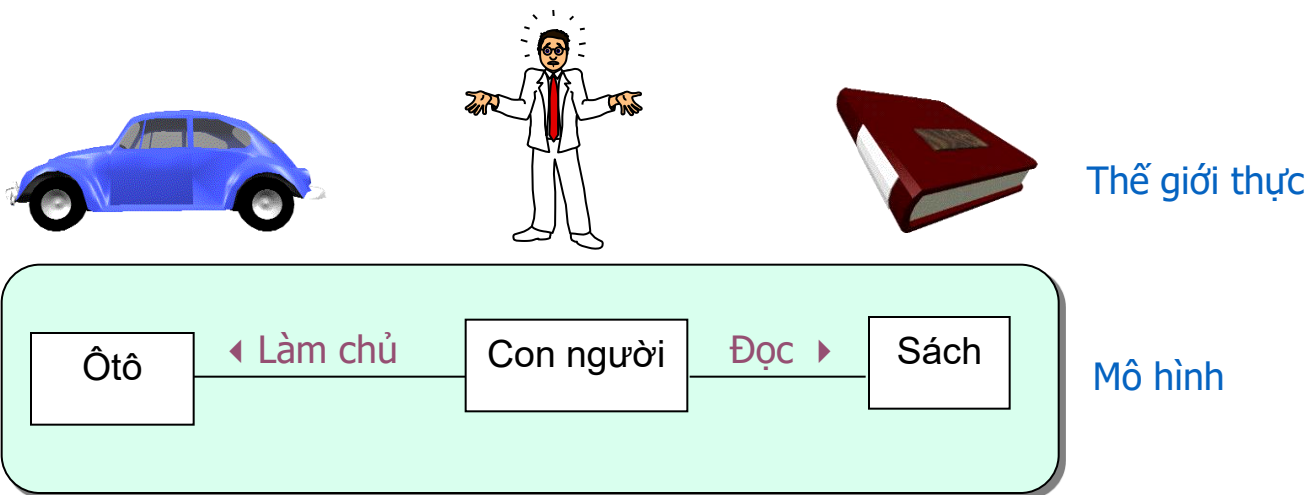


Tổng quan về ngôn ngữ mô hình hóa thống nhất

- Mô hình hóa
- Kỹ thuật mô hình hướng đối tượng
- Lịch sử của UML
- Giới thiệu ngắn gọn về UML
 - Khái niệm
 - Biểu đồ
 - Khung nhìn

Mô hình và Mô hình hóa

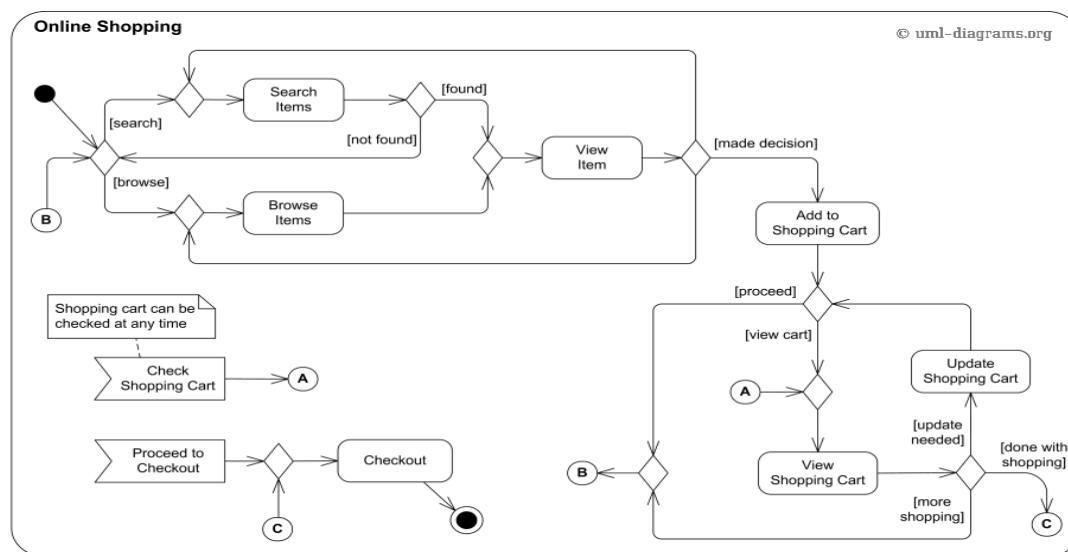
- Một **mô hình** là một sự đơn giản hóa thực tế. Chúng tôi xây dựng các mô hình để chúng tôi có thể hiểu rõ hơn về hệ thống mà chúng tôi đang phát triển.
- **Mô hình hóa** là quá trình xây dựng các mô hình để đại diện cho một hệ thống



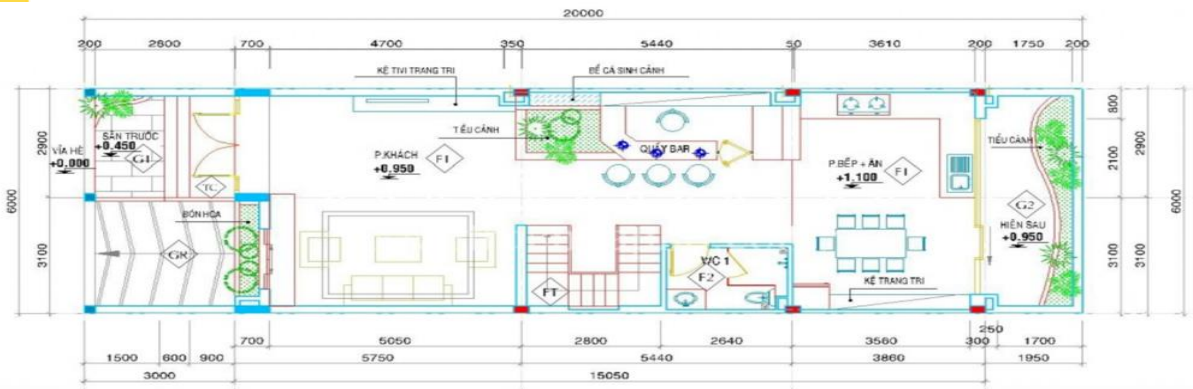
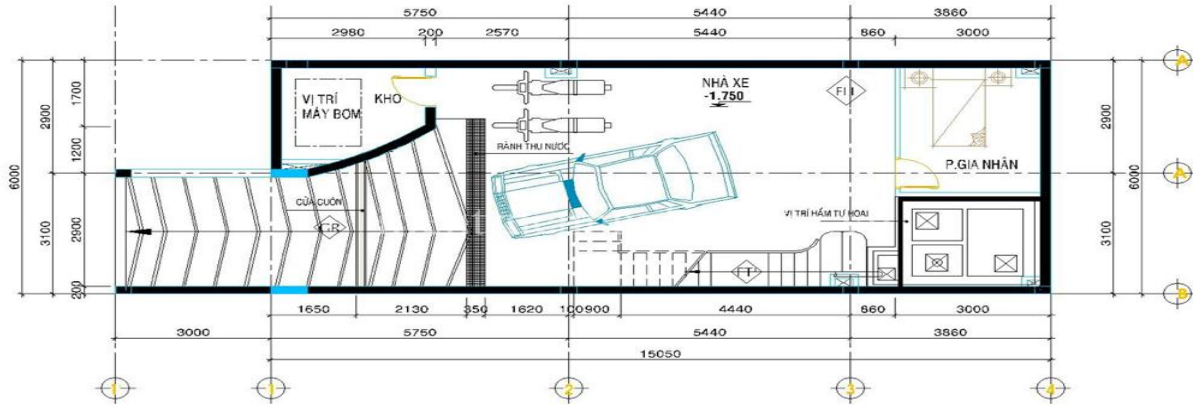
Mô hình và Mô hình hóa

- Mô hình hóa

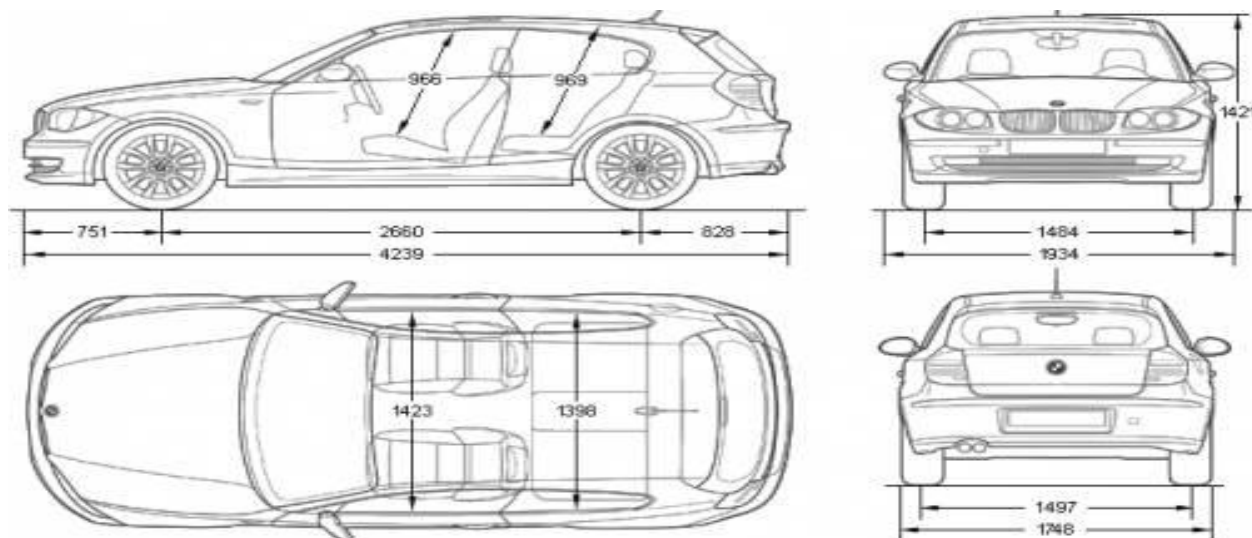
- giúp chúng tôi hình dung một hệ thống như nó vốn có hoặc như chúng tôi muốn
- cho phép chúng tôi chỉ định cấu trúc hoặc hành vi của một hệ thống
- cung cấp cho chúng tôi một mẫu hướng dẫn chúng tôi xây dựng một hệ thống
- ghi lại quyết định chúng tôi đã đưa ra



Mô hình và Mô hình hóa : Ví dụ



Mô hình và Mô hình hóa : Ví dụ



BMW 116i Sport

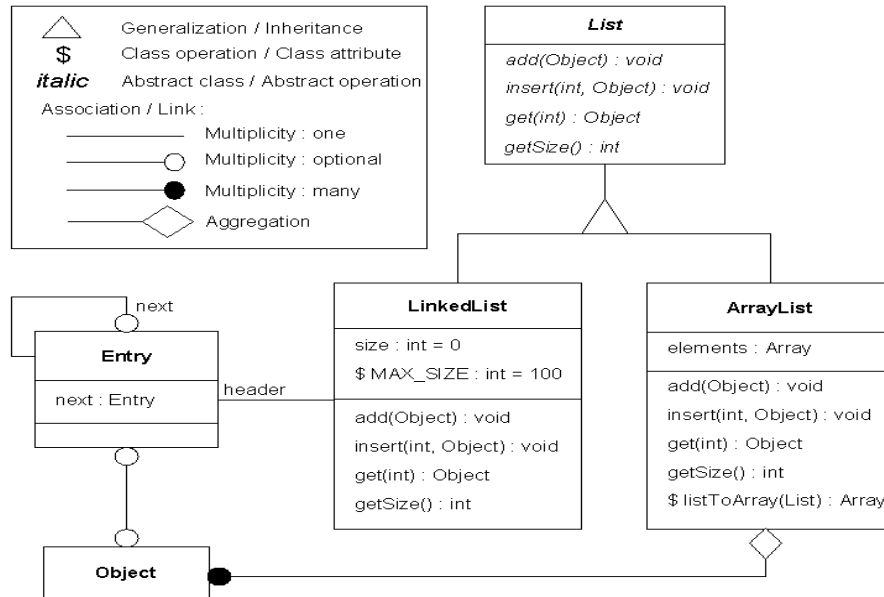


Kỹ thuật mô hình hướng đối tượng

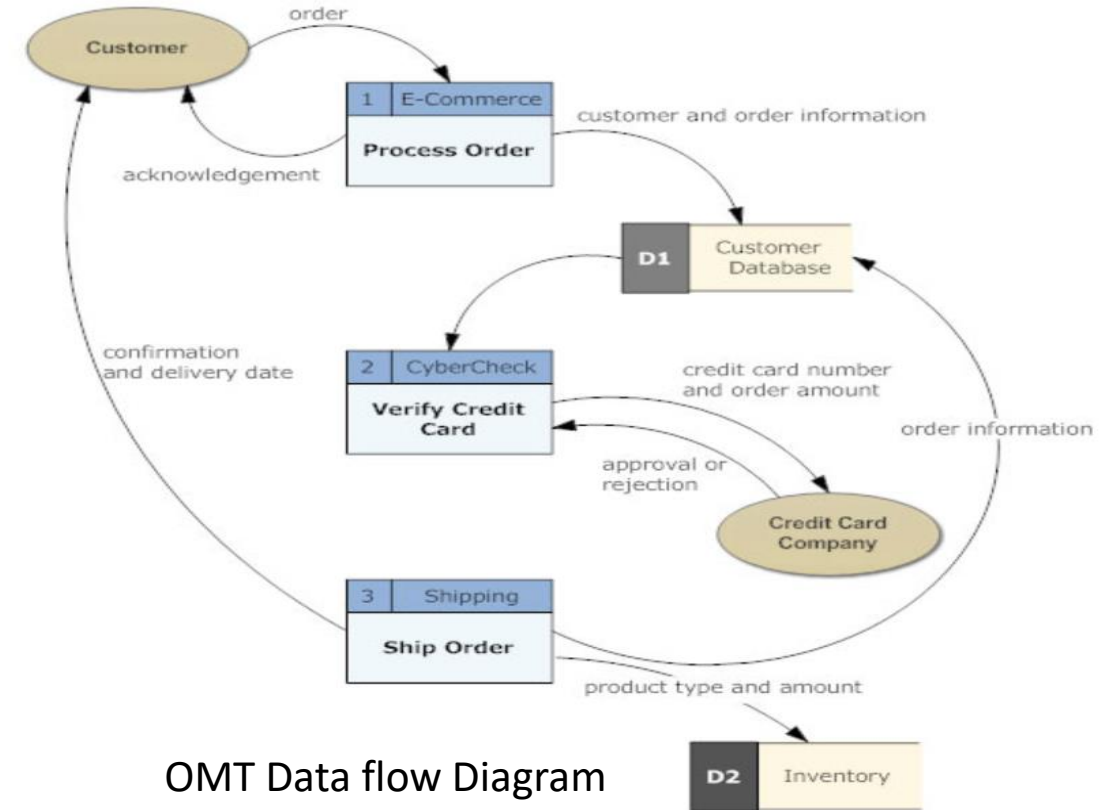
- Các kỹ thuật mô hình hóa hướng đối tượng là các quy trình / phương pháp luận / cách tiếp cận để mô hình hóa và thiết kế phần mềm
 - 1975 - 1990: một số kỹ thuật hướng đối tượng được phát triển
 - 1990 - 1994: có hơn 50 kỹ thuật mô hình hướng đối tượng
- Các kỹ thuật nổi tiếng nhất
 - OOD (Object-Oriented Design)
 - OOSE (Object-Oriented Software Engineering)
 - OMT (Object Modelling Technique)

Kỹ thuật OMT

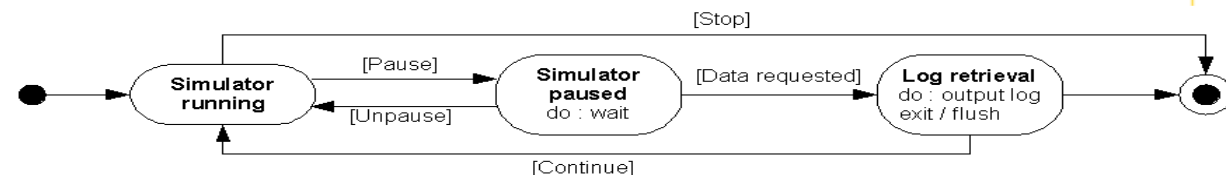
- Được phát triển bởi Jim Rumbaugh (1991)
- Gồm 3 loại mô hình chính
 - Mô hình đối tượng: Biểu đồ đối tượng
 - Mô hình động: Biểu đồ trạng thái
 - Mô hình chức năng: Biểu đồ luồng dữ liệu



OMT Object Diagram



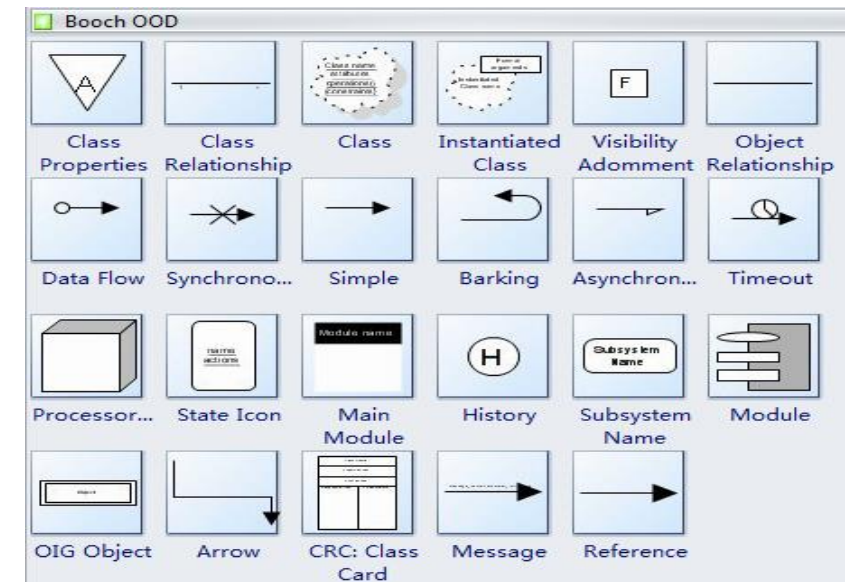
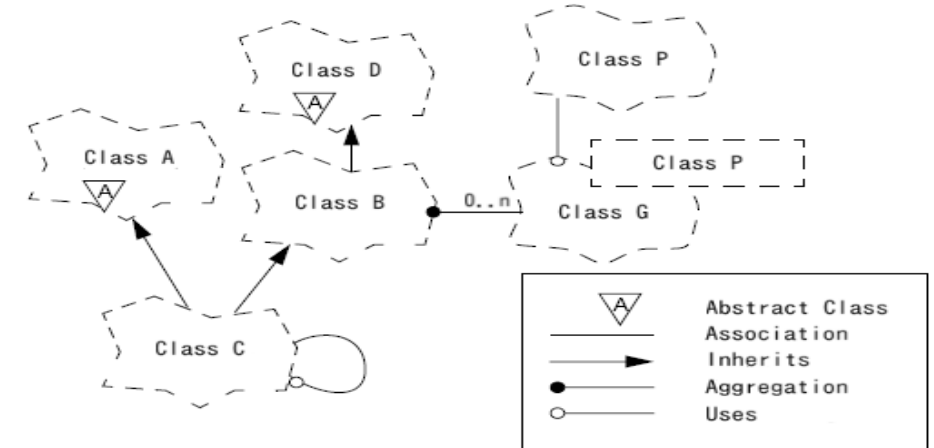
OMT Data flow Diagram



OMT State Diagram

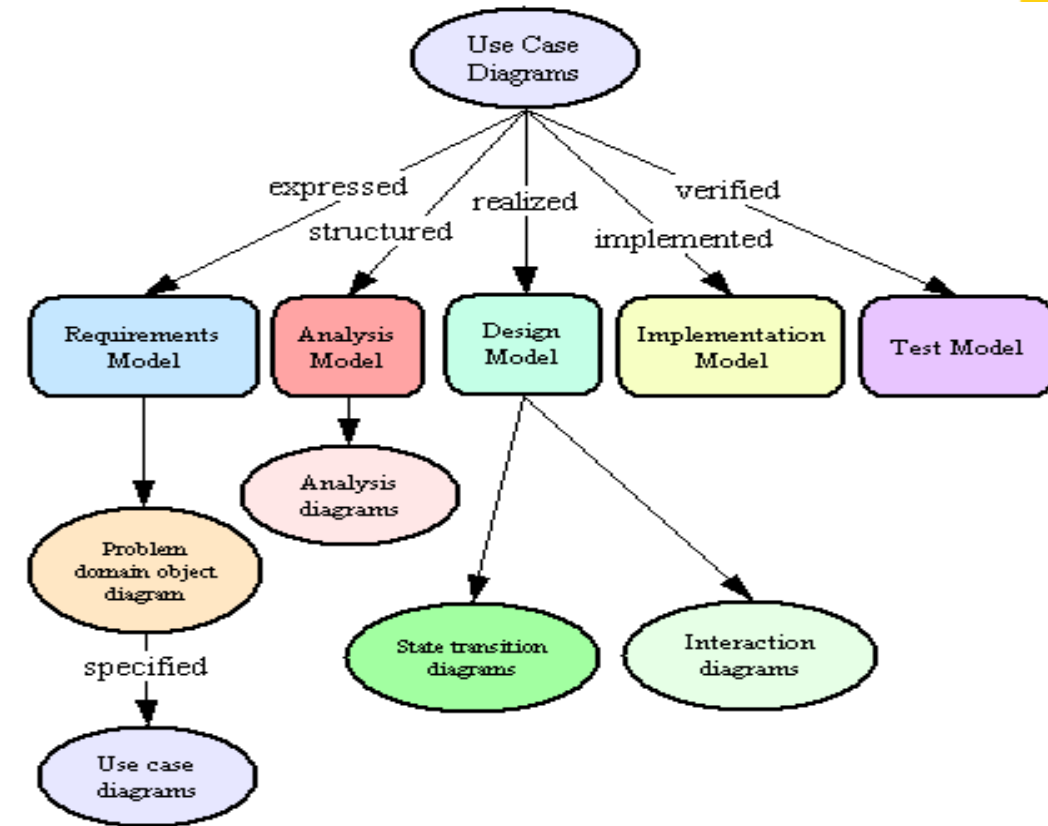
Kỹ thuật OOD

- Được phát triển bởi Grady Booch (1991)
- Bao gồm:
- Khung nhìn tĩnh
 - Biểu đồ lớp
 - Biểu đồ đối tượng
 - Biểu đồ mô-đun
- Khung nhìn động
 - Biểu đồ chuyển trạng thái
 - Biểu đồ quá trình
 - Biểu đồ tương tác



Kỹ thuật OOSE

- Được phát triển bởi Ivar Jacobson (1992)
- Bao gồm 5 mô hình
 - Mô hình yêu cầu: Biểu đồ miền vấn đề, Biểu đồ ca sử dụng
 - Mô hình phân tích: Biểu đồ phân tích
 - Mô hình thiết kế: Biểu đồ chuyển trạng thái, Biểu đồ tương tác
 - Mô hình triển khai
 - Mô hình thử nghiệm

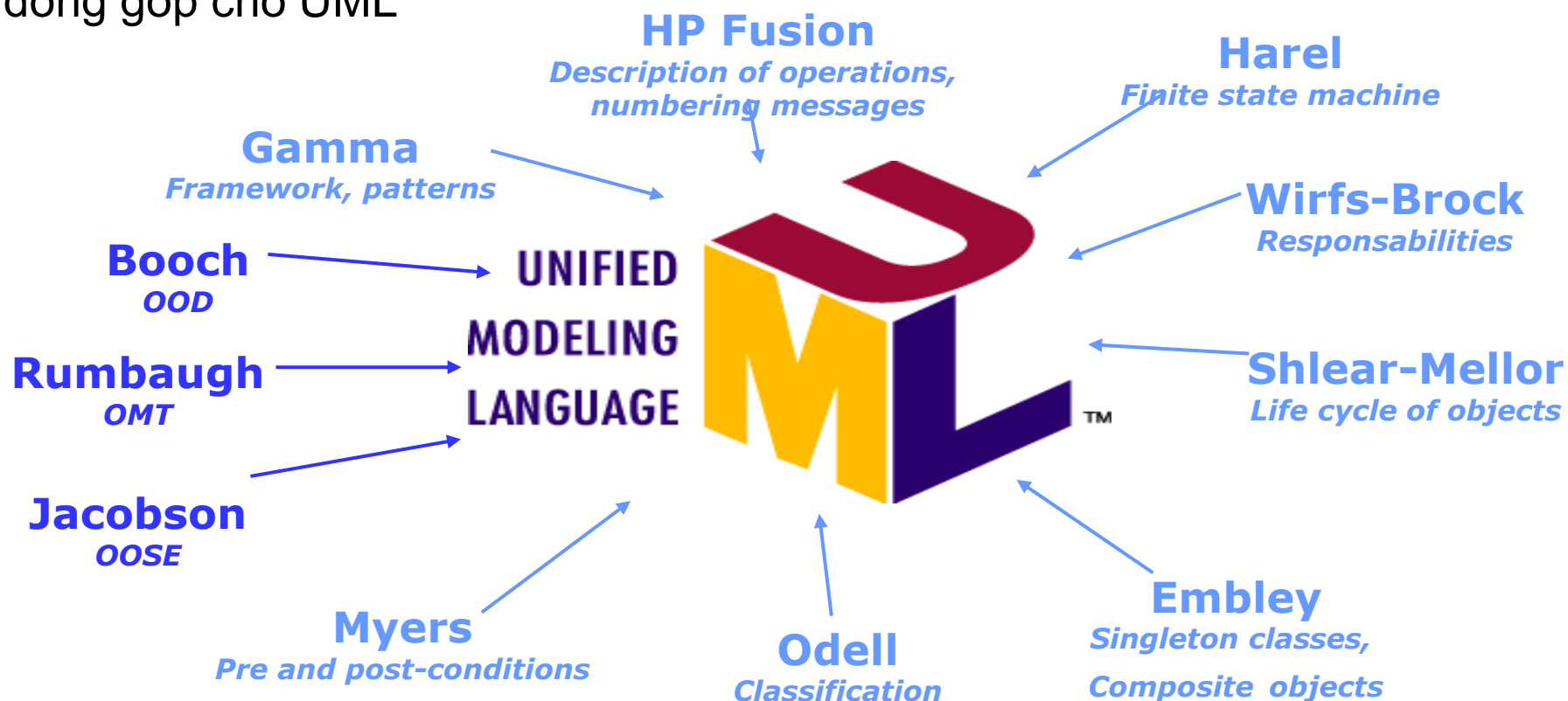


Lịch sử của UML

- Quá nhiều kỹ thuật mô hình hóa hướng đối tượng
 - Cần tiêu chuẩn hóa
 - Hợp nhất các kỹ thuật mô hình hóa
- Năm 1994: Rumbaugh và Booch thống nhất các phương pháp tiếp cận của họ cho dự án UML tại Rational Software
- Năm 1995: Phiên bản đầu tiên được phát hành dưới tên “Unified Method” v0.8
- Năm 1996: Jacobson tham gia đội
- Năm 1997: Sự ra đời của UML v0.9 tích hợp OOSE. Hội nghị đầu tiên của UML được tổ chức
- Năm 2005, UML 2.0 được phát hành
 - Biểu đồ mới, cải tiến biểu đồ hiện có
- 09/2013, UML v.2.5 RTF - Beta 2
- 06/2015, UML v.2.5

Lịch sử của UML

- Những đóng góp cho UML



Giới thiệu về UML

- **UML** (Ngôn ngữ mô hình thống nhất) là một ngôn ngữ mô hình hóa
 - bao gồm từ vựng, cú pháp và ngữ nghĩa
 - cho phép biểu diễn một hệ thống ở các cấp độ khác nhau: khái niệm, vật lý
 - bao gồm từ vựng và quy tắc để mô tả các mô hình khác nhau đại diện cho một hệ thống
- **UML**
 - không phải là một phương pháp luận cũng không phải là một quy trình
 - cho phép tự do thiết kế
 - có thể được kết hợp với một số quy trình phát triển



Giới thiệu về UML

- UML là ngôn ngữ trực quan hóa
 - sử dụng biểu diễn đồ họa
 - cung cấp cái nhìn tốt hơn về hệ thống (nhờ các biểu diễn đồ họa)
- UML là một ngôn ngữ đặc tả
 - cho phép xác định một hệ thống mà không mơ hồ
 - cho phép xác định một hệ thống ở các giai đoạn khác nhau: phân tích, thiết kế, triển khai
- UML là một ngôn ngữ xây dựng
 - cho phép mô phỏng hệ thống
 - Các mô hình UML dễ dàng chuyển đổi thành mã nguồn
- UML là một ngôn ngữ tài liệu
 - cho phép mô tả tất cả các giai đoạn phát triển của hệ thống
 - Các mô hình đã xây dựng là tài liệu hoàn chỉnh của hệ thống



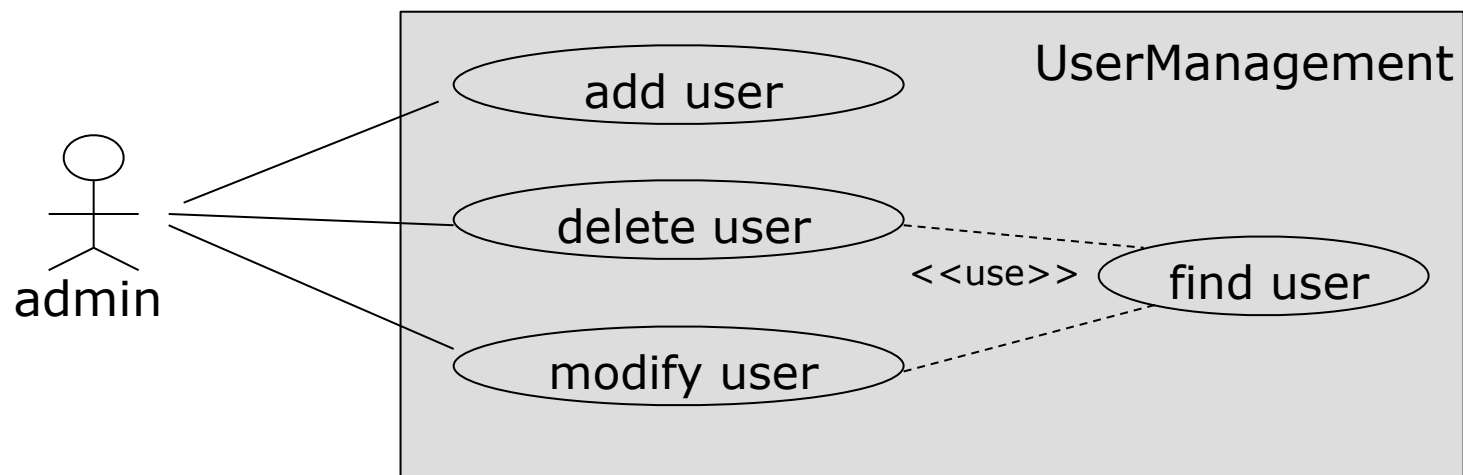
Giới thiệu về UML: Biểu đồ



- Bao gồm 10 biểu đồ chính
 - **Mô hình hóa yêu cầu**
 - Biểu đồ ca sử dụng
 - **Mô hình hóa cấu trúc tĩnh**
 - Biểu đồ lớp
 - Biểu đồ đối tượng
 - **Mô hình hóa hành vi động**
 - Biểu đồ tương tác
 - Biểu đồ tuần tự
 - Biểu đồ giao tiếp
 - Biểu đồ hoạt động
 - Biểu đồ trạng thái
 - **Mô hình hóa kiến trúc**
 - Biểu đồ gói
 - Biểu đồ thành phần
 - Biểu đồ triển khai

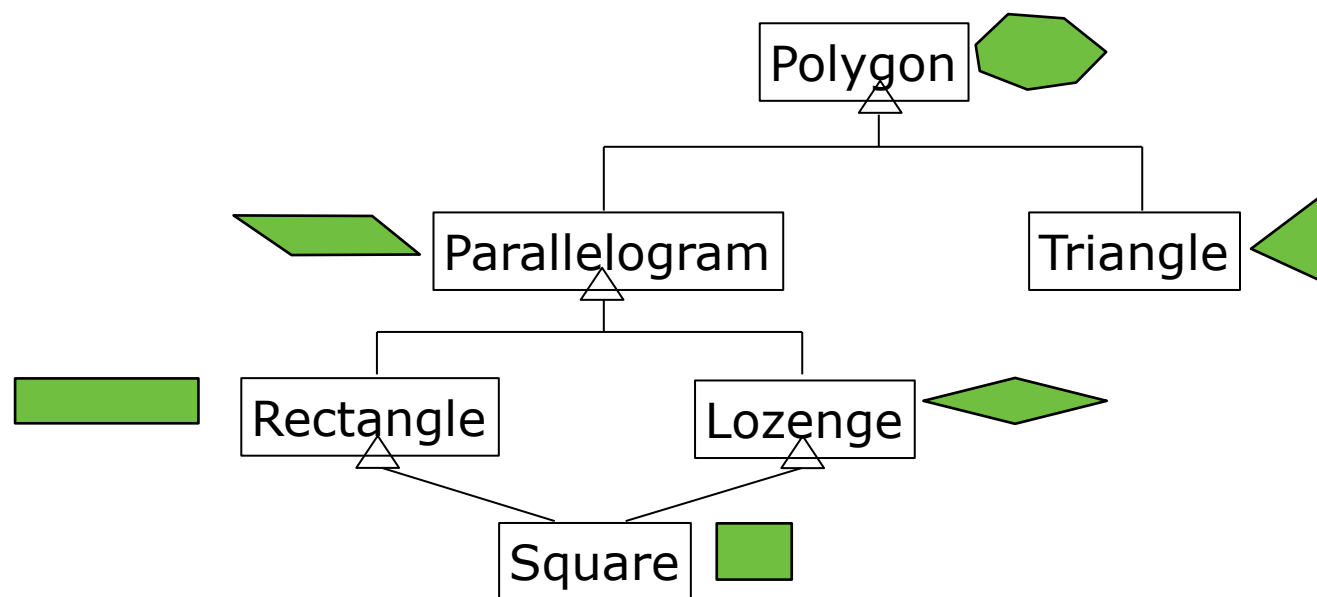
Giới thiệu về UML: Biểu đồ ca sử dụng

- Hiện thị các cách sử dụng có thể có của một hệ thống
- Mô tả **khung nhìn tĩnh/ static view** của hệ thống theo góc nhìn của người dùng
- Rất quan trọng để hiểu các chức năng của hệ thống
- Ví dụ:



Giới thiệu về UML: Biểu đồ lớp

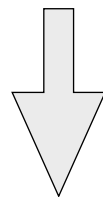
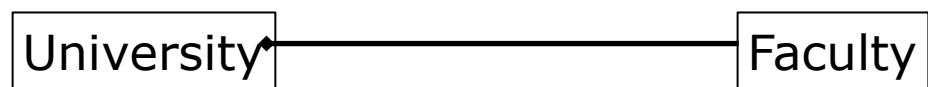
- Mô tả các lớp và mối quan hệ của chúng
- Mô tả **khung nhìn tĩnh** của hệ thống
- Ví dụ:



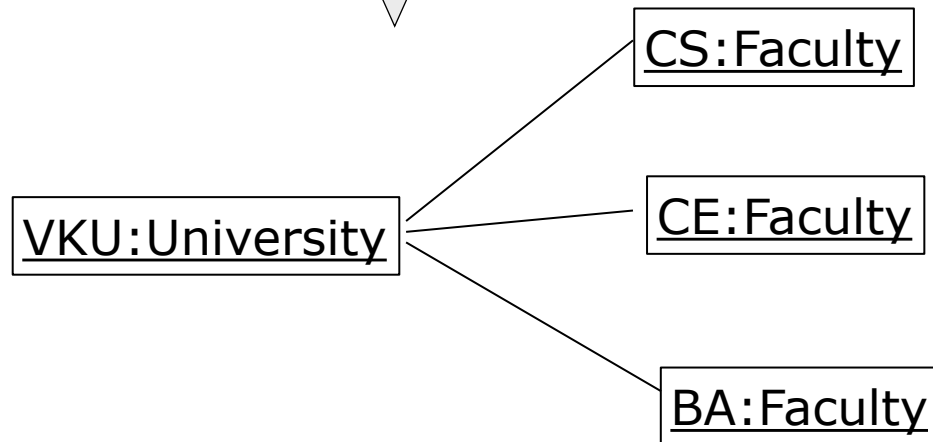
Giới thiệu về UML: Biểu đồ đối tượng

- Mô tả một tập hợp các đối tượng và mối quan hệ của chúng
- Biểu đồ đối tượng biểu diễn cho cùng một thông tin với biểu đồ lớp nhưng ở mức thể hiện của các lớp
- Mô tả **khung nhìn tĩnh** của hệ thống
- Ví dụ

Biểu đồ lớp



Biểu đồ đối tượng





Giới thiệu về UML: Biểu đồ tương tác

- Mô tả các hành vi của hệ thống bằng các tương tác giữa các đối tượng hợp thành
- Mô hình hóa **khung nhìn động** của hệ thống
- Biểu đồ tương tác là phần mở rộng của biểu đồ đối tượng bằng cách mô tả các tương tác giữa các đối tượng
- Bao gồm hai loại Biểu đồ
 - **Biểu đồ trình tự** mô tả sự tương tác giữa các đối tượng với trọng tâm là trình tự sắp xếp các thông điệp
 - **Biểu đồ giao tiếp** mô tả sự tương tác giữa các đối tượng với trọng tâm là cấu trúc của các đối tượng

Giới thiệu về UML: Biểu đồ tương tác

- Ví dụ Biểu đồ trình tự



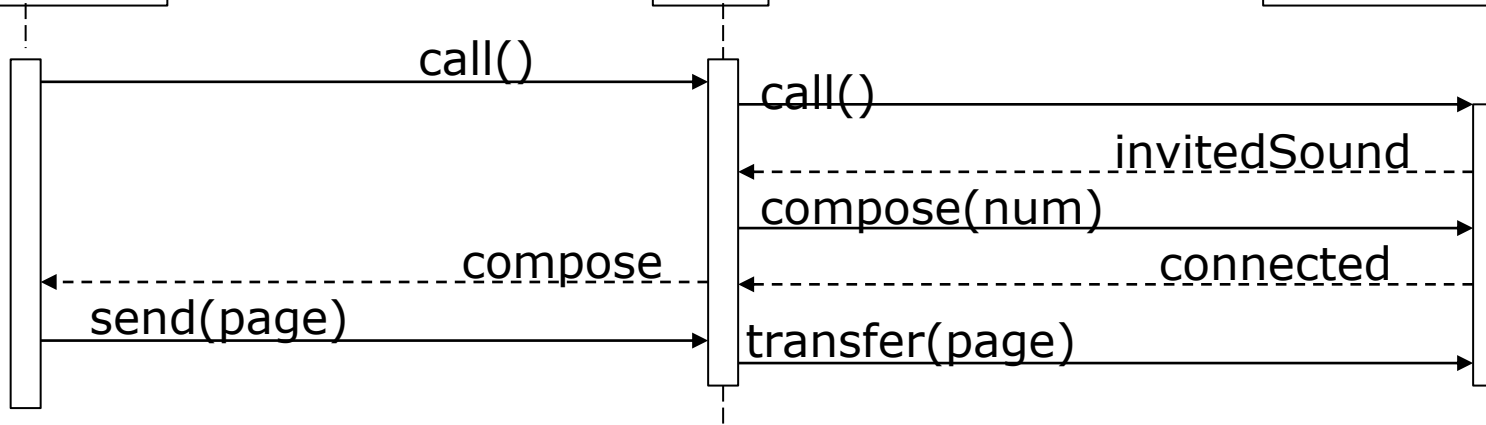
:Document



:Fax



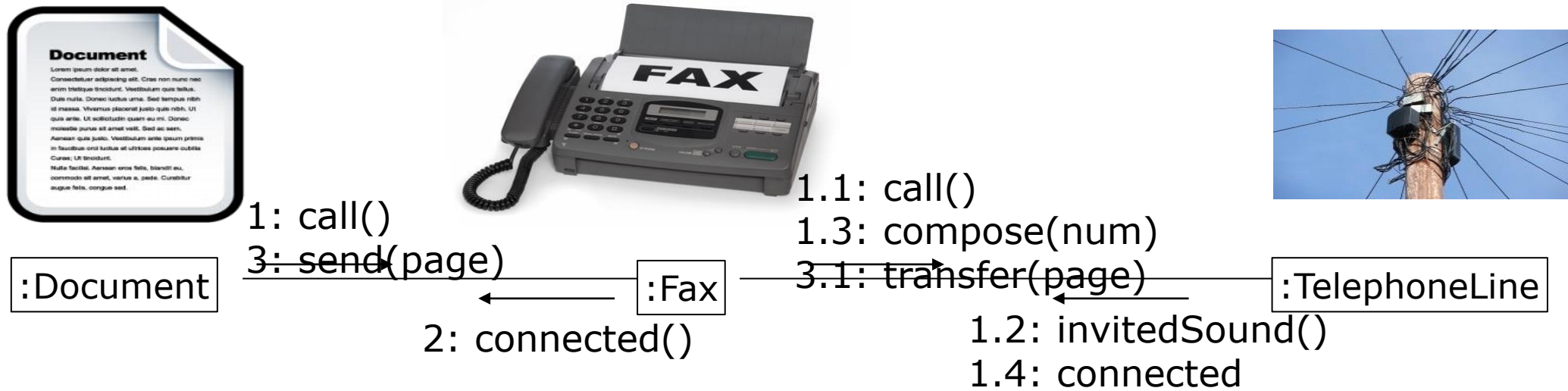
:TelephoneLine



Biểu đồ trình tự “Sending Fax”

Giới thiệu về UML: Biểu đồ tương tác

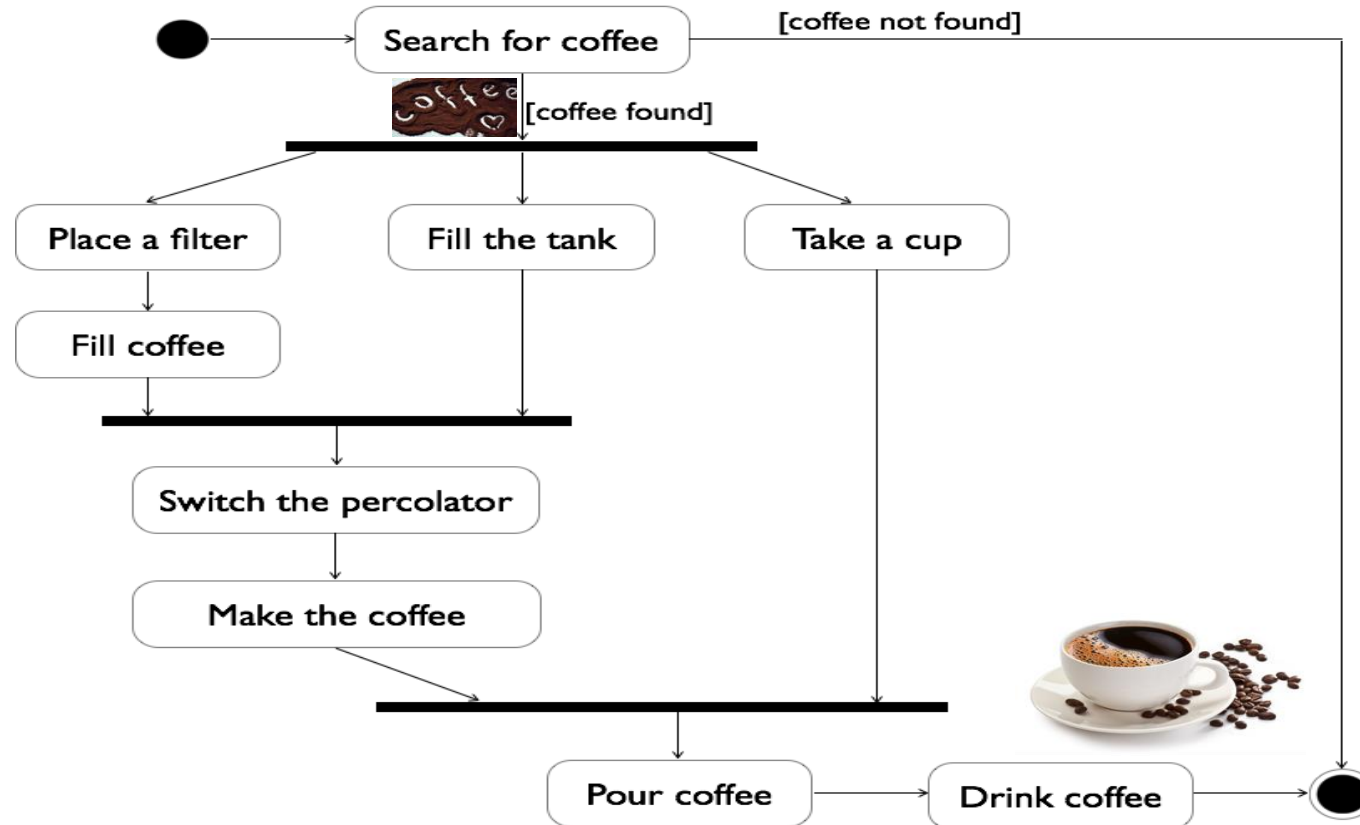
- Ví dụ Biểu đồ giao tiếp



Biểu đồ giao tiếp “Sending Fax”

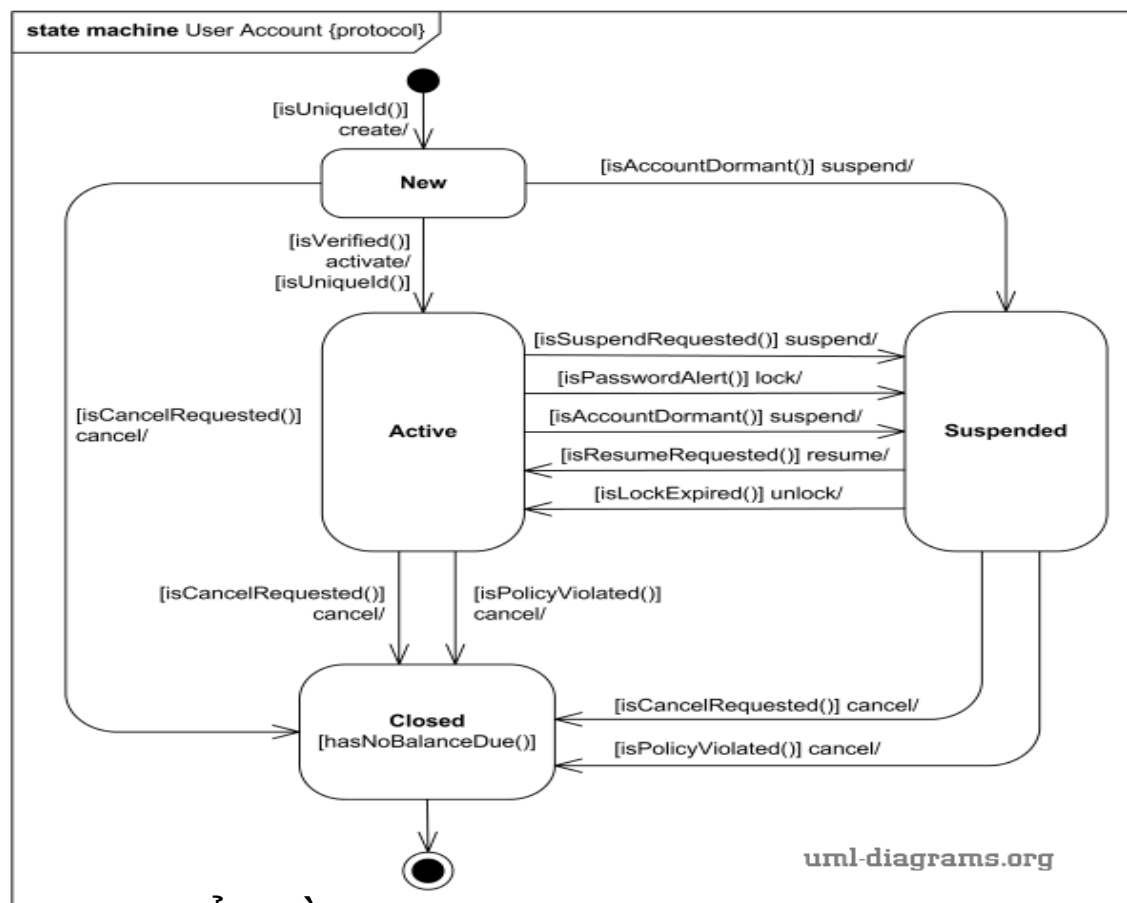
Giới thiệu về UML: Biểu đồ hoạt động

- Mô tả các luồng thông tin trong hệ thống
- Mô hình hóa **khung nhìn động** của hệ thống
- VD: Pha cà phê



Giới thiệu về UML: Biểu đồ trạng thái

- Mô tả hành vi nội bộ của hệ thống
- Mô hình hóa **khung nhìn động** của hệ thống
- Ví dụ



Biểu đồ trạng thái "Online Shopping Account"



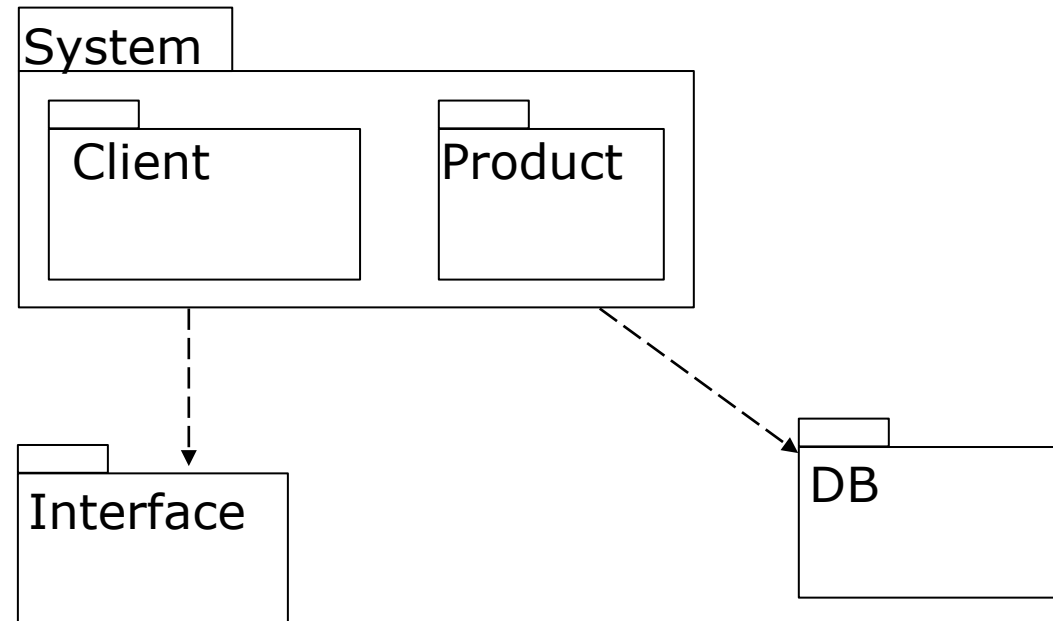
ebay™

amazon

Alibaba.com™

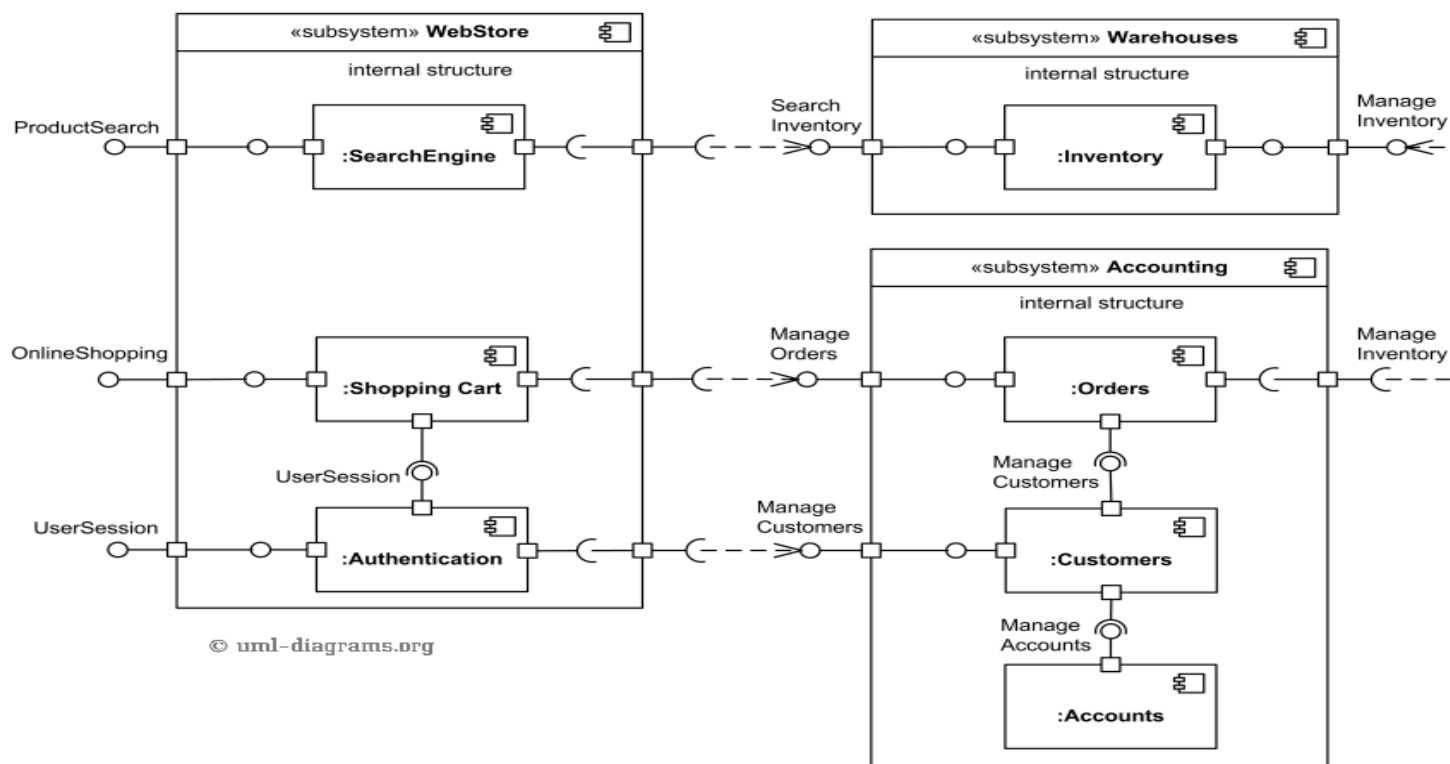
Giới thiệu về UML: Biểu đồ gói

- Gói
 - Là phần tử đa năng được sử dụng để tổ chức các lớp, hay một số nhóm khác vào trong một nhóm
- Biểu đồ gói
 - Hiện thị các gói và mối quan hệ giữa các gói.
 - Thí dụ



Giới thiệu về UML: Biểu đồ thành phần

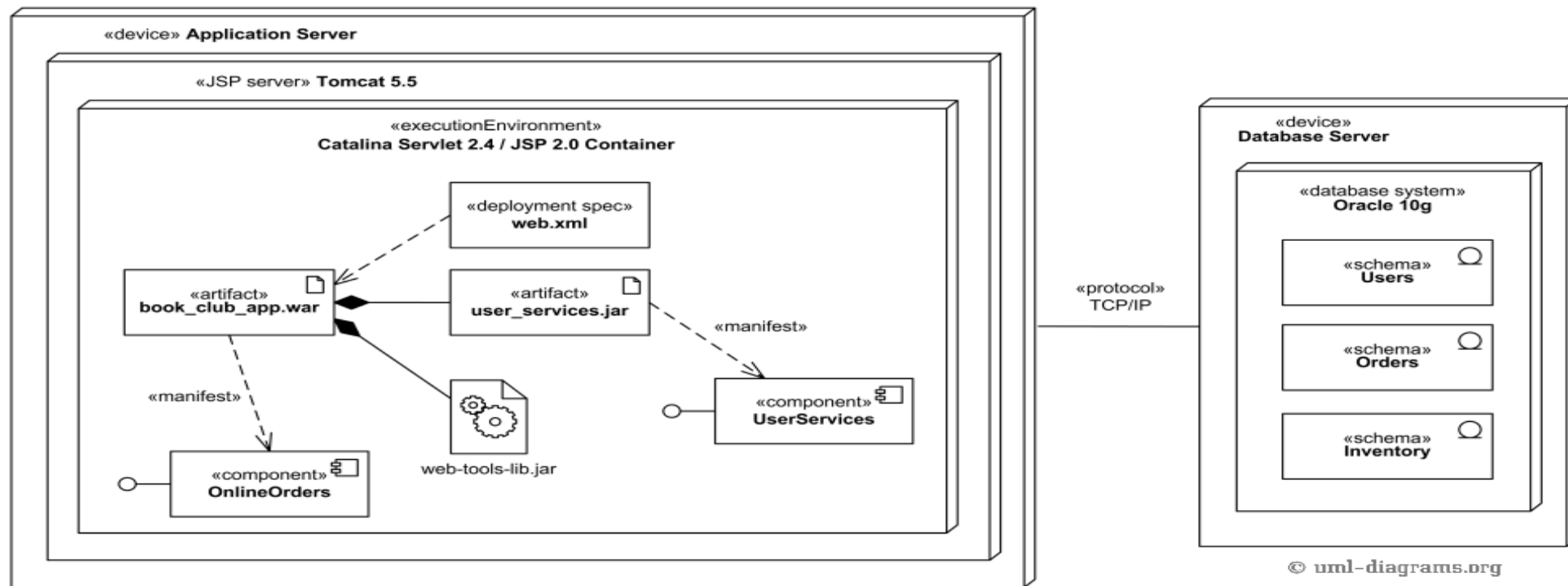
- Mô tả tổ chức các thành phần khác nhau của hệ thống
- Khung nhìn tĩnh của tổ chức hệ thống
- Ví dụ:



Biểu đồ thành phần “Online Shopping Website”

Giới thiệu về UML: Biểu đồ triển khai

- Mô tả tổ chức vật lý của các thành phần khác nhau của hệ thống



Một ví dụ về biểu đồ triển khai của ứng dụng web JEE



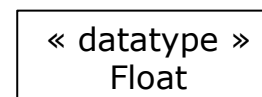
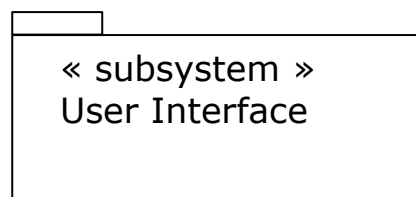
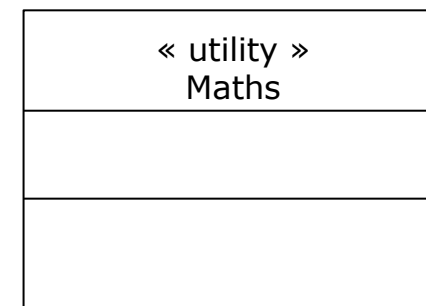
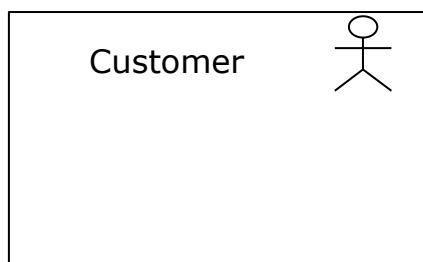
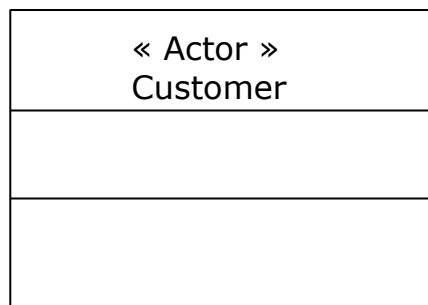
Giới thiệu về UML: cơ chế mở rộng

- Cơ chế mở rộng tích hợp
 - Khuôn mẫu/Stereotypes
 - Giá trị được gắn thẻ
- Ghi chú
- Ràng buộc: Ngôn ngữ văn bản OCL

Giới thiệu về UML: cơ chế tổng quát

- Khuôn mẫu/Stereotype

- là một cơ chế mở rộng được tích hợp sẵn
- mở rộng từ vựng của UML
- được sử dụng để tạo các kiểu phần tử UML mới bắt nguồn từ các loại hiện có nhưng được điều chỉnh cho phù hợp với một vấn đề nhất định
- có những khuôn mẫu được xác định trước trong UML
- Ký hiệu
 - "Tên của khuôn mẫu"
 - Có thể giới thiệu một biểu tượng



Giới thiệu về UML: cơ chế tổng quát

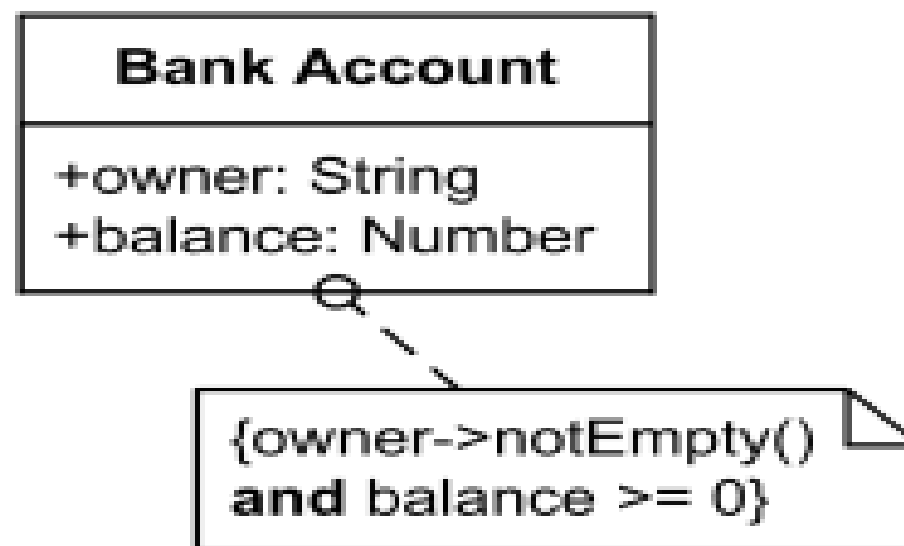
- Giá trị được gắn thẻ
 - Một cơ chế mở rộng khác
 - Cung cấp thông tin bổ sung về các phần tử của UML
 - Các cặp kiểu {tên = giá trị}
 - Thí dụ

Class {author = NTB, version = 2.0}

Giới thiệu về UML: cơ chế tổng quát

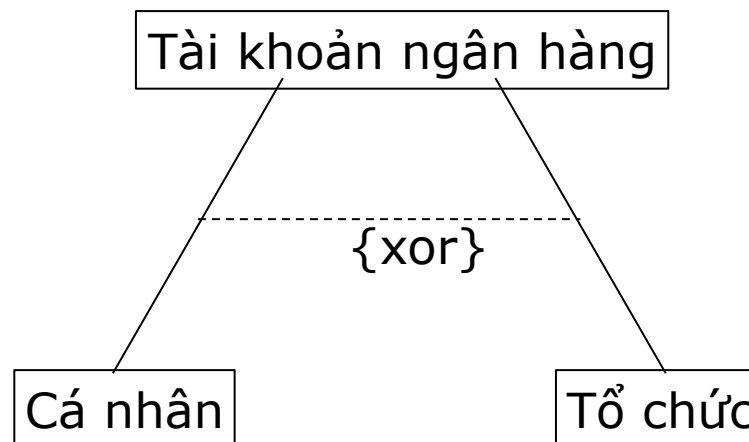
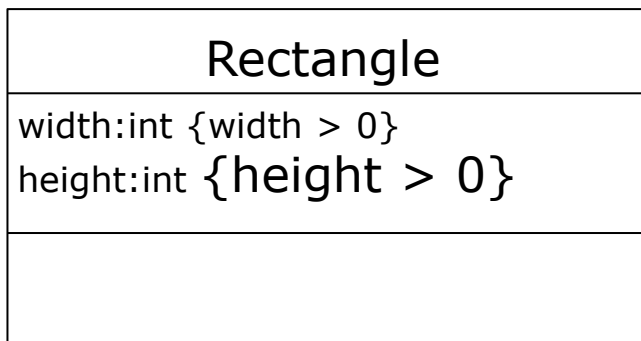
- Ghi chú

- là các nhận xét được đính kèm với một hoặc nhiều phần tử mô hình
- cung cấp thông tin bổ sung về các phần tử mô hình
- thuộc về chế độ xem, không phải các mô hình



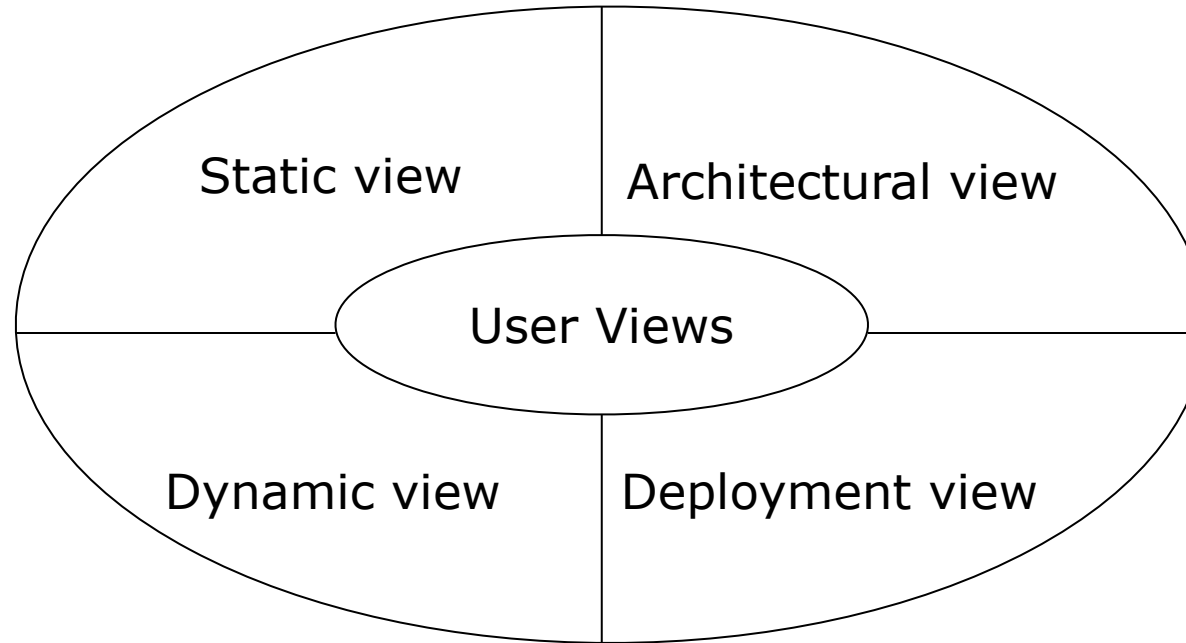
Giới thiệu về UML: cơ chế tổng quát

- Ràng buộc
- là những hạn chế giới hạn việc sử dụng một phần tử hoặc ngữ nghĩa phần tử
- được diễn đạt bằng ngôn ngữ tự nhiên
- được thể hiện bằng Ngôn ngữ ràng buộc đối tượng (Object Constraint Language)
- Thí dụ



Giới thiệu về UML: Khung nhìn/Views

- Một hệ thống được mô hình hóa bởi 5 khung nhìn khác nhau trong UML



Giới thiệu về UML: Khung nhìn

- Biểu đồ và khung nhìn

