

## Nội dung

- UML là gì?
- Lịch sử UML
- Tại sao chúng ta cần UML?
- Các mô hình ý niệm trong UML
  - Things
  - Quan hệ (Relationships)
  - Lược đồ (Diagrams)
  - Cơ chế mở rộng (Extensibility Mechanism)

## 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



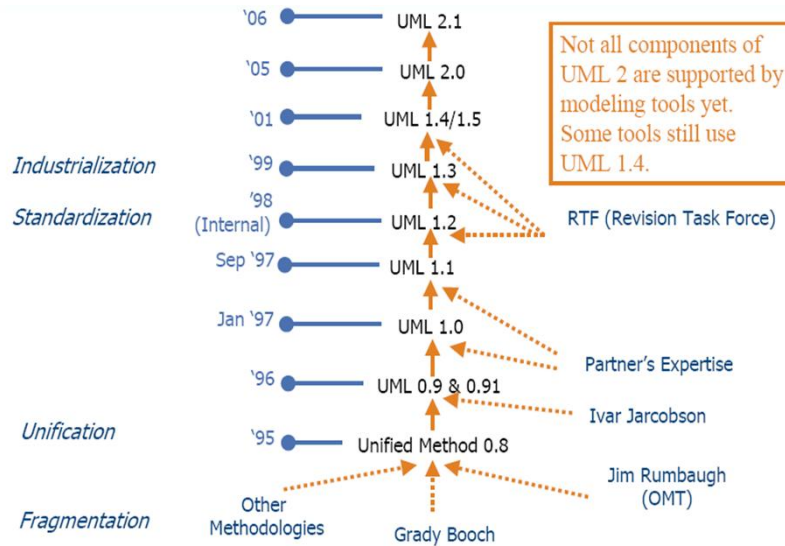
- Các mô hình UML có thể ánh xạ sang
  - Một ngôn ngữ lập trình
    - Java
    - C++
    - Visual Basic
  - Một bảng trong CSDL quan hệ (R-DBMS)
  - Một lưu trữ bền vững của CSDL HĐTg (OO-DBMS)
- Forward engineering

## 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

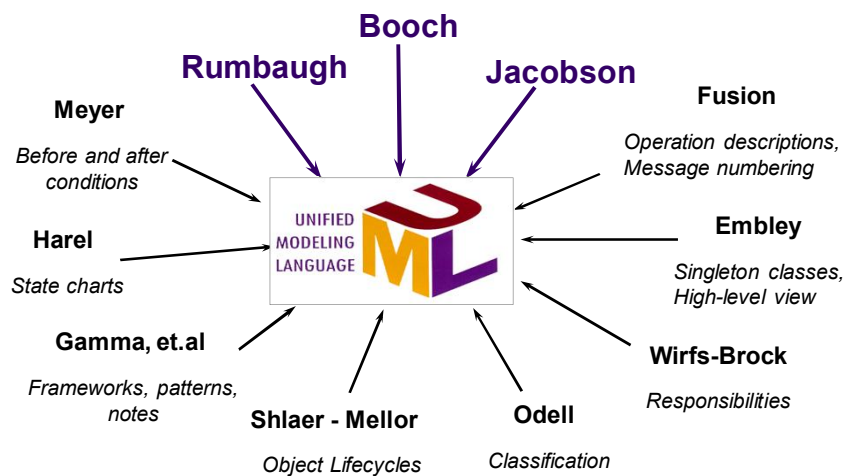
## Sơ lược về lịch sử UML



2005 - Faculty of Information Technology - Natural Sciences University HCMC

7

## Những đóng góp cho UML



2005 - Faculty of Information Technology - Natural Sciences University HCMC

8

## Tại sao chúng ta cần UML?



- Một ngôn ngữ mô hình hóa chuẩn, trực quan bằng hình ảnh
- Giúp chúng ta phát triển hệ thống hiệu quả, thiết kế chính xác và hiệu quả
- Giúp 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 chúng ta “cái nhìn tổng thể” về project

## Nội dung



- UML là gì?
- Lịch sử UML
- Tại sao chúng ta cần UML?
- Các mô hình ý niệm trong UML
  - Things
  - Quan hệ (Relationships)
  - Lược đồ (Diagrams)
  - Cơ chế mở rộng (Extensibility Mechanism)

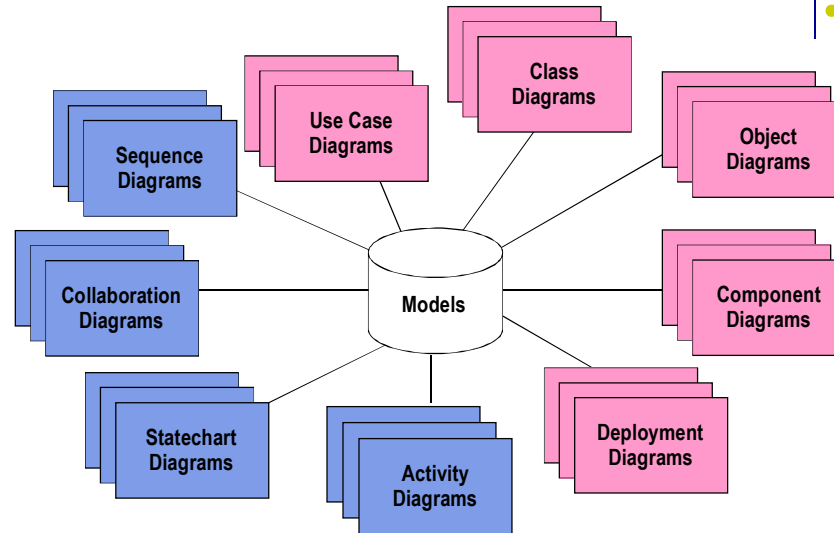
## Things

- Class
  - Interface
  - Collaboration
  - Use-case
  - Active class
  - Component
  - Node
  - Interaction
  - State machine
  - Group
  - Annotatio
- Structural Things
- Behavioral Things

## Quan hệ (Relationships)

- Quan hệ phụ thuộc (Dependency)
- Quan hệ kết hợp (Association)
- Quan hệ tổng quát hóa (Generalization)
- Quan hệ hiện thực hóa (Realization)

## Lược đồ (Diagrams)



## Cơ chế mở rộng

- Loại (Stereotype)
- Giá trị nhãn (Tagged value)
- Các ràng buộc (Constraint)