Phân tích thiết kế hướng đối tượng

Bài 2: Một số khái niệm cơ bản

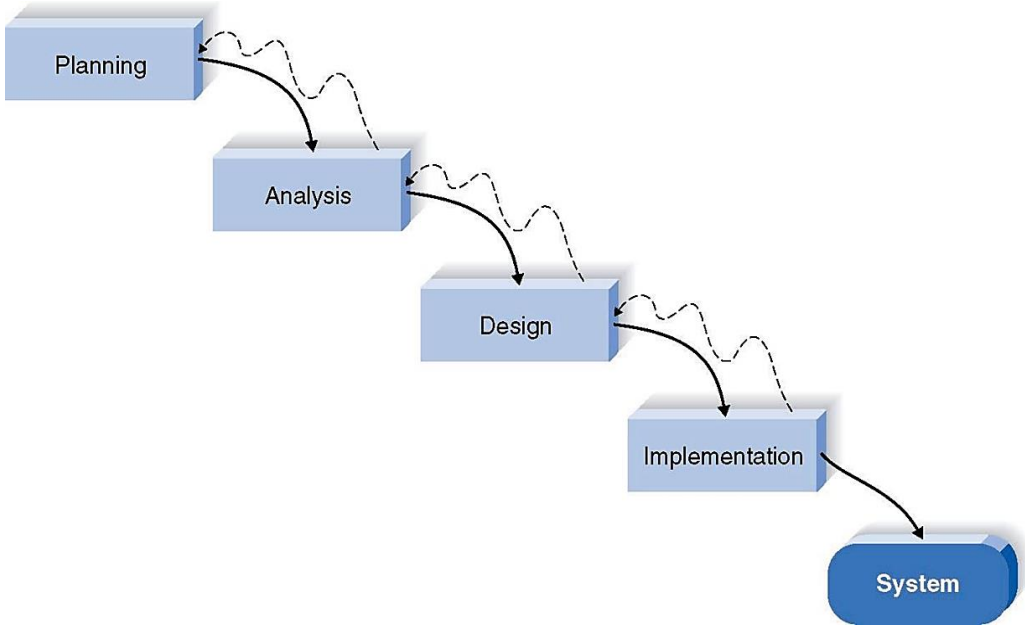
TS. Nguyễn Hiếu Cường
Bộ môn CNPM, Khoa CNTT
Trường ĐH GTVT
cuonggt@gmail.com

Phương pháp phát triển hệ thống

- Mục đích
  - Hiểu những nhu cầu, phân tích bài toán, thiết kế hệ thống
  - Xây dựng, chuyển giao, bảo trì, nâng cấp
- Hai khía cạnh trong phát triển hệ thống
  - **Quy trình** (Vòng đời phát triển – System Development Life Cycle): Các bước cần thực hiện để phát triển một hệ thống
  - **Mô hình** (Model): Các phương tiện giúp nhận thức và diễn tả hệ thống
- Các phương pháp luận (methodology) phát triển hệ thống
  - Waterfall
  - Parallel
  - RAD methods
  - ...

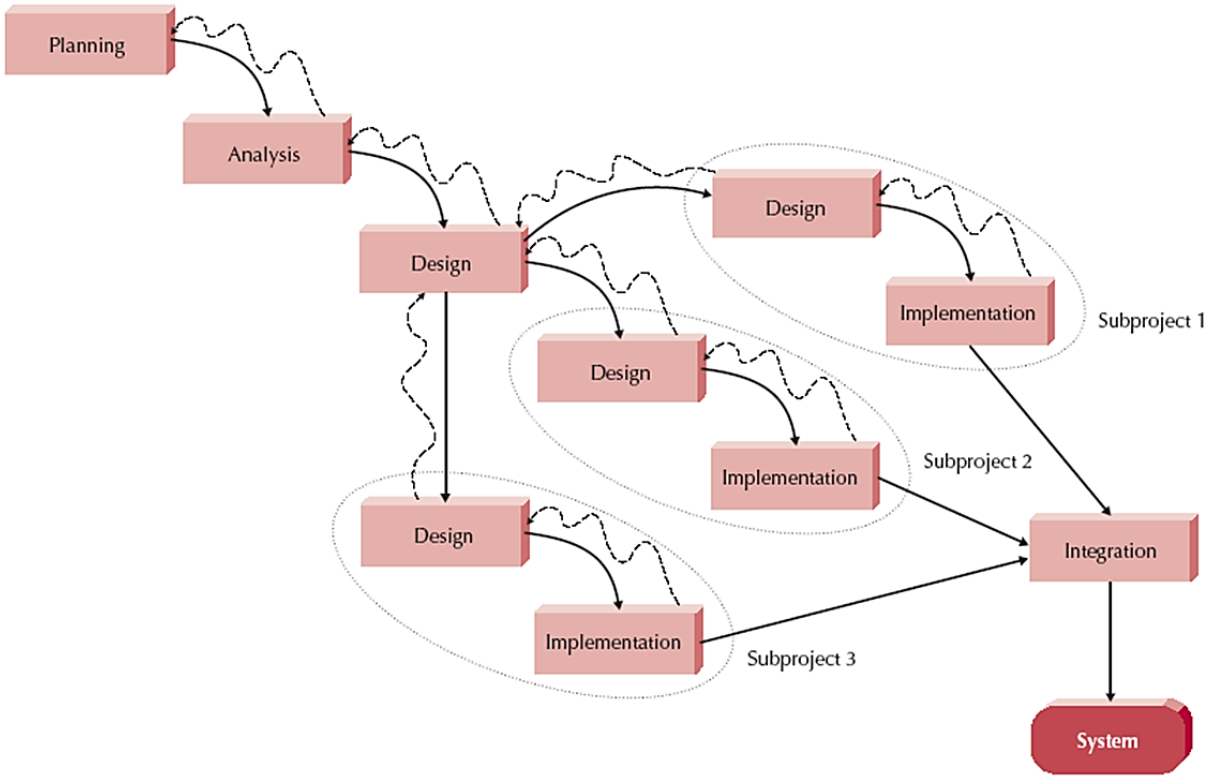
Thác nước (Waterfall)

- Ưu điểm
  - Các pha được xác định rõ
  - Hỗ trợ tốt cho lên kế hoạch
  - Phòng ngừa sớm
  - Có kết quả sau mỗi pha
- Nhược điểm
  - Phải hoàn chỉnh toàn bộ các pha trước mới sang pha sau
  - Xử lý lỗi muộn
  - Thời gian chuyển giao lâu



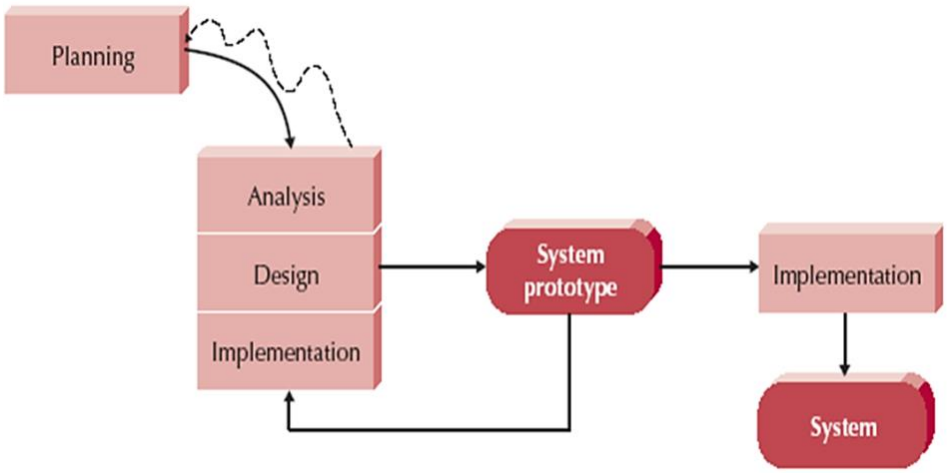
Phát triển song song

- Ưu điểm
  - Thiết kế tổng thể
  - Giảm thời gian
- Nhược điểm
  - Tích hợp các dự án con (subprojects)



Phát triển ứng dụng nhanh (RAD)

- Rapid Application Dev.
  - Nhanh, dễ giao tiếp với NSD
  - Thích hợp khi yêu cầu NSD không rõ ràng
- Làm bản mẫu (Prototyping)
- Agile development
- …

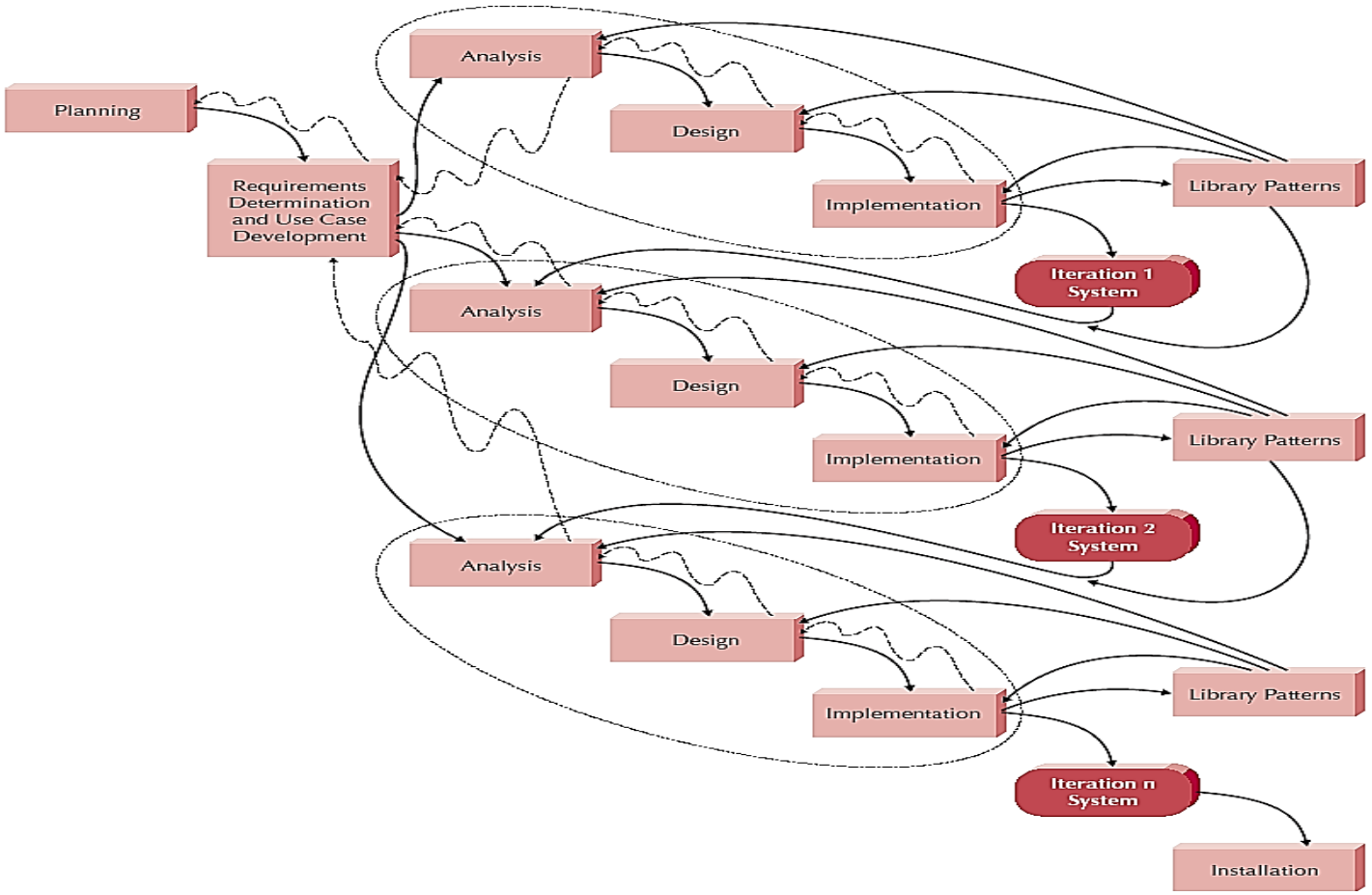


*"If I had asked people what they wanted, they would have said **faster horses**."* – Henry Ford

Lựa chọn phương pháp phát triển

Ability to Develop Systems	Structured Methodologies			RAD Methodologies		Agile Methodologies
	Waterfall	Parallel	Phased	Prototyping	Throwaway Prototyping	XP
with Unclear User Requirements	Poor	Poor	Good	Excellent	Excellent	Excellent
with Unfamiliar Technology	Poor	Poor	Good	Poor	Excellent	Poor
that are Complex	Good	Good	Good	Poor	Excellent	Poor
that are Reliable	Good	Good	Good	Poor	Excellent	Good
with a Short Time Schedule	Poor	Good	Excellent	Excellent	Good	Excellent
with Schedule Visibility	Poor	Poor	Excellent	Excellent	Good	Good

Thực tế


Planning
Requirements Determination and Use Case Development
Analysis
Design
Implementation
Library Patterns
Iteration 1 System
Analysis
Design
Implementation
Library Patterns
Iteration 2 System
Analysis
Design
Implementation
Library Patterns
Iteration n System
Installation

Phát triển hệ thống thông tin

- Hai khía cạnh của phát triển hệ thống?

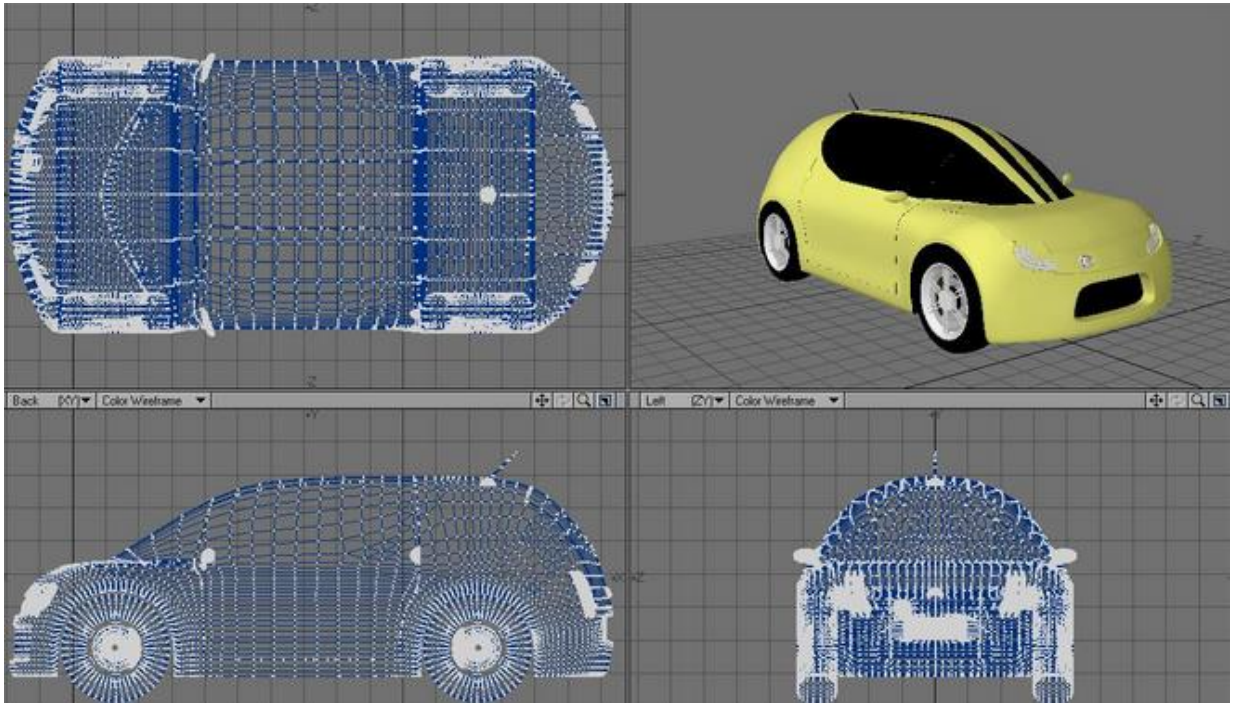
1. **Quy trình phát triển** (hoặc SDLC)
2. **Mô hình hóa**
   - Hướng chức năng
   - Hướng đối tượng

Mô hình hóa

- Mô hình
  - Một dạng trừu tượng hóa của thế giới thực
  - Mục đích: hiểu, trao đổi, hoàn chỉnh…
- Mục đích của mô hình hóa?
  - Đơn giản hóa hệ thống
  - Mô phỏng được hình ảnh *tương tự* của hệ thống
  - Sáng tỏ vấn đề
  - Tập trung vào các khía cạnh cần quan tâm

Mô hình hóa

- Mỗi đối tượng có thể có nhiều loại mô hình (các góc nhìn khác nhau)



Mô hình hóa hướng chức năng

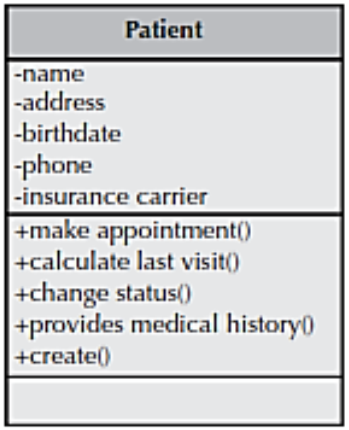
- Phân rã một vấn đề theo các chức năng
- Ưu điểm
  - Cấu trúc chức năng hệ thống rõ ràng
  - Dễ tổ chức các cơ sở dữ liệu
- Nhược điểm?
  - Không có sự gắn kết giữa dữ liệu và phương thức
  - Thay đổi một module có thể ảnh hưởng các module khác
  - Dữ liệu không được bảo mật
  - Hạn chế trong sử dụng lại và phát triển chương trình

Mô hình hóa hướng đối tượng

- Hệ thống bao gồm các đối tượng tương tác với nhau
- Ưu điểm
  - Gắn kết dữ liệu và quá trình xử lý
  - Có khả năng bảo mật dữ liệu (che giấu thông tin)
  - Tăng cường khả năng sử dụng lại và thích ứng tốt hơn với những thay đổi (có thể phát triển thêm mà ít phải sửa những phần đã có)
- Nhược điểm?
  - Nhiều khái niệm, phương pháp mô hình mới
  - Tận dụng được thế mạnh của OO không đơn giản

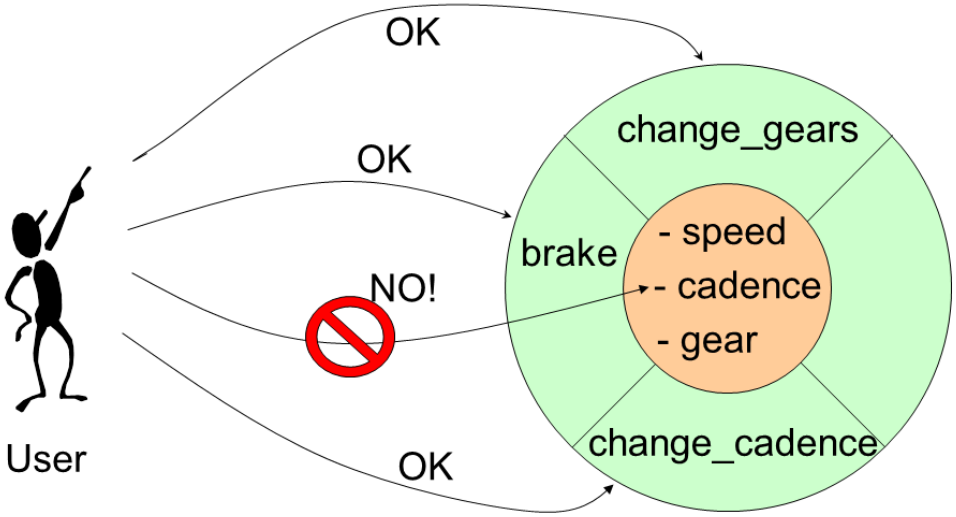
Một số khái niệm

- **Lớp** (class): Khuôn mẫu để định nghĩa các đối tượng
  - Thuộc tính (attribute)
  - Phương thức (method)
- **Đối tượng** (object): Thể hiện cụ thể (instance) của một lớp



Ví dụ về lớp

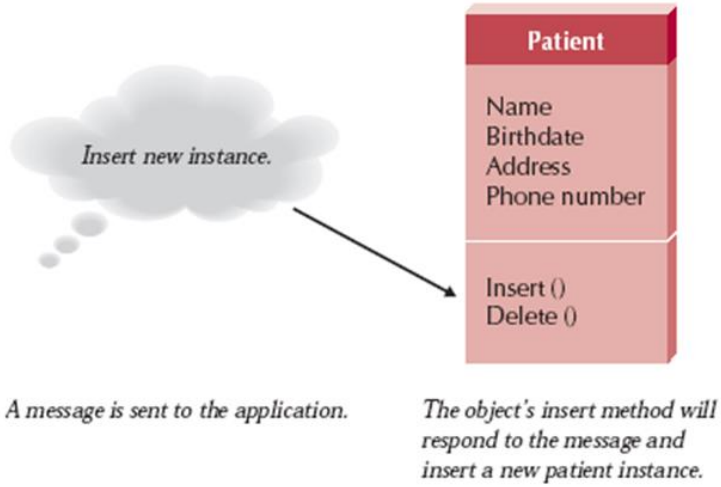
```
class Bicycle
{
private:
    float speed;
    float cadence;
    int gear;
public:
    void change_gears(int gear);
    void brake() { speed = 0; }
    void chage_cadence(float cadence);
};

Bicycle b;
b.speed = 0;    // ERROR
b.break();      // OK
```



Đối tượng và thông điệp

- Đối tượng
- **Thông điệp** (message)
  - Kích hoạt một hành động đối tượng

```
Patient aPatient;
aPatient.Insert();
```

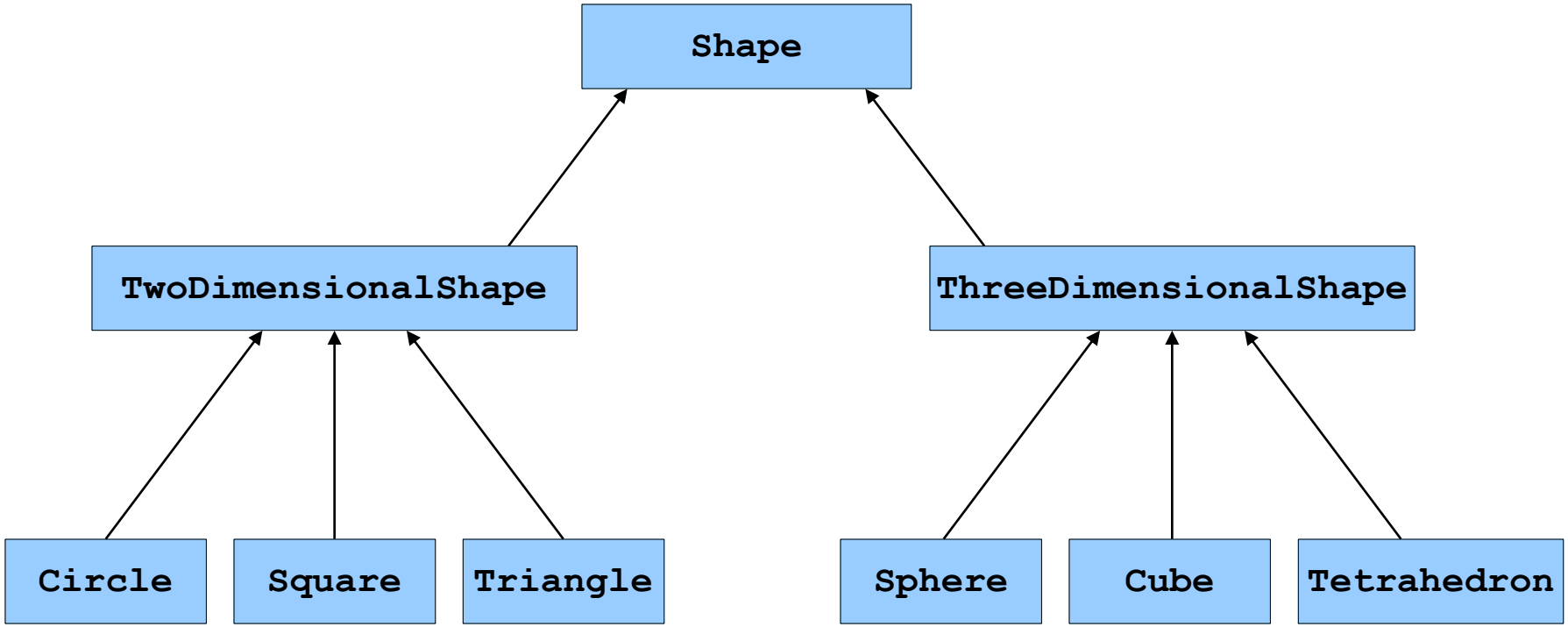


- Phương thức ≠ thông điệp?

Thừa kế

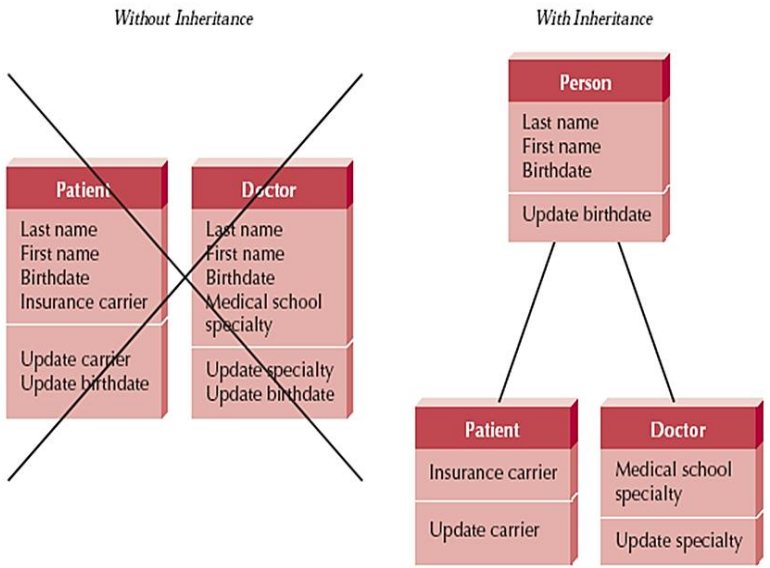
- Khái niệm thừa kế
  - Xây dựng một lớp mới (lớp dẫn xuất, lớp cụ thể) bằng cách kế thừa từ các lớp đã có (lớp cơ sở, lớp trừu tượng)
- Một lớp có thể
  - Được thừa kế từ một hoặc nhiều lớp khác
  - Là cơ sở của một hoặc nhiều lớp khác
- Những gì được kế thừa?
  - Dữ liệu
  - Các phương thức

Ví dụ


Shape
TwoDimensionalShape
ThreeDimensionalShape
Circle
Square
Triangle
Sphere
Cube
Tetrahedron

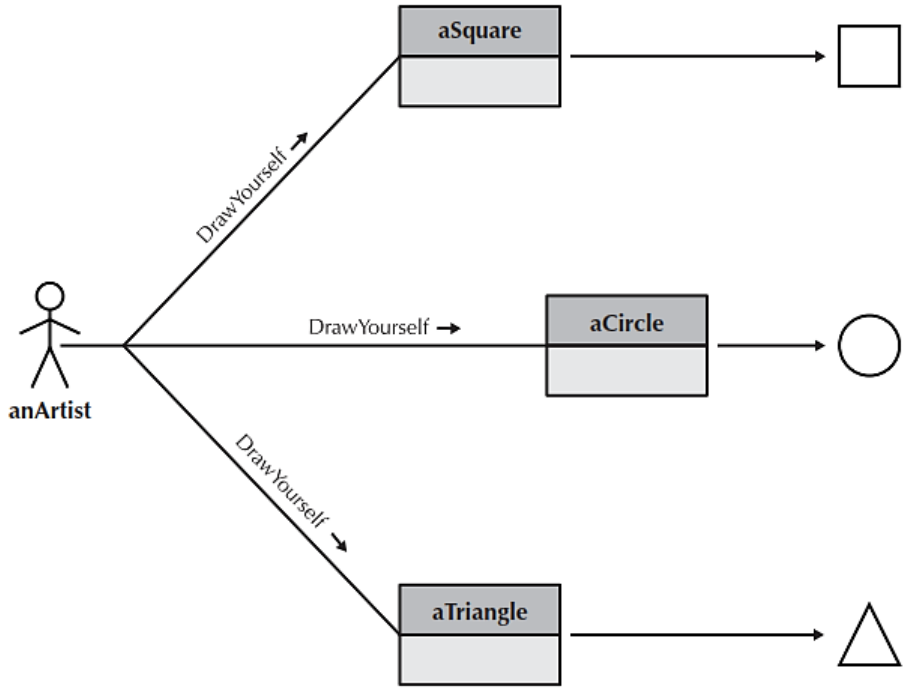
Tác dụng của thừa kế

- Sử dụng lại
- Phát triển chương trình
- Thiết kế hiệu quả
  - ***So sánh 2 thiết kế bên***



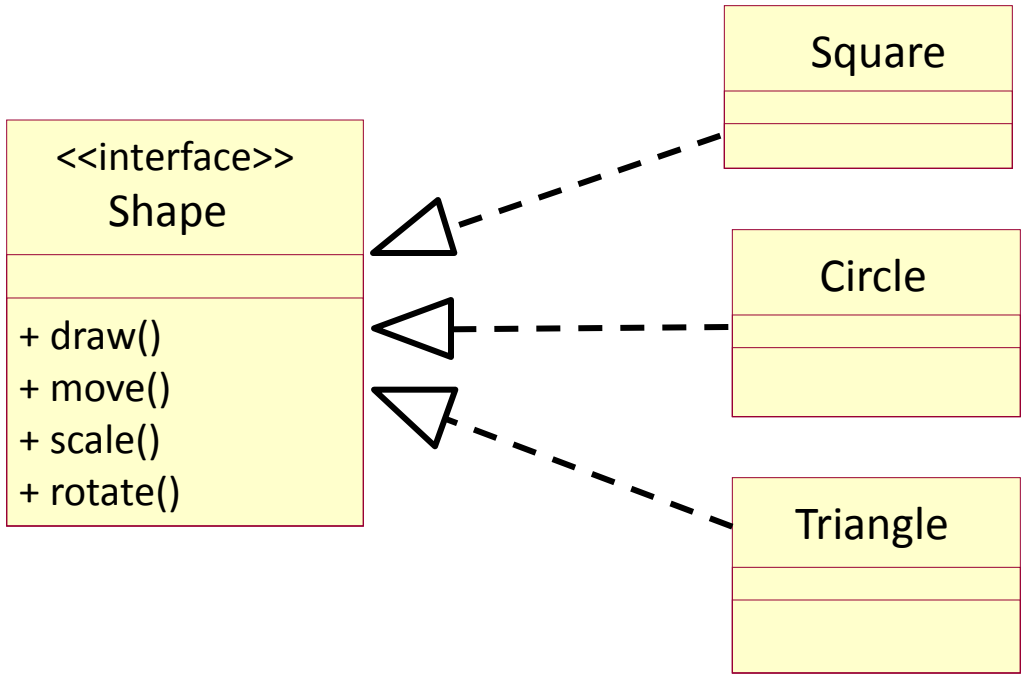
Tương ứng bội

- Tương ứng bội (polymorphism)
  - Một thông điệp có thể được diễn giải theo các cách khác nhau
  - Có thể ẩn các chi tiết cài đặt dưới một giao diện chung



Ví dụ

- Thiết kế tương ứng bội: cần có lớp trừu tượng và hàm ảo



Ví dụ về hàm ảo

```
class Shape {
public:
  virtual void draw()
  {
    cout<<"draw shape \n";
  }
  void paint()
    {
    cout<<"paint shape \n";
  }
};
class Square: public Shape {
public:
  void draw(){cout<<"draw square \n";}
  void paint(){cout<<"paint square"; }
};
class Circle: public Shape {
public:
  void draw() {cout<<"draw circle";}
  void paint() {cout<<"paint circle";}
};
```

```
void main()
{
  Shape *ptr;
  Shape p; //OK
  Circle c;
  Square s;
  ptr= &c;
  ptr->draw();    // draw circle
  ptr->paint();   // paint shape
  ptr= &s;
  ptr->draw();    // draw square
  ptr->paint();   // paint shape
}
```

Có gì khác nhau giữa **draw()** và **paint()** ?

Ví dụ về hàm ảo

```
#include <iostream>
using namespace std;
class A {
public:
     A() { cout << "A constructor\n"; }
     void m1() { cout << "A.m1\n"; m2();  }
     virtual void m2() { cout << "A.m2\n"; }
 };
 class B : public A     {
 public:
     B() { cout << "B constructor\n";}
     void m1() { cout << "B.m1\n"; }
     void m2() { cout << "B.m2\n"; }
  };
  void func(A &a) { a.m1(); }
  int main() {
     B b;
     func(b);
  }
```

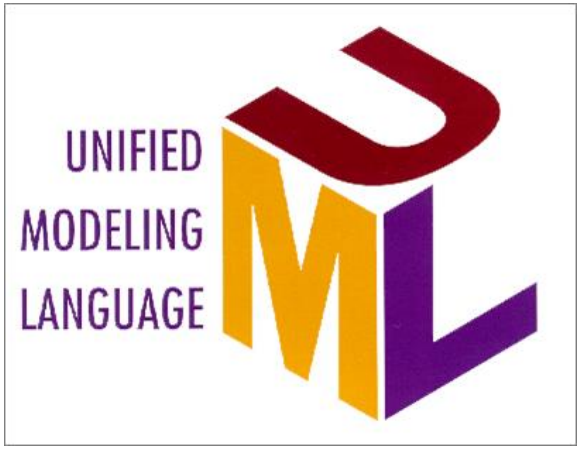
```
A constructor
B constructor
A.m1
B.m2
```

Tóm tắt

- Khái niệm hệ thống và hệ thống thông tin
- Phát triển hệ thống thông tin
  - Các pha trong quá trình phát triển hệ thống
  - Phương pháp (chu trình) phát triển
  - Lựa chọn phương pháp phát triển
- Mô hình hóa hướng đối tượng
- Các khái niệm cơ bản về hướng đối tượng

Ngôn ngữ mô hình hóa UML

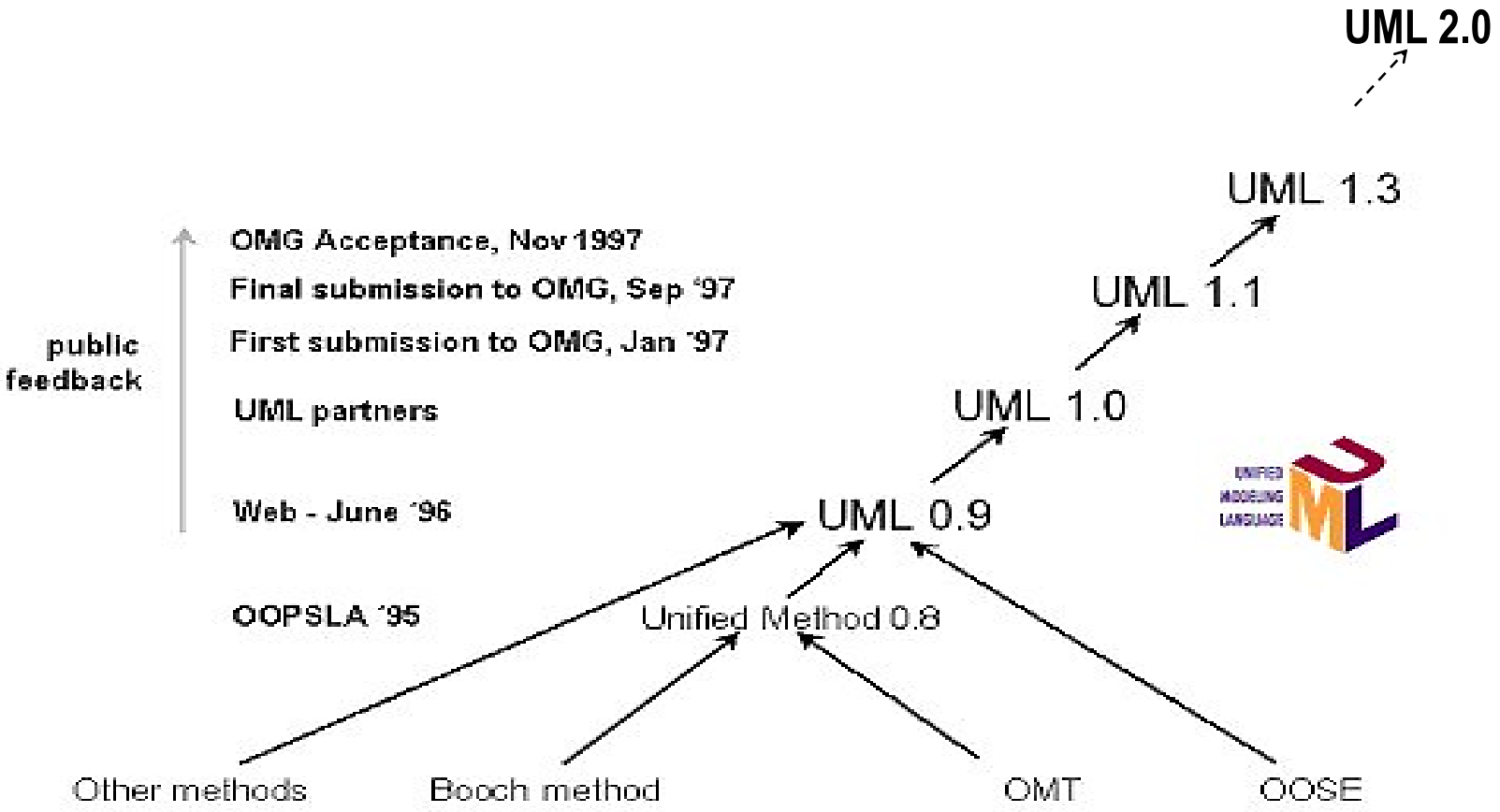
- UML là một ngôn ngữ để mô hình hóa
  - Các nguyên tắc và ký hiệu đã chuẩn hóa
- Các đặc điểm của UML
  - Mô hình trực quan
  - Đặc tả rõ ràng, chính xác
  - Có thể ánh xạ sang các NNLT và CSDL
  - Làm tài liệu: Mô tả yêu cầu, đặc tả
  - ...



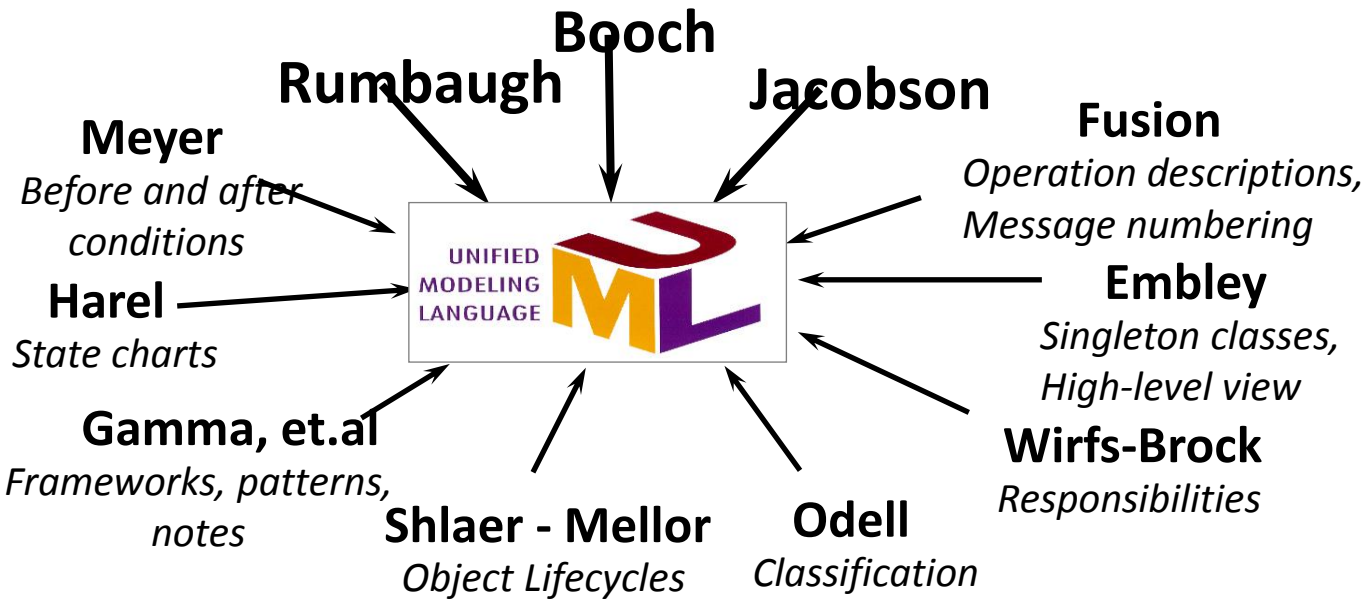
UML trong các giai đoạn phát triển

- Xác định nhu cầu (Requirement determination)
  - Các ca sử dụng (use case) để xác định các yêu cầu
  - Biểu đồ ca sử dụng
- Phân tích (Analysis)
  - Biểu đồ lớp thể hiện cấu trúc tĩnh của hệ thống
  - Các biểu đồ trình tự, biểu đồ trạng thái thể hiện cấu trúc động
- Thiết kế (Design)
  - Các lớp được mô hình hóa chi tiết với các phương thức
- Cài đặt (Implementation)
  - Các mô hình UML có thể ánh xạ sang code

Lịch sử phát triển



Các tác giả chính



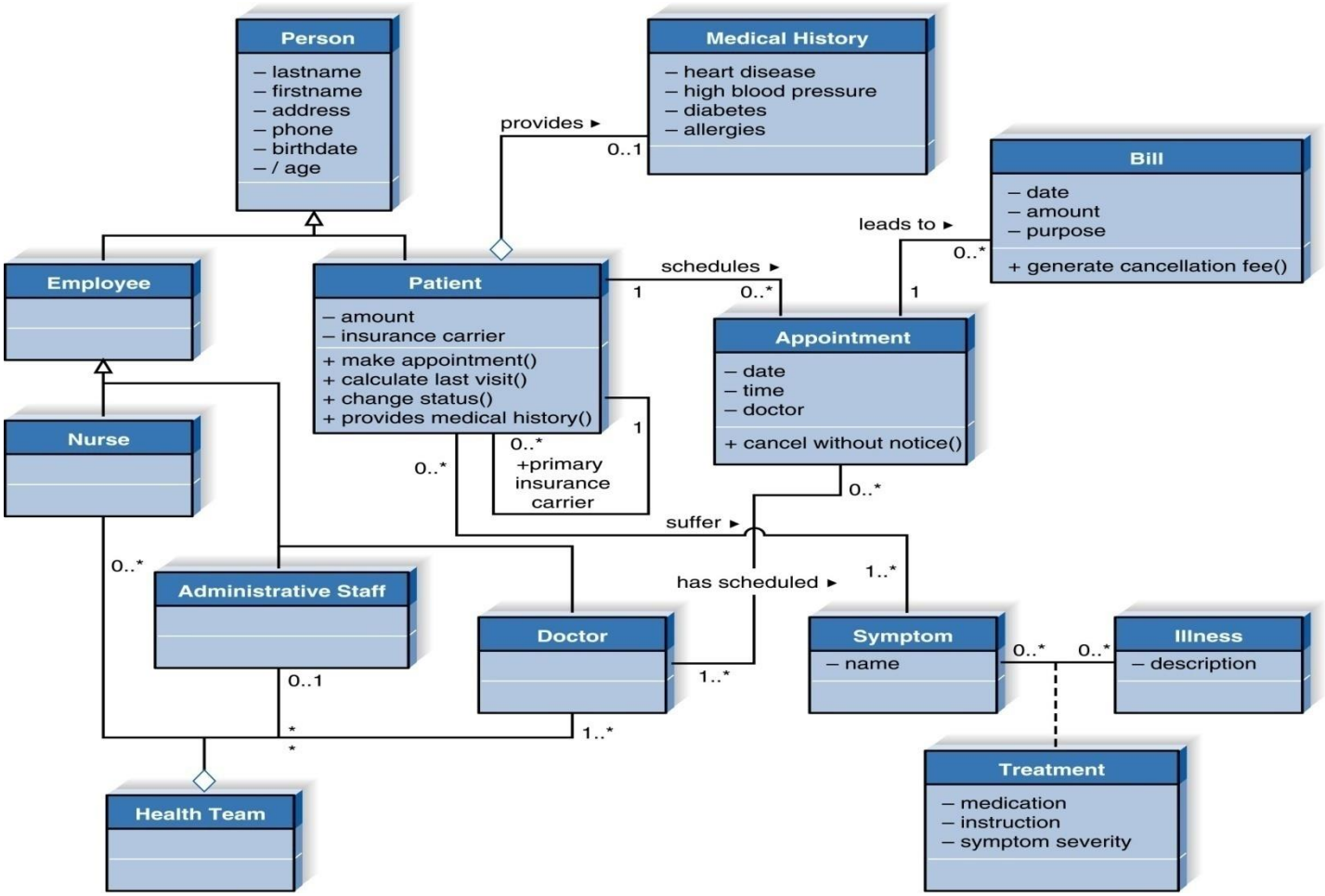
The three amigos:

Grady Booch, Jim Rumbaugh, Ivar Jacobson

Các biểu đồ trong UML

- Các biểu đồ cấu trúc (structured diagrams)
  - **Lớp** (class)
  - Đối tượng (object)
  - Thành phần (component)
  - Bố trí (deployment)
- Các biểu đồ ứng xử (behavioral diagrams)
  - **Ca sử dụng** (use case)
  - **Tuần tự** (sequence)
  - **Cộng tác** (collaboration)/ UML 2.0 communication
  - Trạng thái (statechart)/ UML 2.0 state machine
  - Hoạt động (activity)

Ví dụ một biểu đồ lớp



Câu hỏi

1. Các bước chính của pha Lập kế hoạch?

2. So sánh phương pháp phát triển kiểu Thác nước và phát triển ứng dụng nhanh (RAD)?

3. Những yếu tố nào quan trọng trong lựa chọn phương pháp phát triển hệ thống?

4. Các phương pháp có thể kết hợp với nhau. Giả sử kết hợp giữa Thác nước và Làm bản mẫu cải tiến? Vẽ sơ đồ?

Trả lời

1. Các bước chính của pha Lập kế hoạch?

- Nghiên cứu khả thi (Feasible analysis)
  - Kiểm tra các yếu tố khả thi về kỹ thuật, kinh tế, tổ chức
- Quản lý dự án

Trả lời

2. So sánh phương pháp phát triển kiểu Thác nước và phát triển ứng dụng nhanh (RAD)?

- *Thác nước: một cách hệ thống, tuần tự theo từng bước từ lập kế hoạch, phân tích... đến xây dựng và triển khai.*
- *RAD: tạo nhanh một phiên bản của hệ thống và tinh chỉnh từng bước.*

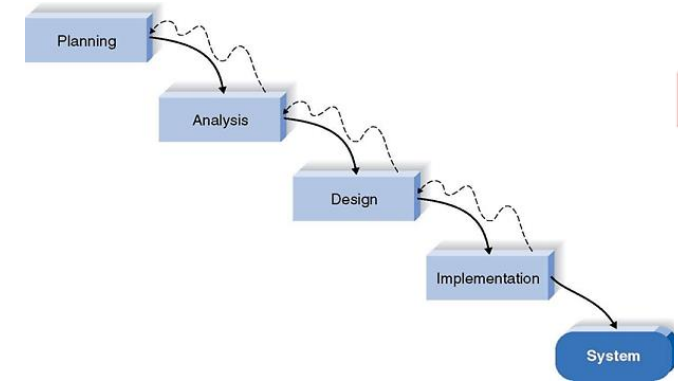
Trả lời

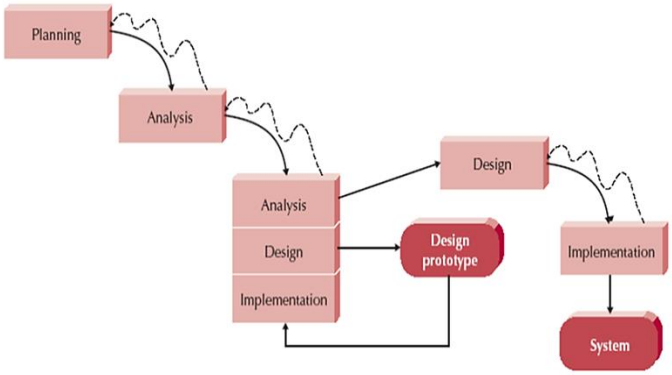
3. Những yếu tố nào quan trọng trong lựa chọn phương pháp phát triển hệ thống?

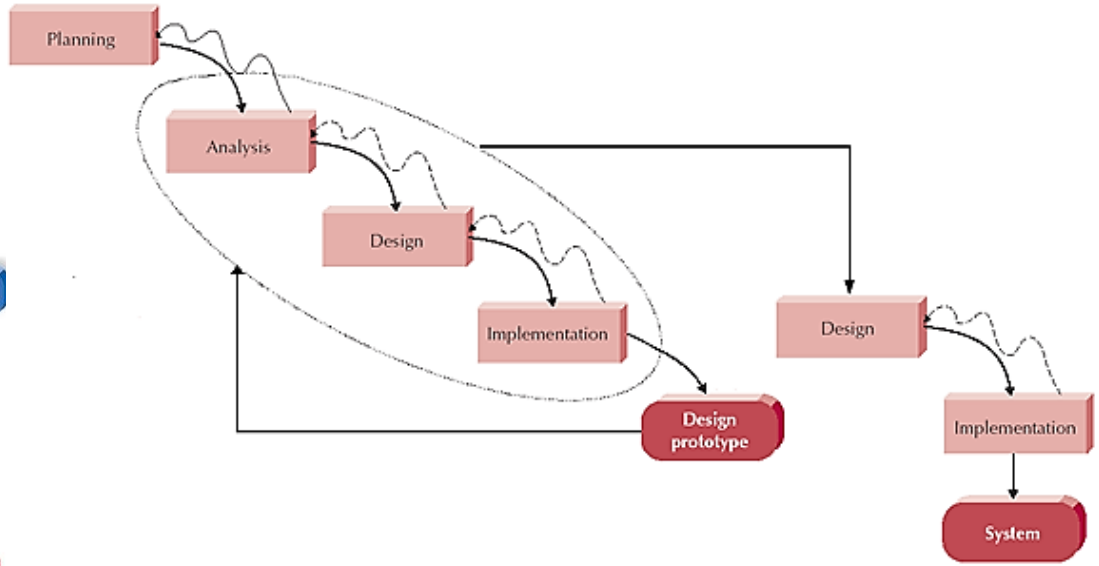
- *Làm rõ các yêu cầu NSD*
- *Quen thuộc với các công nghệ nền*
- *Độ phức tạp của hệ thống*
- *Độ ổn định cần thiết của hệ thống*
- *Thời gian phải hoàn thành*
- *Nhu cầu thấy rõ các quá trình trong từng giai đoạn*

Trả lời

4. Các phương pháp có thể kết hợp với nhau. Giả sử kết hợp giữa Thác nước và Làm bản mẫu cải tiến? Vẽ sơ đồ?







Câu hỏi (2)

1. Đưa ra 3 lý do sử dụng UML?
2. Cái nào sau đây là một mô hình?
   - Một biểu đồ lớp của UML
   - Một tập các biểu đồ lớp mô tả các lớp trong hệ thống
   - Một bản sao xe hơi thể thao bằng đất nặn tỷ lệ 1:100
   - Một xe hơi thể thao
3. So sánh lớp (class) và đối tượng (object)?
4. So sánh phương thức (method) và thông điệp (message)?
5. Phát biểu bài toán "Quản lý thư viện" đơn giản, xác định các yêu cầu (requirement), các tác nhân (actor) và các ca sử dụng (use case) cho hệ thống này.

Trả lời

5. Bài toán "Quản lý thư viện":

Độc giả đến thư viện, xuất trình thẻ thư viện cho thủ thư. Nếu độc giả biết chính xác tên sách, vị trí của nó có thể đến lấy sách đọc. Độc giả cũng có thể tra cứu trên hệ thống máy tính để biết cuốn sách mình cần có trong thư viện không, nếu có thì đang nằm ở vị trí nào. Tuỳ theo loại thẻ, độc giả chỉ có thể ngồi đọc ở thư viện hoặc có thể mượn về nhà. Nếu mượn về nhà thì thủ thư sẽ kiểm tra thông tin độc giả trên máy tính, cập nhật thông tin mượn mới...

- Các **tác nhân**
  - Thủ thư
  - Quản lý thư viện
- Các **ca sử dụng**
  - Mượn sách
  - Trả sách
  - Tra cứu, tìm kiếm sách
  - Báo cáo, thống kê