

Kỹ thuật phần mềm

Các mô hình hệ thống

Mục tiêu

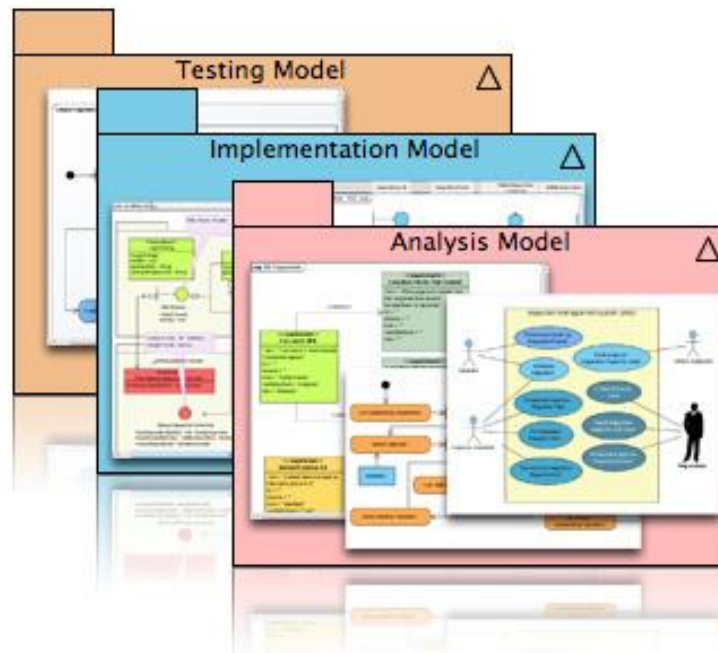
- Giải thích tại sao trong quy trình kĩ nghệ yêu cầu lại cần mô hình hóa **ngữ cảnh** của hệ thống
- Miêu tả về **mô hình hóa hành vi, mô hình hóa dữ liệu, và mô hình hóa đối tượng.**
- Giới thiệu một số kí pháp dùng trong Unified Modeling Language (**UML**)
- Minh họa các CASE workbench hỗ trợ mô hình hóa hệ thống như thế nào

Nội dung

- Mô hình ngữ cảnh – Context model
- Mô hình hành vi – Behavioural model
- Mô hình dữ liệu – Data model
- Mô hình đối tượng – Object model
- CASE workbench

Mô hình hóa hệ thống

- Việc mô hình hóa hệ thống giúp người phân tích hiểu được chức năng của hệ thống
 - Các mô hình được dùng để giao tiếp với khách hàng.



Nhiều mô hình

- Các mô hình khác nhau thể hiện hệ thống từ **các góc nhìn khác nhau**
 - Nhìn từ bên ngoài: mô tả ngữ cảnh hoặc môi trường vận hành của hệ thống;
 - Thể hiện tất cả các thực thể bên ngoài có thể tương tác với hệ thống.
 - Phương diện hành vi: mô tả hành vi của hệ thống;
 - Hoạt động bên trong của các thành tố của hệ thống
 - Phương diện cấu trúc: mô tả kiến trúc hệ thống hoặc kiến trúc dữ liệu
 - Các thành tố và quan hệ giữa chúng

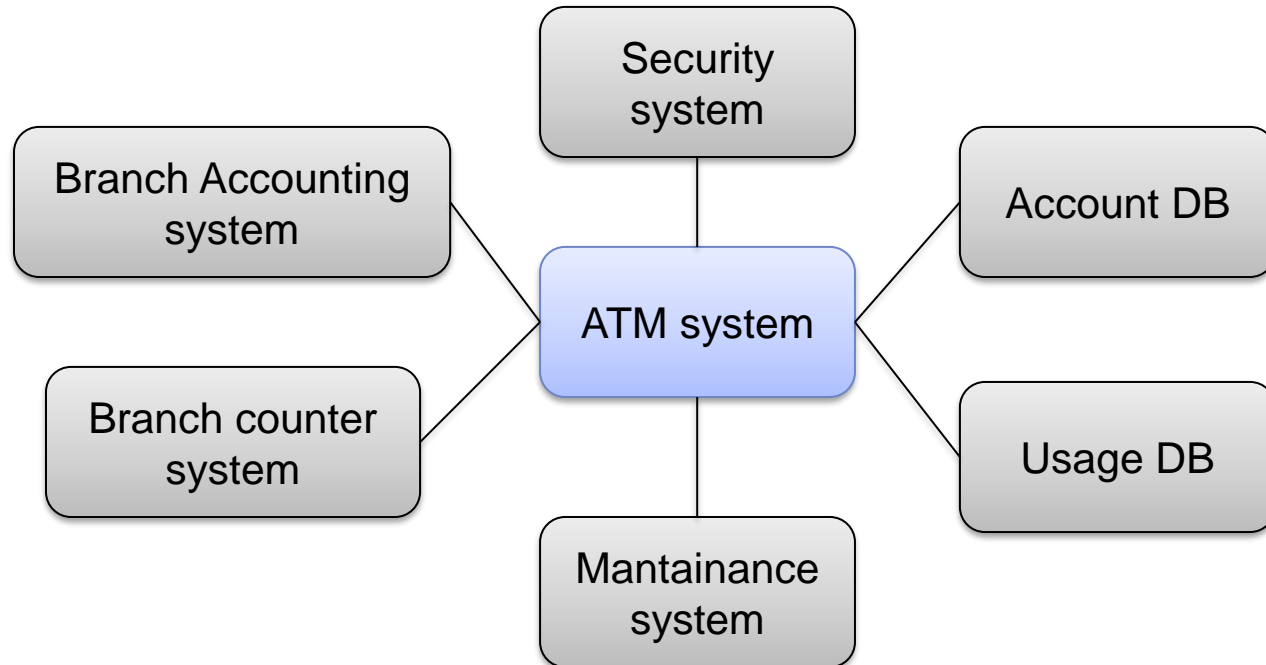
Các kiểu mô hình

- **Data processing** model cho biết dữ liệu được xử lý như thế nào ở các giai đoạn khác nhau.
- **Composition** model cho biết các thực thể được cấu thành từ các thực thể khác như thế nào.
- **Architectural** model mô tả các hệ thống con chính yếu.
- **Classification** model cho biết các thực thể có các đặc điểm chung như thế nào.
- **Stimulus/response** model trình bày phản ứng của hệ thống đối với các sự kiện.

Các mô hình ngữ cảnh

- Các mô hình ngữ cảnh (context model) được dùng để mô tả ngữ cảnh vận hành của một hệ thống –
 - Cái gì nằm ngoài phạm vi của hệ thống
 - Các mối quan tâm về xã hội và tổ chức có thể ảnh hưởng tới quyết định về việc phạm vi hệ thống rộng tới đâu (cái gì thuộc về hệ thống, cái gì nằm ngoài)

Ngữ cảnh của một hệ thống ATM

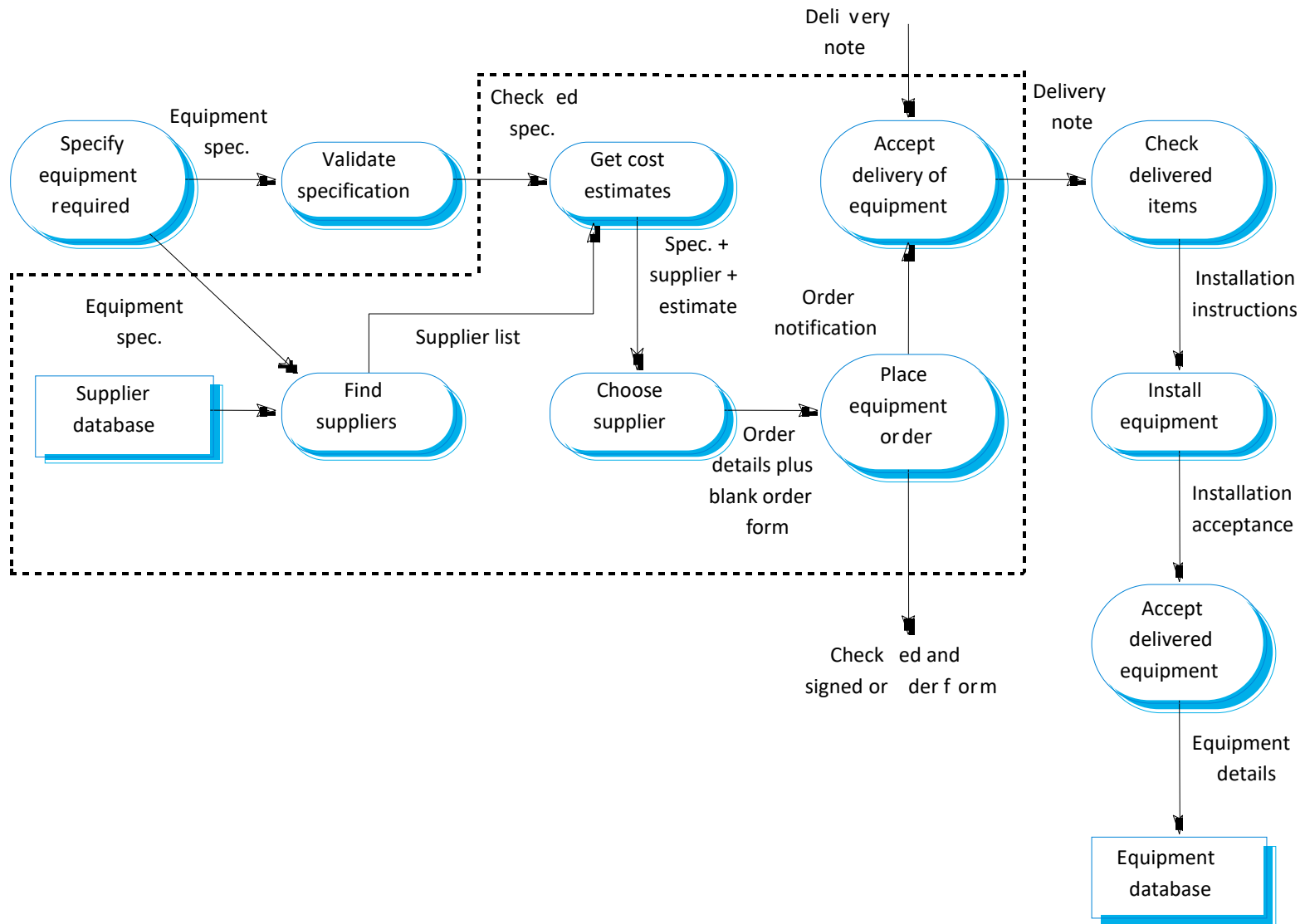


- Không vẽ các vị trí vật lý, chúng kết nối với nhau như thế nào, dữ liệu di chuyển ra sao..

Mô hình tiến trình

- Mô hình tiến trình (process model) mô tả tiến trình chung và các tiến trình mà hệ thống hỗ trợ.
- Mô hình luồng dữ liệu (data flow model) có thể dùng để mô tả các tiến trình và luồng thông tin từ tiến trình này tới tiến trình khác.

Mô hình tiến trình mua thiết bị



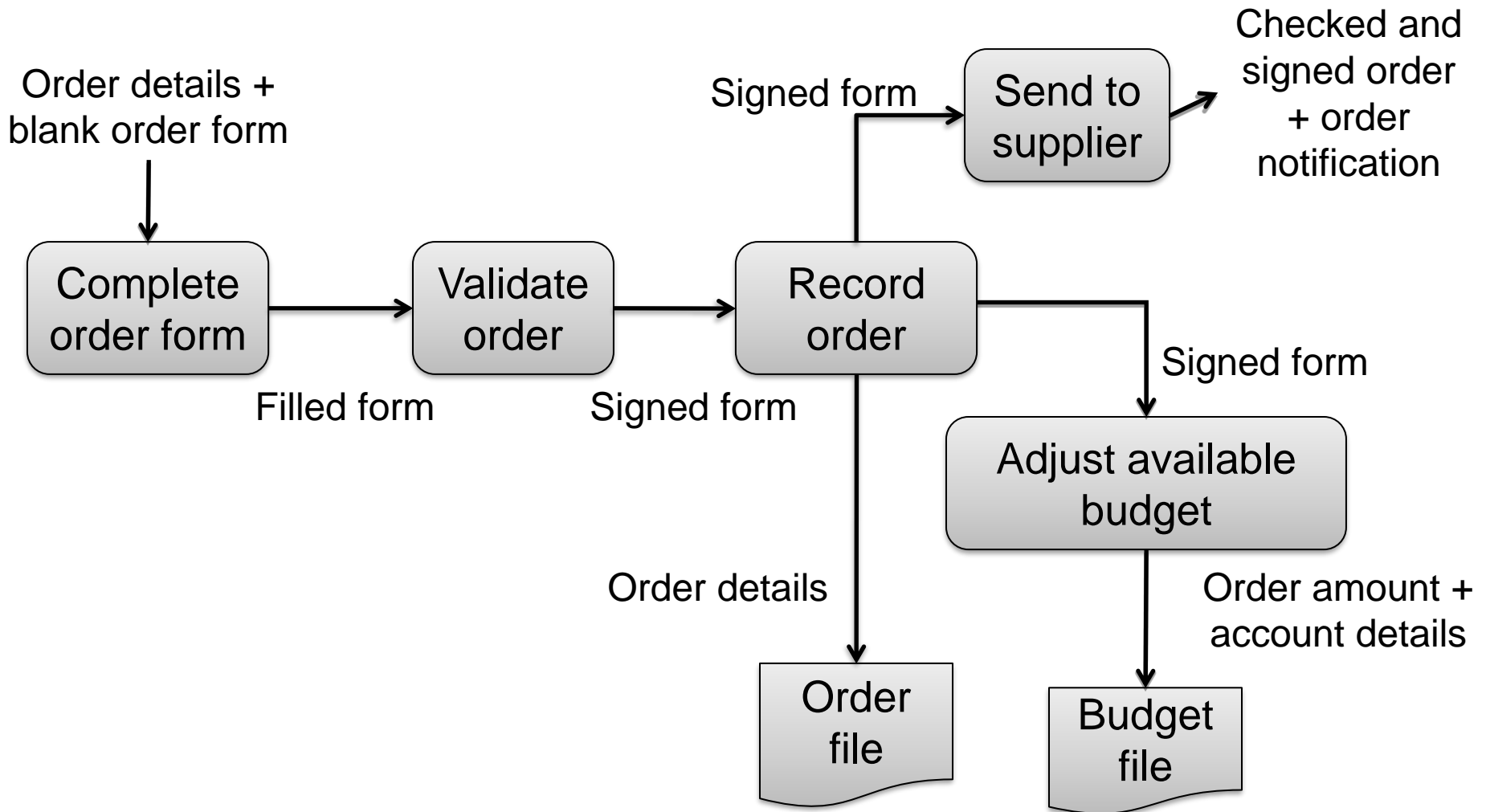
Mô hình hành vi

- Mô hình hành vi (behavioural model) dùng để mô tả hành vi chung của một hệ thống
- Hai loại mô hình hành vi:
 - Mô hình xử lý dữ liệu (data processing) mô tả dữ liệu được xử lý như thế nào khi nó di chuyển trong hệ thống;
 - Mô hình trạng thái (state machine) mô tả phản ứng của hệ thống đối với các sự kiện.
- Cả hai đều cần cho việc mô tả hành vi của hệ thống

Các mô hình xử lý dữ liệu

- Biểu đồ luồng dữ liệu (data flow diagram - DFD) dùng để mô hình hóa việc xử lý dữ liệu trong hệ thống.
- Mô tả các **bước xử lý** khi dữ liệu chảy qua một hệ thống.
- DFD là phần không thể thiếu của nhiều phương pháp phân tích.
 - Được dùng rộng rãi
- **Kí pháp đơn giản và trực quan giúp khách hàng dễ hiểu.**
- Mô tả xử lý dữ liệu từ khi vào hệ thống cho đến khi ra khỏi hệ thống (end-to-end).

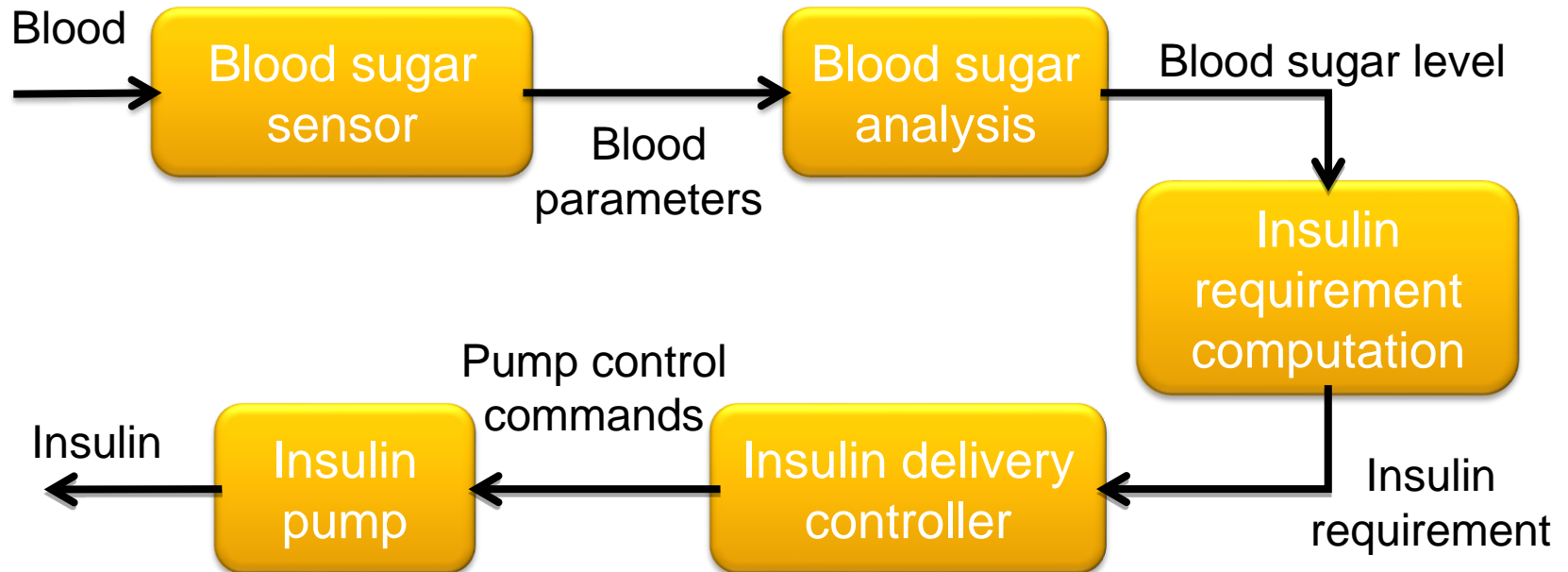
Ví dụ: DFD xử lý đơn đặt hàng



Sơ đồ luồng dữ liệu - DFD

- Mô hình hóa hệ thống từ phương diện **chức năng**.
- Lưu vết và làm tài liệu về mối quan hệ giữa dữ liệu một tiến trình
 - **Giúp hiểu hơn về hệ thống.**
- Có thể dùng để minh họa sự trao đổi dữ liệu giữa một hệ thống và các hệ thống khác

Ví dụ: DFD bơm Insulin



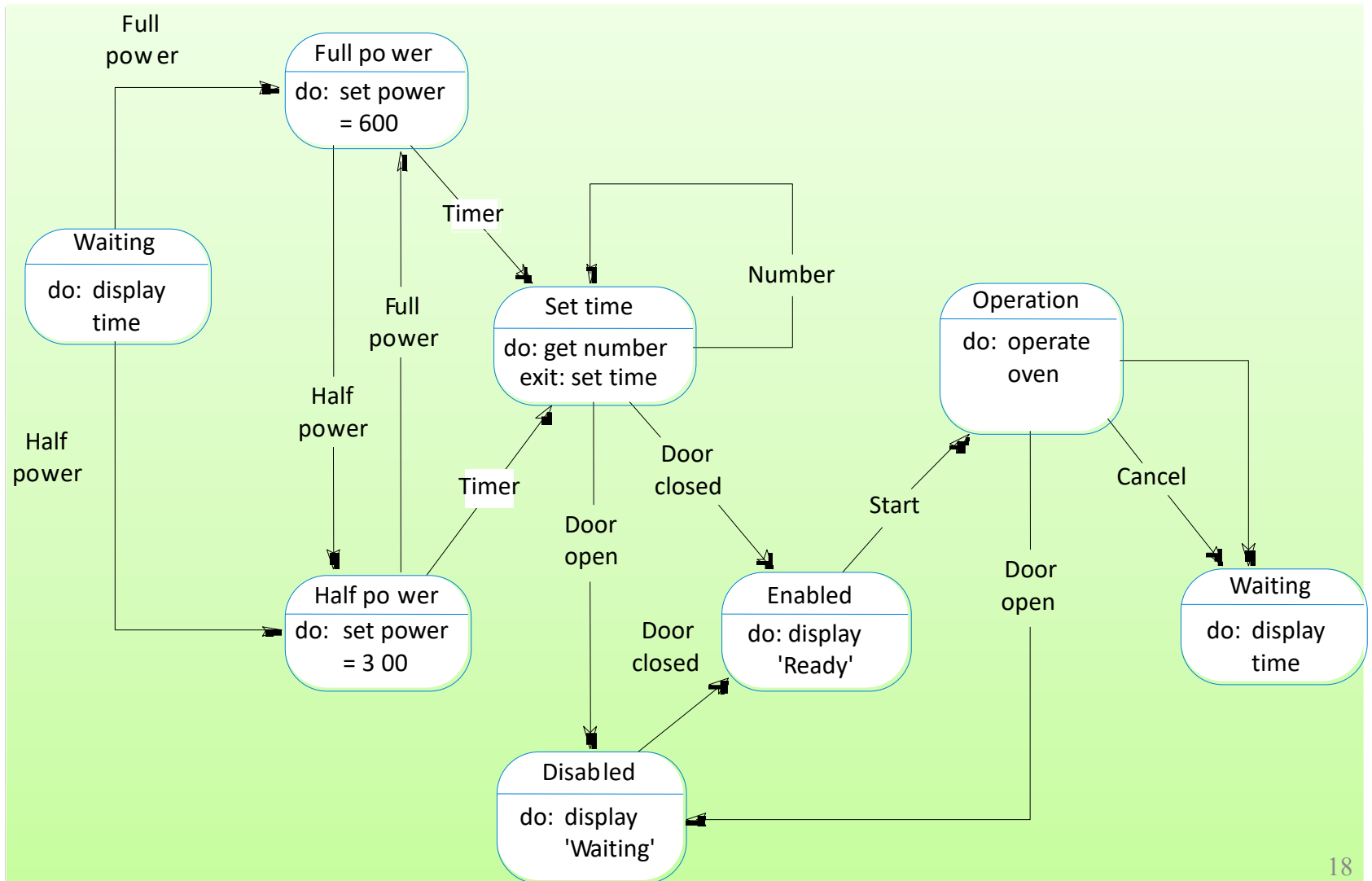
Sơ đồ trạng thái

- (State chart) mô hình hóa hành vi của hệ thống đáp ứng các **sự kiện** bên trong và bên ngoài.
 - Do đó thường được dùng để mô hình hóa các hệ thống thời gian thức.
- Thể hiện các trạng thái hệ thống bằng các nút và các sự kiện là các cung nối các nút.
 - Khi một sự kiện xảy ra, hệ thống chuyển từ trạng thái này sang trạng thái khác.

Sơ đồ trạng thái

- Cho phép phân rã một mô hình thành các mô hình con (xem slide sau)
- Trong mỗi trạng thái, phần '**do**' mô tả ngắn về hành động thực hiện trong trạng thái đó
- Có thể kèm theo các **bảng phương án** mô tả các trạng thái và các sự kiện
- Là một phần của UML

Sơ đồ trạng thái lò vi sóng



Mô tả trạng thái lò vi sóng

Trạng thái	Mô tả
Waiting	Lò đợi input. Màn hình hiện giờ hiện tại.
Half power	Đặt công suất lò là 300 watt. Màn hình hiển thị dòng 'Half power'.
Full power	Đặt công suất lò là 600 watt. Màn hình hiển thị dòng 'Full power'.
Set time	Thời gian nấu được đặt theo input của người dùng. Màn hình hiện thời gian nấu đã chọn và được cập nhật khi đặt thời gian.
Disabled	Tắt chế độ vận hành lò vì lí do an toàn. Đèn bên trong lò bật. Màn hình hiện 'Not ready'.
Enabled	Bật chế độ vận hành lò. Đèn trong lò tắt. Màn hình hiện 'Ready to cook'.
Operation	Lò hoạt động. Đèn trong lò bật. Màn hình hiển thị thời gian đếm ngược. Màn hình. Khi nấu xong, chuông reo trong 5 giây. Đèn bật. Màn hình hiển thị 'Cooking complete' trong khi chuông kêu.

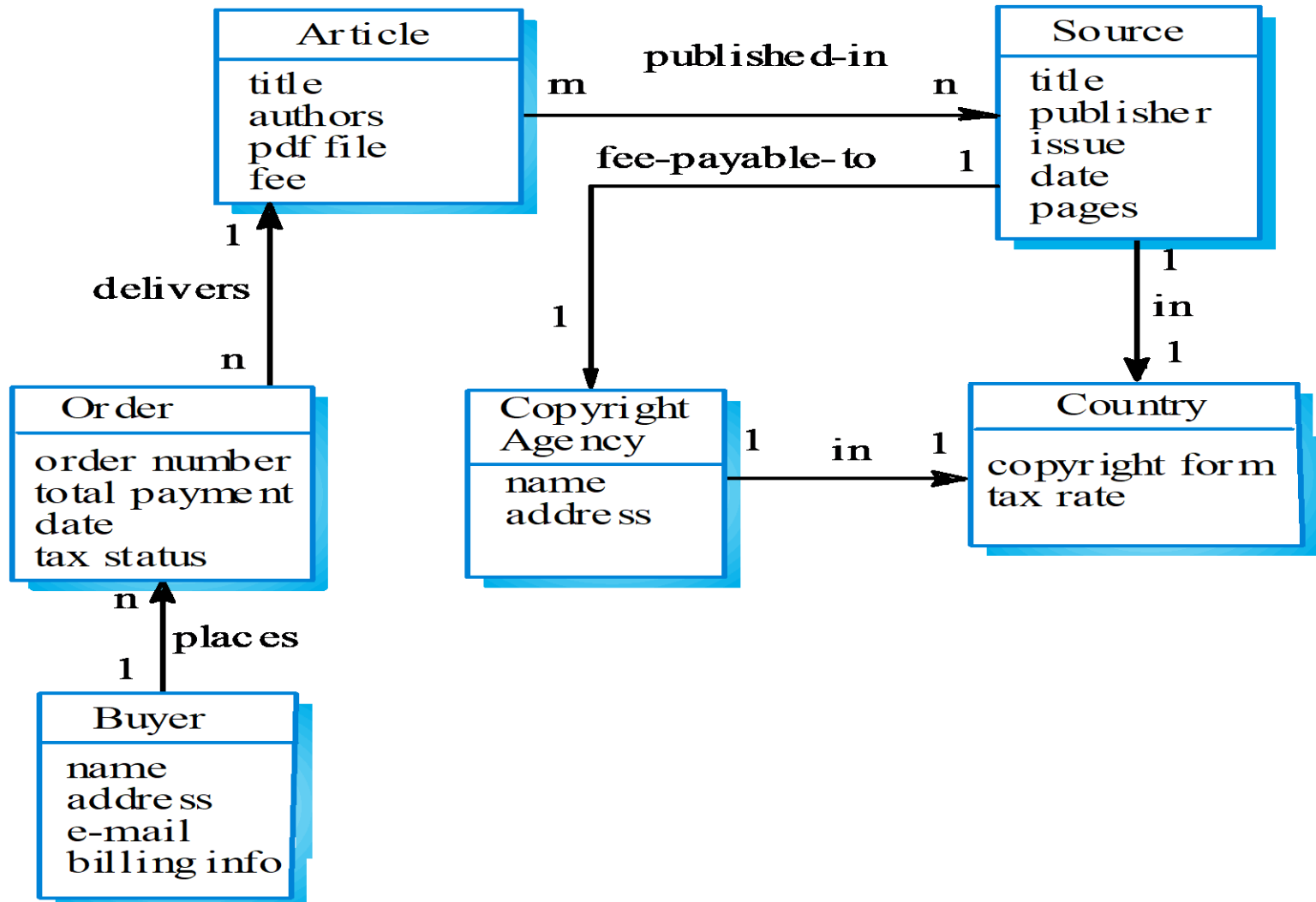
Các sự kiện lò vi sóng

Sự kiện	mô tả
Half power	Người dùng vừa ấn nút half power
Full power	Người dùng vừa ấn nút full power
Timer	Người dùng vừa ấn một trong các nút timer
Number	Người dùng vừa ấn một phím số
Door open	Chốt cửa lò không đóng
Door closed	Chốt cửa lò đóng
Start	Người dùng vừa ấn nút start
Cancel	Người dùng vừa ấn nút cancel

Các mô hình dữ liệu

- Dùng để mô tả cấu trúc lô-gic của dữ liệu mà hệ thống xử lý.
- Mô hình entity-relation-attribute mô tả các thực thể trong hệ thống, quan hệ giữa chúng và các thuộc tính của chúng
- Dùng rộng rãi trong thiết kế cơ sở dữ liệu, có thể cài đặt được ngay bằng CSDL quan hệ.
- UML không cung cấp kí pháp cụ thể nhưng có thể dùng các đối tượng (object) và quan hệ (association).

Mô hình dữ liệu



Từ điển dữ liệu - Data dictionary

- Từ điển dữ liệu là danh sách tất cả các **tên** được dùng
 - Ngoài ra còn mô tả về các thực thể, quan hệ, và thuộc tính
- Ưu điểm
 - Giúp quản lý tên và tránh trùng tên;
 - Kho chứa kiến thức về tổ chức, kết nối các hoạt động phân tích, thiết kế và cài đặt
- Nhiều CASE workbench hỗ trợ lập từ điển dữ liệu.

Ví dụ một phần từ điển dữ liệu

Name	Description	Type	Date
Article	Details of the published article that may be ordered by people.	Entity	30.08.2021
authors	The names of the authors of the article who may be due a share of the fee.	Attribute	30.08.2021
Buyer	The person or organisation that orders a copy of the article.	Entity	30.08.2021
fee-payable-to	A 1:1 relationship between Article and the Copyright Agency who should be paid the copyright fee.	Relation	29.08.2021
Address (Buyer)	The address of the buyer. This is used to any paper billing information that is required.	Attribute	30.08.2021

Mô hình đối tượng

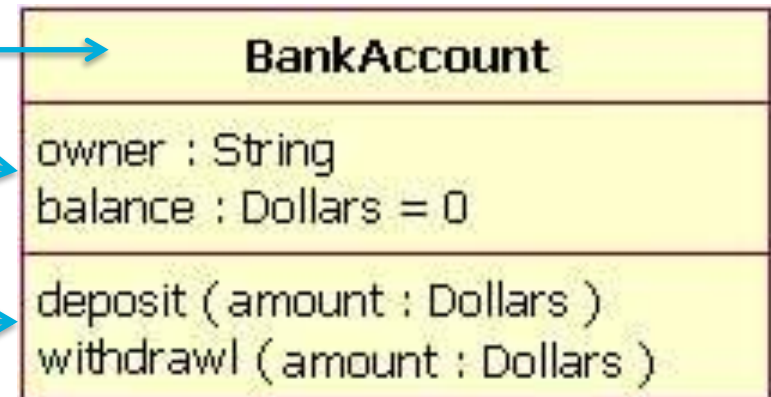
- Mô tả hệ thống theo các lớp đối tượng (class) và quan hệ giữa chúng
- Một lớp đối tượng là một trừu tượng hóa của một tập đối tượng có bộ thuộc tính chung và các thao tác giống nhau.
- Các loại mô hình đối tượng:
 - Inheritance model – mô hình kế thừa;
 - Aggregation model – mô hình cộng gộp;
 - Interaction model – mô hình tương tác.
- UML là một chuẩn mô hình hóa hướng đối tượng hiệu quả

Mô hình đối tượng

- Là cách **tự nhiên** phản ánh các thực thể trong thế giới thực mà hệ thống xử lý
- Các đối tượng càng trừu tượng thì càng khó dùng loại mô hình này
- Việc xác định các lớp là một quy trình khó
 - **Đòi hỏi hiểu biết sâu về miền ứng dụng**
- Có thể tái sử dụng các lớp đối tượng phản ánh các thực thể của miền ứng dụng cho các hệ thống khác

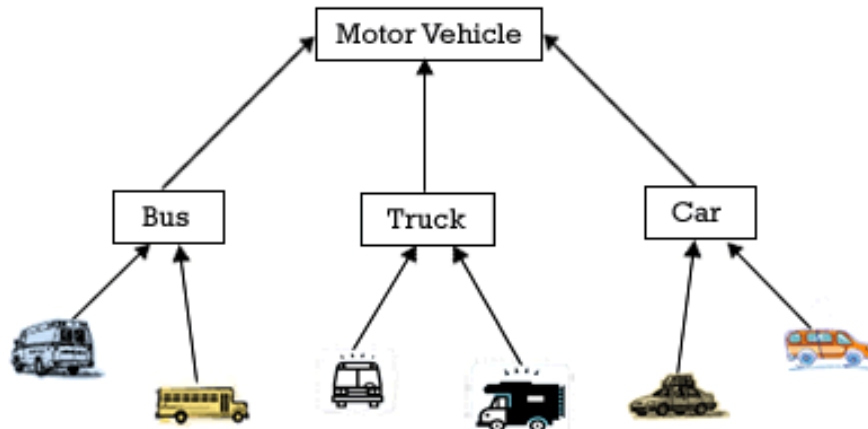
Kí pháp lớp đối tượng của UML

- Mỗi lớp là một hình chữ nhật với:
 - Tên đặt tại đỉnh,
 - Các thuộc tính nằm ở ngăn giữa, và
 - Các thao tác nằm ở ngăn dưới;
- Quan hệ giữa các lớp thể hiện bằng các đường nối giữa các đối tượng;

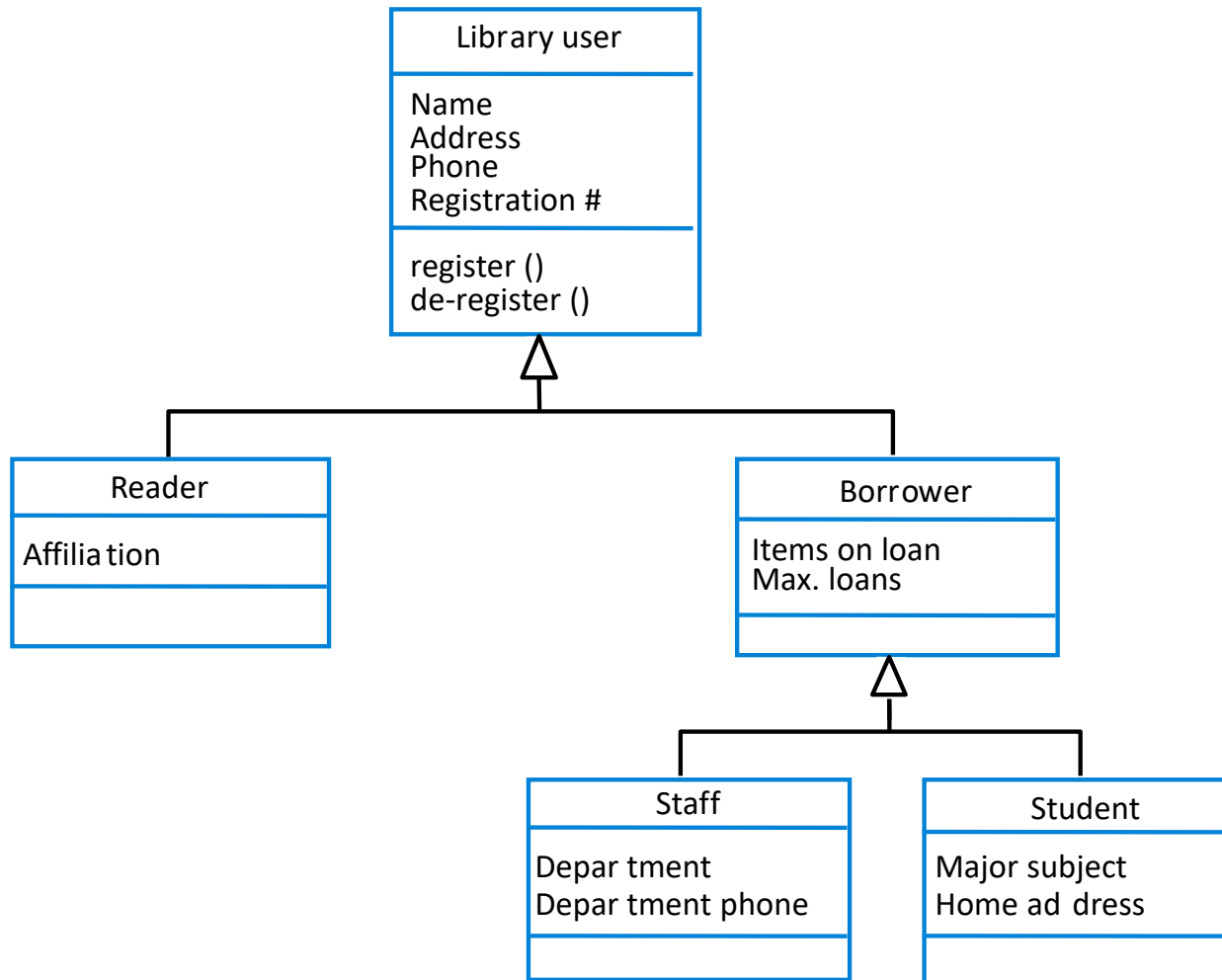


Mô hình thừa kế

- Tổ chức các lớp đối tượng thuộc miền ứng dụng thành một cấu trúc phân cấp
- Các lớp trên đỉnh cấu trúc phản ánh các tính chất chung của tất cả các lớp
- Các lớp thừa kế thuộc tính và các dịch vụ từ các lớp cha
 - Các dịch vụ này có thể được chuyên hóa nếu cần
- Thiết kế cấu trúc phân cấp các lớp đối tượng có thể là một quy trình khó nếu phải tránh trùng lặp giữa các nhánh khác nhau



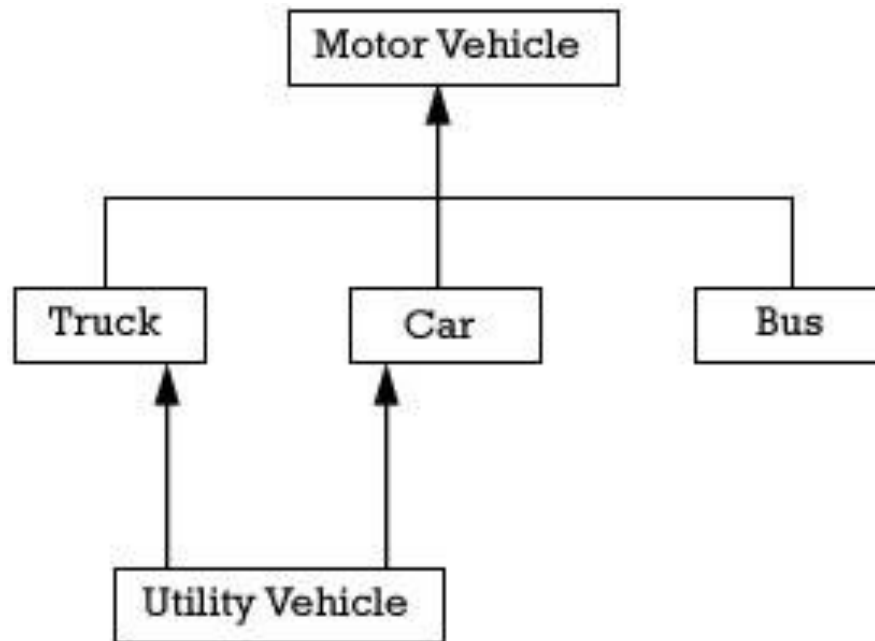
Ví dụ về cây phân cấp các lớp



Đa thừa kế

- Cho phép các lớp thừa kế từ nhiều hơn một lớp cha
 - Thay vì chỉ thừa kế các thuộc tính và dịch vụ từ một lớp cha duy nhất
- Có thể dẫn đến xung đột ngữ nghĩa
 - Khi thuộc tính/dịch vụ trùng tên ở các lớp cha khác nhau có ngữ nghĩa khác nhau
- Đa thừa kế làm cho cấu trúc phân cấp phức tạp hơn

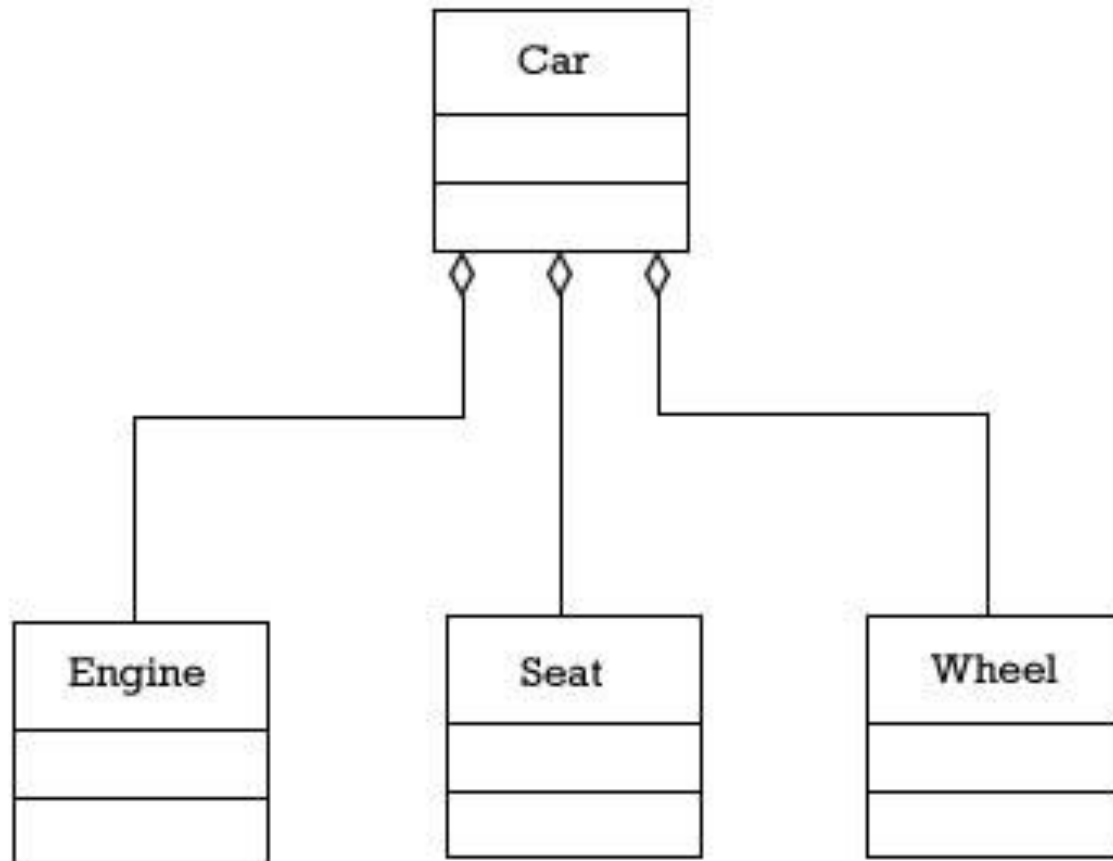
Ví dụ về đa thừa kế



Object aggregation

- Mô hình cộng gộp (aggregation model) mô tả các lớp được cấu thành từ các lớp khác.
- Mô hình cộng gộp tương tự với quan hệ part-of trong các mô hình dữ liệu.

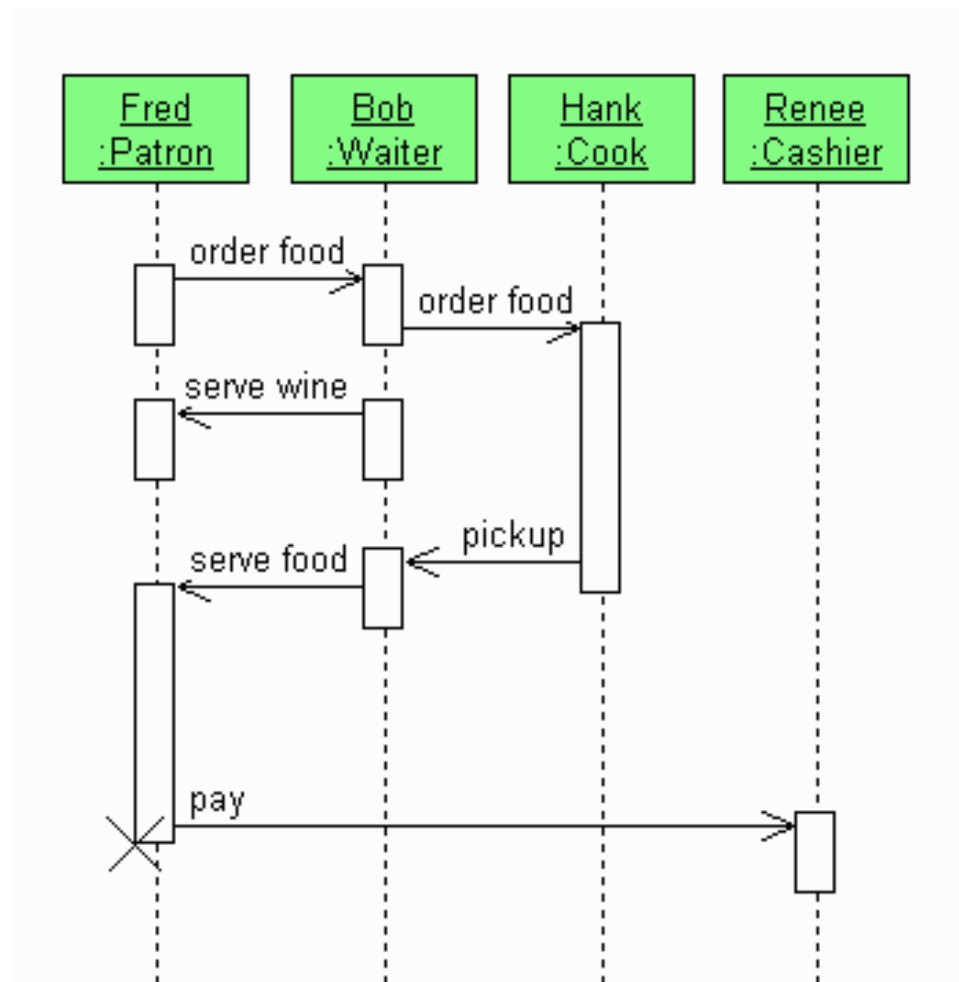
Ví dụ



Mô hình tương tác

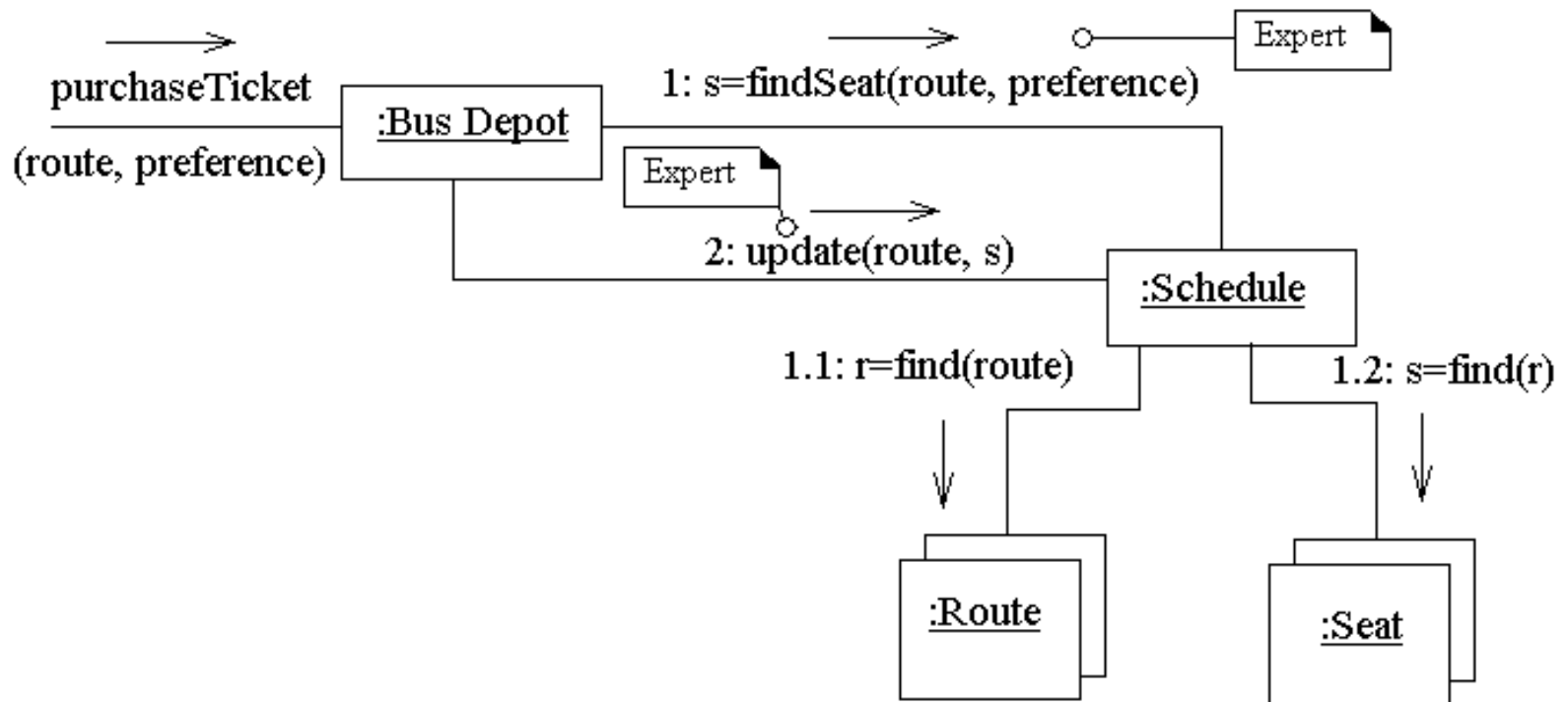
- Là một mô hình hành vi mô tả **sự tương tác giữa các đối tượng** để tạo ra một hành vi cụ thể của hệ thống đã được đặc tả bằng một use-case.
- Trong UML, tương tác giữa các đối tượng được mô hình hóa bằng:
 - Biểu đồ tuần tự - sequence diagram
 - Biểu đồ cộng tác - collaboration diagram

Ví dụ biểu đồ tuần tự



Ví dụ biểu đồ cộng tác

Collaboration Diagram for Purchasing Bus Ticket



Các phương pháp có cấu trúc

- Tích hợp việc mô hình hóa hệ thống vào trong phương pháp
 - Dành cho thu thập và phân tích yêu cầu
- Quy định
 - Một bộ các mô hình,
 - Một quy trình để lập được các mô hình đó, và
 - Các quy tắc và hướng dẫn cần áp dụng cho các mô hình đó
- Được hỗ trợ bởi một số CASE tool

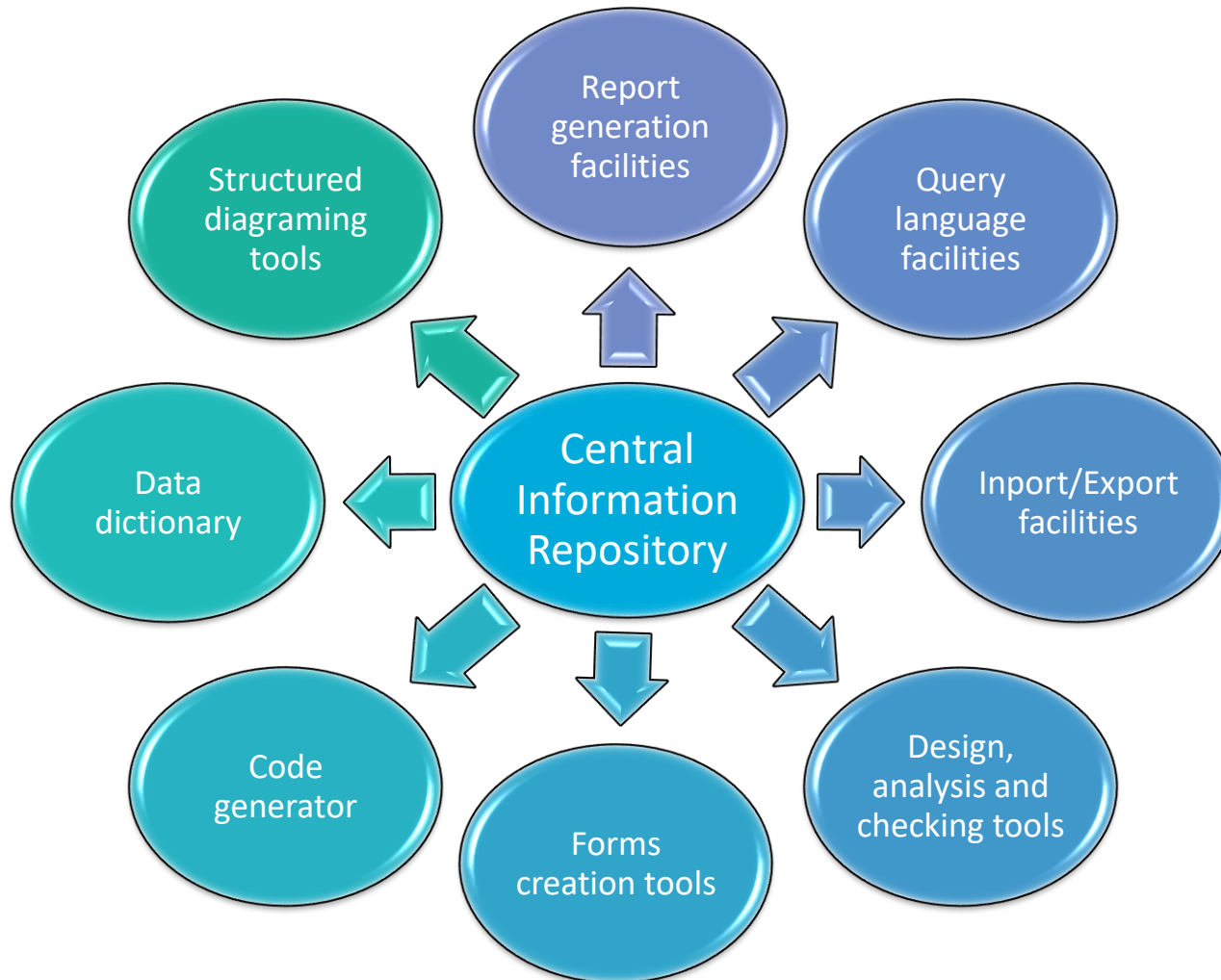
Nhược điểm của các phương pháp có cấu trúc

- Không mô hình hóa được các yêu cầu hệ thống loại **phi chức năng**
- Thường không chứa thông tin về một phương pháp có thích hợp với một bài toán nào đó hay không
- Có thể tạo quá nhiều tài liệu
- Các mô hình hệ thống quá chi tiết và quá khó để người dùng có thể hiểu được.

CASE workbook

- Một bộ công cụ được thiết kế để hỗ trợ các hoạt động quy trình phần mềm có liên quan như phân tích, thiết kế hoặc kiểm thử.
- Các workbook dành cho phân tích và thiết kế hỗ trợ việc mô hình hóa hệ thống trong cả hai hoạt động kỹ nghệ yêu cầu và thiết kế hệ thống.
- Các workbook này có thể hỗ trợ một phương pháp thiết kế cụ thể hoặc có thể hỗ trợ tạo các kiểu mô hình hệ thống khác nhau.

Một workbench cho phân tích và thiết kế



Thành phần của workbench phân tích

- Các chương trình vẽ biểu đồ
- Các công cụ phân tích và kiểm tra mô hình
- Repository và ngôn ngữ truy vấn liên quan
- Từ điển dữ liệu
- Các công cụ định nghĩa và tạo report
- Các công cụ định nghĩa form
- Các bộ import/export
- Các công cụ sinh mã

Tổng kết

- Một **mô hình** là một góc nhìn trừu tượng về hệ thống. Các loại mô hình kết hợp với nhau cung cấp các thông tin hệ thống khác nhau.
- **Mô hình ngữ cảnh** mô tả vị trí của một hệ thống trong môi trường với các hệ thống và tiến trình khác.
- **Mô hình luồng dữ liệu** có thể dùng để mô hình hóa việc xử lý dữ liệu trong một hệ thống.
- **Mô hình trạng thái** mô hình hóa hành vi của hệ thống khi đáp ứng các sự kiện trong và ngoài.

Tổng kết

- **Semantic data** model mô tả cấu trúc lô-gic của dữ liệu ra vào hệ thống.
- **Mô hình đối tượng** mô tả các thực thể lô-gic của hệ thống, phân loại (classification) và cấu tạo của chúng (aggregation).
- **Mô hình tuần tự** mô tả tương tác giữa các actor và các đối tượng hệ thống mà họ sử dụng.
- **Các phương pháp có cấu trúc** cung cấp một framework cho việc phát triển các mô hình hệ thống.