

Chương 12

■ Thiết kế dựa trên khuôn mẫu

Slide được xây dựng theo cuốn

Software Engineering: A Practitioner's Approach, 7/e
của **Roger S. Pressman**

Slides copyright © 1996, 2001, 2005, 2009 by Roger S. Pressman

Chỉ dành cho mục đích giáo dục phi lợi nhuận.

Sinh viên đại học có thể sao chép slide này nếu dùng chung với cuốn *Software Engineering: A Practitioner's Approach, 7/e*. Mọi hình vi tái bản hoặc sử dụng đều bị cấm khi chưa có sự cho phép của tác giả.

Tất cả các thông tin bản quyền phải xuất hiện trong các slide nếu xuất hiện trên trang web dành cho sinh viên.

Khuôn mẫu thiết kế (Design Patterns)

- Mỗi chúng ta đều đã trải qua vấn đề thiết kế và từng nghĩ: *Liệu đã có ai phát triển một lời giải cho vấn đề này chưa?*
 - Điều gì sẽ xảy ra nếu đã có một cách tiêu chuẩn để mô tả một vấn đề (để bạn có thể nhìn nhận nó), và một phương pháp có tổ chức để trình bày lời giải cho vấn đề đó?
- *Khuôn mẫu thiết kế* là một phương pháp có hệ thống để mô tả các vấn đề và giải pháp, cho phép các cộng đồng công nghệ phần mềm có thể nắm bắt kiến thức thiết kế theo cách có thể tái sử dụng

Khuôn mẫu thiết kế (Design Patterns)

- *Mỗi khuôn mẫu mô tả một vấn đề xảy ra hết lần này đến lần khác trong môi trường của chúng ta và sau đó mô tả cốt lõi của giải pháp cho vấn đề đó theo một cách mà bạn có thể sử dụng nó hàng triệu lần mà không bao giờ phải làm lại một việc lần thứ hai.*
 - Christopher Alexander, 1977
- “một quy tắc ba phần diễn tả một mối quan hệ giữa một ngữ cảnh, một vấn đề, và một giải pháp.”

Khái niệm cơ bản

- *Ngữ cảnh* cho phép người đọc hiểu được môi trường trong đó các vấn đề phát sinh và giải pháp nào có thể thích hợp trong môi trường đó.
- Một tập hợp các yêu cầu, bao gồm cả những hạn chế và khó khăn, hoạt động như một hệ thống các lực lượng có ảnh hưởng tới cách:
 - các vấn đề có thể được diễn giải trong ngữ cảnh của nó và
 - giải pháp có thể được áp dụng hiệu quả như thế nào

Khuôn mẫu hiệu quả

- Coplien [Cop05] nêu đặc điểm cho một khuôn mẫu thiết kế hiệu quả theo các cách sau đây:
 - *Nó giải quyết một vấn đề:* Khuôn mẫu đưa ra giải pháp, không chỉ là quy tắc hay chiến lược trừu tượng.
 - *Nó là một khái niệm đã được chứng minh:* Khuôn mẫu đưa ra giải pháp dựa trên dữ liệu theo dõi, không phải là lý thuyết hay suy đoán.
 - *Giải pháp không hiển nhiên:* Rất nhiều kỹ thuật giải quyết vấn đề (chẳng hạn như mô hình hoặc các phương pháp thiết kế phần mềm) cố gắng đạt được giải pháp từ quy tắc đầu tiên. Khuôn mẫu tốt nhất *tạo ra* một giải pháp cho một vấn đề một cách gián tiếp - một cách tiếp cận cần thiết cho những vấn đề khó nhất của thiết kế.
 - *Nó mô tả một mối quan hệ:* Khuôn mẫu không chỉ mô tả mô-đun mà mô tả kiến trúc và cơ chế sâu hơn của hệ thống.
 - *Khuôn mẫu có một thành phần con người quan trọng (giảm thiểu ảnh hưởng của con người).* Tất cả phần mềm cung cấp cho con người sự thoải mái hoặc chất lượng cuộc sống; khuôn mẫu tốt nhất hấp dẫn rõ ràng về cả thẩm mỹ và tiện ích.

Khuôn mẫu khả sinh (Generative patterns)

- *Khuôn mẫu khả năng* mô tả một khía cạnh quan trọng và có thể lặp lại của một hệ thống và sau đó cung cấp cho chúng ta một cách xây dựng khía cạnh đó trong một hệ thống của các nguồn lực riêng biệt đối với từng ngữ cảnh.
- Một tập hợp các khuôn mẫu thiết kế khả sinh có thể được dùng để sinh một ứng dụng hoặc một hệ thống dựa trên máy tính có kiến trúc cho phép nó thích nghi với sự thay đổi.

Các loại khuôn mẫu

- *Architectural patterns* describe broad-based design problems that are solved using a structural approach.
- *Data patterns* describe recurring data-oriented problems and the data modeling solutions that can be used to solve them.
- *Component patterns* (also referred to as *design patterns*) address problems associated with the development of subsystems and components, the manner in which they communicate with one another, and their placement within a larger architecture
- *Interface design patterns* describe common user interface problems and their solution with a system of forces that includes the specific characteristics of end-users.
- *WebApp patterns* address a problem set that is encountered when building WebApps and often incorporates many of the other patterns categories just mentioned.

Các loại khuôn mẫu

- ***Khuôn mẫu sáng tạo (Creational patterns)*** tập trung vào việc khởi tạo, sáng tác và biểu diễn của các đối tượng, ví dụ:
 - Abstract factory pattern
 - Factory method pattern
- ***Khuôn mẫu cấu trúc*** tập trung vào các vấn đề và các giải pháp liên quan đến cách các lớp và các đối tượng được tổ chức và tích hợp để xây dựng một cấu trúc lớn hơn, ví dụ như:
 - Adapter pattern
 - Aggregate pattern
- ***Khuôn mẫu hành vi*** giải quyết các vấn đề liên quan đến việc phân công trách nhiệm giữa các đối tượng và cách thức mà giao tiếp được thực hiện giữa các đối tượng, ví dụ như:
 - Chain of responsibility pattern:
 - Command pattern:

Khung (frameworks)

- Bản thân khuôn mẫu có thể không đủ để phát triển một thiết kế đầy đủ
 - Trong một số trường hợp, cần thiết phải cung cấp một cơ sở hạ tầng thực hiện cụ thể, được gọi là một khung (*framework*) cho công việc thiết kế.
 - Có nghĩa là, bạn có thể chọn một “*một kiến trúc nhỏ có thể tái sử dụng cung cấp*” cấu trúc chung và hành vi cho một họ các sự trừu tượng hóa phần mềm, cùng với một ngữ cảnh... xác định sự cộng tác và sử dụng của chúng trong một miền nhất định” [Amb98]
- Một framework không phải là một khuôn mẫu kiến trúc, mà là một bộ khung với một tập hợp các “**plug points**” (còn được gọi là hooks hay slots) cho phép nó thích ứng với một miền vấn đề xác định.
 - Những “plug points” cho phép bạn tích hợp các lớp vấn đề cụ thể hoặc chức năng trong các bộ khung.

Mô tả một khuôn mẫu

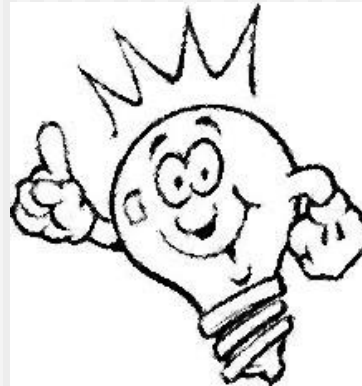
- **Tên khuôn mẫu**—mô tả bản chất của khuôn mẫu với một cái tên ngắn nhưng biểu cảm.
- **Vấn đề**—Mô tả vấn đề mà khuôn mẫu giải quyết.
- **Động lực**—cung cấp một ví dụ cho vấn đề
- **Ngữ cảnh**—*Mô tả môi trường mà vấn đề phát sinh, bao gồm miền ứng dụng*
- **Nguồn lực**—*liệt kê các hệ thống của các lực lượng có ảnh hưởng đến cách thức mà các vấn đề được giải quyết; bao gồm một cuộc thảo luận về các giới hạn và ràng buộc phải được xem xét*
- **Giải pháp**—cung cấp một mô tả chi tiết về các giải pháp đề xuất cho vấn đề
- **Mục đích**—mô tả các khuôn mẫu và những gì nó làm
- **Sự cộng tác**—mô tả cách các khuôn mẫu khác đóng góp vào giải pháp
- **Hệ quả**—mô tả những sự đánh đổi tiềm năng phải được xem xét khi khuôn mẫu được thực hiện và những hệ quả của việc sử dụng khuôn mẫu.
- **Sự triển khai**—xác định các vấn đề đặc biệt cần được xem xét khi thực hiện khuôn mẫu
- **Ứng dụng đã biết**—cung cấp các ví dụ về cách sử dụng của các khuôn mẫu thiết kế trong ứng dụng thực tế.
- **Khuôn mẫu liên quan**—tham khảo chéo liên quan đến các khuôn mẫu thiết kế

Ngôn ngữ khuôn mẫu

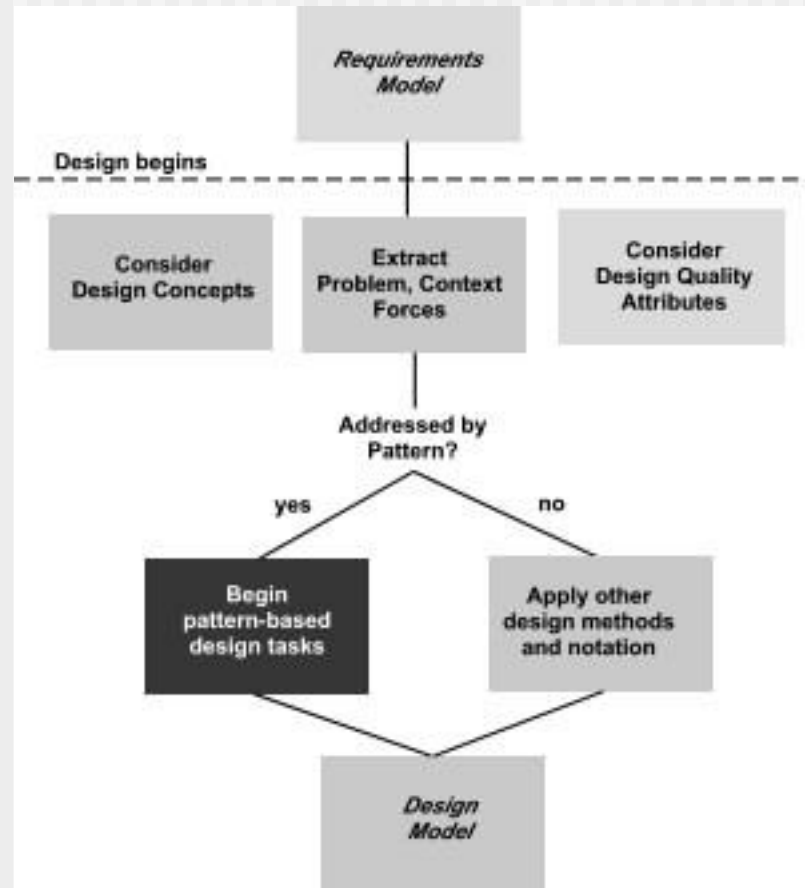
- Một ngôn ngữ khuôn mẫu bao gồm tập hợp các khuôn mẫu:
 - được mô tả sử dụng một bản mẫu tiêu chuẩn (phần 12.1.3) và
 - quan hệ với nhau để cho thấy cách các khuôn mẫu cộng tác để giải quyết vấn đề trên một miền ứng dụng.
- một ngôn ngữ khuôn mẫu tương tự như một sách hướng dẫn siêu văn bản để giải quyết vấn đề trong một miền ứng dụng cụ thể.
 - Các miền vấn đề được xem xét đầu tiên được mô tả theo thứ bậc, bắt đầu với các vấn đề thiết kế rộng liên quan đến miền và sau đó tinh chỉnh từng vấn đề rộng vào mức độ trừu tượng thấp hơn.

Thiết kế dựa trên khuôn mẫu

- Một nhà thiết kế phần mềm bắt đầu với một mô hình yêu cầu (hoặc là rõ ràng hay ngụ ý) trình bày một thể hiện trừu tượng của hệ thống.
- Các mô hình yêu cầu mô tả các tập vấn đề, thiết lập ngữ cảnh, và xác định các hệ thống của các nguồn lực.
- Sau đó ...



Thiết kế dựa trên khuôn mẫu



Suy nghĩ trong khuôn mẫu

- Shalloway và Trott [Sha05] đề xuất các phương pháp sau đây cho phép một nhà thiết kế suy nghĩ trong mô hình:
 - 1. Hãy chắc chắn rằng bạn hiểu được bức tranh lớn - ngữ cảnh trong đó phần mềm được xây dựng. Các yêu cầu mô hình phải truyền thông điệp này cho bạn.
 - 2. Xem xét các bức tranh lớn, trích xuất các khuôn mẫu có mặt ở mức độ trừu tượng đó.
 - 3. Bắt đầu thiết kế của bạn với khuôn mẫu "bức tranh lớn" mà thiết lập một ngữ cảnh hoặc bộ khung cho công việc thiết kế sau này.
 - 4. "Làm việc hướng nội từ ngữ cảnh" [Sha05] tìm kiếm các khuôn mẫu ở các mức trừu tượng thấp hơn góp phần vào các giải pháp thiết kế.
 - 5. Lặp lại các bước 1-4 cho đến khi thiết kế hoàn chỉnh được tạo ra.
 - 6. Hoàn thiện thiết kế bằng cách thích ứng từng mô hình vào các chi tiết cụ thể của phần mềm bạn đang cố gắng xây dựng.

Nhiệm vụ thiết kế-I

- Kiểm tra các mô hình yêu cầu và phát triển một hệ thống phân cấp vấn đề.
- Xác định nếu một ngôn ngữ khuôn mẫu đáng tin cậy đã được phát triển cho miền vấn đề.
- Bắt đầu với một vấn đề lớn, xác định một hoặc nhiều khuôn mẫu kiến trúc có sẵn cho nó.
- Sử dụng sự cộng tác được cung cấp cho các khuôn mẫu kiến trúc, kiểm tra hệ thống phụ hoặc các vấn đề mức thành phần và tìm kiếm các khuôn mẫu thích hợp để giải quyết chúng.
- Lặp lại các bước từ 2 đến 5 cho đến khi tất cả các vấn đề lớn đã được giải quyết.

Nhiệm vụ thiết kế-II

- Nếu vấn đề thiết kế giao diện người dùng đã được phân lập (điều này là hầu như luôn luôn như vậy), hãy tìm kiếm các giao diện người dùng trong kho mẫu thiết kế để tìm ra khuôn mẫu phù hợp.
- Bất kể mức độ trừu tượng của nó, nếu một ngôn ngữ khuôn mẫu và / hoặc kho khuôn mẫu hoặc khuôn mẫu riêng cho thấy sự hứa hẹn, hãy so sánh các vấn đề cần giải quyết đối với các khuôn mẫu hiện có.
- Hãy chắc chắn để tinh chỉnh các thiết kế như là nó có nguồn gốc từ các khuôn mẫu sử dụng các tiêu chí chất lượng thiết kế như một hướng dẫn.

Bảng sắp xếp khuôn mẫu

	Database	Application	Implementation	Infrastructure
<i>Data/Content</i>				
Problem statement ...	PatternName (s)		PatternName (s)	
Problem statement ...		PatternName (s)		PatternName (s)
Problem statement ...	PatternName (s)			PatternName (s)
<i>Architecture</i>				
Problem statement ...		PatternName (s)		
Problem statement ...		PatternName (s)		PatternName (s)
Problem statement ...				
<i>Component-level</i>				
Problem statement ...		PatternName (s)	PatternName (s)	
Problem statement ...				PatternName (s)
Problem statement ...		PatternName (s)	PatternName (s)	
<i>User Interface</i>				
Problem statement ...		PatternName (s)	PatternName (s)	
Problem statement ...		PatternName (s)	PatternName (s)	
Problem statement ...		PatternName (s)	PatternName (s)	

Lỗi thiết kế thông thường

- Không đủ thời gian được sử dụng để hiểu được vấn đề cơ bản, ngữ cảnh và các nguồn lực của mình, và như một hệ quả, bạn chọn một khuôn mẫu có vẻ đúng, nhưng không phù hợp với giải pháp được yêu cầu.
- Một khi chọn sai khuôn mẫu, bạn không chịu nhìn ra vấn đề và sử dụng khuôn mẫu gượng ép.
- Trong các trường hợp khác, các vấn đề có nguồn lực không được xem xét bởi các khuôn mẫu bạn đã chọn, dẫn đến kết quả nghèo nàn.
- Đôi khi một khuôn mẫu được áp dụng quá thô thiển và sự thích nghi cần thiết cho không gian vấn đề của bạn không được thực hiện.

Khuôn mẫu kiến trúc

- Ví dụ: mỗi nhà (và mọi phong cách kiến trúc đối với nhà ở) sử dụng một khuôn mẫu nhà bếp.
- Khuôn mẫu nhà bếp và các khuôn mẫu mà nó cộng tác giải quyết các vấn đề liên quan tới việc lưu trữ và chuẩn bị thức ăn, các công cụ cần thiết để thực hiện những nhiệm vụ, và các quy tắc cho vị trí của các công cụ liên quan đến công việc trong phòng. Ngoài ra, các mô hình có thể giải quyết các vấn đề liên quan đến mặt bàn, chiếu sáng, công tắc tường, sàn nhà...
- Rõ ràng, có nhiều hơn một thiết kế cho một nhà bếp, thường quyết định bởi bối cảnh và hệ thống của lực lượng. Tuy nhiên, mỗi thiết kế có thể được hình thành trong bối cảnh của "giải pháp" được đề xuất bởi các khuôn mẫu nhà bếp.

Kho khuôn mẫu

- Có rất nhiều nguồn cho các khuôn mẫu thiết kế có sẵn trên web. Một số khuôn mẫu có thể được lấy từ các ngôn ngữ khuôn mẫu xuất bản riêng, trong khi những cái khác đang có sẵn như là một phần của một cổng hoặc kho khuôn mẫu.
- Một danh sách các kho khuôn mẫu được thể hiện trong mục 12.3

Khuôn mẫu mức thành phần

- Các khuôn mẫu thiết kế mức thành phần cung cấp một giải pháp đã được chứng minh để giải quyết một hoặc nhiều vấn đề phụ phát sinh từ các mô hình yêu cầu. Trong nhiều trường hợp, các khuôn mẫu thiết kế của loại hình này tập trung vào một số yếu tố chức năng của một hệ thống.
- Ví dụ, ứng dụng **SafeHomeAssured.com** phải giải quyết các vấn đề thiết kế phụ sau đây:
Làm thế nào chúng ta có thể có được thông số kỹ thuật sản phẩm và các thông tin liên quan cho bất kỳ thiết bị **SafeHome** nào?

Khuôn mẫu mức thành phần

- Sau khi đề ra các vấn đề phụ cần giải quyết, xem xét ngữ cảnh và hệ thống các lực lượng có ảnh hưởng đến các giải pháp.
- Kiểm tra các trường hợp sử dụng các yêu cầu mô hình thích hợp, các đặc điểm kỹ thuật cho một thiết bị SafeHome (ví dụ, một bộ cảm biến an ninh hoặc camera) được sử dụng cho mục đích thông tin của người tiêu dùng.
 - Tuy nhiên, các thông tin khác có liên quan đến các đặc điểm kỹ thuật (ví dụ, giá cả) có thể được sử dụng khi chức năng thương mại điện tử được chọn.
- Các giải pháp cho vấn đề phụ bao gồm một sự tìm kiếm. Kể từ khi tìm kiếm trở thành vấn đề rất phổ biến, không có gì ngạc nhiên khi có rất nhiều các khuôn mẫu liên quan đến tìm kiếm.
- Xem Phần 12.4

Khuôn mẫu giao diện người dùng (UI Patterns)

- **Giao diện người dùng tổng thể (Whole UI).** Cung cấp hướng dẫn thiết kế cho các cấu trúc cấp cao nhất và điều hướng trong suốt toàn bộ giao diện.
- **Bố cục trang.** Giải quyết các tổ chức chung của trang (cho các trang web) hay hiển thị màn hình riêng biệt (cho các ứng dụng tương tác)
- **Biểu mẫu và đầu vào.** Xem xét nhiều kỹ thuật thiết kế cho việc hoàn thành đầu vào dạng biểu mẫu.
- **Bảng.** Cung cấp hướng dẫn thiết kế cho việc tạo và xử lý dữ liệu dạng bảng dưới mọi dạng.
- **Xử lý dữ liệu trực tiếp.** Address data editing, modification, and transformation.
- **Điều hướng.** Hỗ trợ người dùng định hướng xuyên suốt menu phân cấp, các trang web và màn hình hiển thị tương tác.
- **Tìm kiếm.** Cho phép tính năng tìm kiếm nội dung cụ thể thông qua các thông tin được duy trì trong một trang Web hoặc được lưu trữ trong cơ sở dữ liệu có thể truy cập thông qua một ứng dụng tương tác..
- **Các phần tử trang.** Thực hiện các phần tử cụ thể của một trang web hoặc màn hình hiển thị.
- **Thương mại điện tử.** Tùy vào các trang web, các mô hình triển khai các yếu tố định kỳ của các ứng dụng thương mại điện tử.

Khuôn mẫu WebApp

- **Khuôn mẫu kiến trúc thông tin** liên quan đến cấu trúc tổng thể của không gian thông tin, và những cách mà người dùng sẽ tương tác với thông tin.
- **Khuôn mẫu điều hướng** xác định cấu trúc liên kết chuyển hướng, chẳng hạn như hệ thống phân cấp, vòng, tours,...
- **Khuôn mẫu tương tác** đóng góp vào việc thiết kế giao diện người dùng. Các khuôn mẫu loại này chỉ ra cách giao diện người dùng thông báo về hệ quả của một hành động cụ thể; làm thế nào một người dùng mở rộng nội dung dựa trên việc sử dụng ngữ cảnh và mong muốn của người dùng; làm thế nào để mô tả tốt nhất đích đến đằng sau một liên kết; làm thế nào để thông báo cho người dùng về tình trạng của một tương tác đang thực hiện, và các vấn đề liên quan đến giao diện.
- **Khuôn mẫu trình diễn** hỗ trợ trong việc trình bày các nội dung cho người sử dụng thông qua giao diện. Các khuôn mẫu loại này giải quyết vấn đề tổ chức các chức năng điều khiển giao diện người dùng cho khả năng sử dụng tốt hơn; vấn đề hiển thị các mối quan hệ giữa một hành động giao diện và các đối tượng nội dung nó ảnh hưởng, và vấn đề thiết lập hệ thống phân cấp nội dung hiệu quả.
- **Khuôn mẫu chức năng** xác định các quy trình công việc, hành vi, xử lý, truyền thông, và các yếu tố khác trong một thuật toán WebApp.

Thiết kế mức độ chi tiết

- Khi một vấn đề liên quan đến vấn đề "bức tranh lớn", cố gắng phát triển các giải pháp (và sử dụng các khuôn mẫu có liên quan) tập trung vào bức tranh lớn.
- Ngược lại, khi sự tập trung là rất hẹp (ví dụ, duy nhất chọn một mục từ một tập hợp nhỏ của năm mặt hàng hoặc ít hơn), các giải pháp (và các khuôn mẫu tương ứng) có mục tiêu khá hẹp.
- Xét về mức độ chi tiết, khuôn mẫu có thể được mô tả theo các mức sau:

Thiết kế mức độ chi tiết

- **Khuôn mẫu kiến trúc.** Mức độ trừu tượng này thường sẽ liên quan đến các khuôn mẫu xác định cấu trúc tổng thể của WebApp, chỉ ra các mối quan hệ giữa các thành phần khác nhau, và xác định các quy tắc để xác định mối quan hệ giữa các yếu tố (các trang, gói, thành phần, hệ thống con) của kiến trúc.
- **Khuôn mẫu thiết kế.** Những khuôn mẫu này chỉ ra một yếu tố cụ thể của thiết kế như một tập hợp của các thành phần để giải quyết một số vấn đề thiết kế, mối quan hệ giữa các yếu tố trên một trang, hoặc các cơ chế ảnh hưởng đến giao tiếp giữa các thành phần. Một ví dụ có thể là khuôn mẫu Broadsheet cho cách bố trí của một trang chủ WebApp.
- **Khuôn mẫu thành phần.** Mức độ trừu tượng này liên quan đến các yếu tố quy mô nhỏ lẻ của WebApp. Các ví dụ bao gồm các yếu tố tương tác cá nhân, các mục chuyển hướng, hoặc các yếu tố chức năng.