

"Cơ bản về UML"

Ngôn ngữ mô hình hóa trực quan được sử dụng để tạo ra các mô hình phân tích và thiết kế hệ thống phần mềm hướng đối tượng. Xuất hiện tương đối gần đây, từ năm 1989 đến 1997, những ngôn ngữ này đã có lịch sử phát triển tiêu biểu.

Hiện tại, có ba thể hệ ngôn ngữ mô hình hóa trực quan. Và nếu thể hệ đầu tiên được hình thành bởi 10 ngôn ngữ thì số lượng thể hệ thứ hai đã vượt quá 50 ngôn ngữ. Trong số các ngôn ngữ phổ biến nhất của thể hệ thứ 2 là: ngôn ngữ Booch (G. Booch), ngôn ngữ Rumbaugh (J. Rumbaugh), ngôn ngữ Jacobson (I.

Jacobson), ngôn ngữ Coad-Yourdon, ngôn ngữ Shlaer-Mellor, v.v. Mỗi ngôn ngữ đều có phương thức diễn đạt riêng, tập trung vào cú pháp

và ngữ nghĩa riêng, nói cách khác, nó tự nhận là ngôn ngữ duy nhất. Kết quả là các nhà phát triển (và người dùng các ngôn ngữ này) không còn hiểu nhau nữa. Có một nhu cầu cấp thiết để thống nhất các ngôn ngữ.

Ý tưởng thống nhất đã dẫn đến sự xuất hiện của các ngôn ngữ thể hệ thứ 3. Hợp nhất được sử dụng làm ngôn ngữ tiêu chuẩn thể hệ thứ ba. Ngôn ngữ mô hình hóa (UML), được tạo ra vào năm 1994-1997 (các nhà phát triển chính là ba "bạn thân" G. Buch, J. Rambo, I. Jacobson).

Ngôn ngữ mô hình thống nhất

UML là ngôn ngữ tiêu chuẩn để viết các mô hình phân tích, thiết kế và triển khai các hệ thống phần mềm hướng đối tượng. UML có thể được sử dụng để trực quan hóa, xác định, xây dựng và ghi lại kết quả của các dự án phần mềm.

UML không phải là ngôn ngữ lập trình trực quan, nhưng các mô hình của nó dịch trực tiếp thành văn bản trong ngôn ngữ lập trình và thậm chí thành bảng cho cơ sở dữ liệu quan hệ.

Từ vựng UML bao gồm ba loại khối xây dựng: đồ vật, mối quan hệ, sơ đồ.

Các đối tượng là sự trừu tượng hóa, là các phần tử cơ bản trong một mô hình, các mối quan hệ kết nối các đối tượng này, sơ đồ nhóm các tập hợp đối tượng.

Các chủ đề trong UML

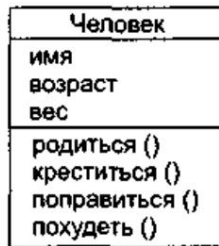
Có bốn loại mục trong UML:

- hạng mục kết cấu;
- các mục hành vi;
- nhóm các đối tượng;
- các mục giải thích.

Những mục này là hướng đối tượng cơ bản khối xây dựng. Chúng được sử dụng để viết mô hình.

Đối tượng cấu trúc là danh từ trong mô hình UML. Chúng đại diện cho các phần tính của mô hình - các phần tử khái niệm hoặc vật lý. Chúng tôi liệt kê tám loại đối tượng cấu trúc.

1. Lớp - mô tả một tập hợp các đối tượng có cùng thuộc tính, hoạt động, mối quan hệ và ngữ nghĩa (ý nghĩa). Một lớp thực hiện một hoặc nhiều giao diện. Như thể hiện trong hình. 1, về mặt đồ họa, một lớp được hiển thị dưới dạng hình chữ nhật, thường bao gồm các phần có tên, thuộc tính (thuộc tính) và hoạt động.



Cơ. 1. Lớp học

2. Giao diện - một tập hợp các hoạt động xác định các dịch vụ của một lớp hoặc thành phần. Giao diện mô tả hành vi của một phần tử có thể nhìn thấy từ bên ngoài. Một giao diện có thể đại diện cho toàn bộ dịch vụ của một lớp hoặc thành phần hoặc một phần của các dịch vụ đó. Một giao diện xác định một tập hợp các đặc tả hoạt động (chữ ký của chúng), chứ không phải là một tập hợp các triển khai hoạt động. Về mặt đồ họa, giao diện được mô tả dưới dạng một vòng tròn có tên, như trong Hình 2. 2. Tên giao diện thường bắt đầu bằng chữ "I". Giao diện hiếm khi được hiển thị độc lập. Nó thường được gắn vào một lớp hoặc thành phần thực hiện giao diện.

3. Hợp tác xác định sự tương tác và là tập hợp các vai trò và các yếu tố khác phối hợp với nhau để tạo ra hành vi tập thể phức tạp hơn tổng đơn giản của tất cả các yếu tố. Vì vậy, hợp tác có cả khía cạnh cấu trúc và hành vi. Một lớp nhất định có thể tham gia vào một số hợp tác. Những sự hợp tác này thể hiện việc triển khai các mô hình hình thành nên hệ thống. Như thể hiện trong hình. 3, sự hợp tác được mô tả bằng đồ họa dưới dạng hình elip có chấm, trong đó

tên của cô ấy được bao gồm.



Cơ. 3. Hợp tác

4. Tác nhân - một tập hợp các vai trò đã được thống nhất mà người dùng có thể thực hiện khi tương tác với hệ thống (các thành phần Ca sử dụng của nó).

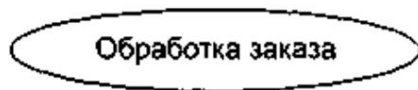
Mỗi vai trò yêu cầu một hành vi cụ thể từ hệ thống. Như thể hiện trong hình. 4, nam diễn viên được miêu tả là một người đàn ông đi dây có tên.



Cơ. 4. Diễn viên

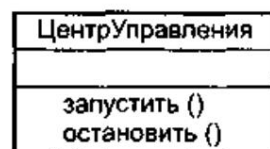
5. Phần tử Ca sử dụng - mô tả một chuỗi hành động (hoặc một số chuỗi) được hệ thống thực hiện vì lợi ích của một tác nhân riêng lẻ và tạo ra kết quả mà tác nhân đó có thể nhìn thấy. Trong mô hình, phần tử Ca sử dụng được sử dụng để cấu trúc các mục hành vi. Phần tử Use Case được thực hiện bằng sự hợp tác. Như thể hiện trong hình. 5, phần tử Ca sử dụng được mô tả dưới dạng hình elip mà nó phù hợp với

Tên.



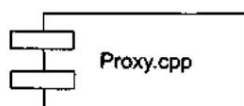
Cơ. 5. Các phần tử Use Case

6. Lớp hoạt động - lớp có các đối tượng có một hoặc nhiều tiến trình (hoặc luồng) và do đó có thể bắt đầu hoạt động điều khiển. Một lớp hoạt động tương tự như một lớp thông thường ngoại trừ việc các đối tượng của nó hoạt động đồng thời với các đối tượng của các lớp khác. Như thể hiện trong hình. 6, lớp hoạt động được mô tả dưới dạng một hình chữ nhật dày, thường bao gồm tên, thuộc tính (thuộc tính) và hoạt động.



Cơ. 6. Các lớp học tích cực

7. Thành phần - một phần vật lý và có thể thay thế được của hệ thống tương ứng với một bộ giao diện và cung cấp cách triển khai bộ giao diện đó. Hệ thống bao gồm cả các thành phần là kết quả của quá trình phát triển (các tệp mã nguồn) và các loại thành phần khác nhau được sử dụng (các thành phần COM+, Đậu Java). Thông thường, một thành phần là một gói vật lý gồm nhiều phần tử logic khác nhau (lớp, giao diện và sự cộng tác). Như thể hiện trong hình. 7, thành phần được mô tả dưới dạng hình chữ nhật theo thẻ, thường bao gồm tên.



Cơ. 7. Linh kiện

8. Nút là một phần tử vật lý tồn tại trong quá trình vận hành hệ thống và đại diện cho một tài nguyên, thường có bộ nhớ và khả năng xử lý. Một nút chứa một tập hợp các thành phần có thể di chuyển từ nút này sang nút khác. Như thể hiện trong hình. 8, một nút được mô tả dưới dạng một khối có tên.



Cơ. 8. Nút thất

Các đối tượng hành vi là các phần động của mô hình UML. Chúng là những động từ mẫu, biểu thị hành vi trong thời gian và không gian. Có hai loại mặt hàng chính

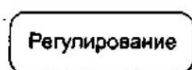
hành vi.

1. Tương tác là hành vi bao gồm một tập hợp các thông điệp được trao đổi giữa một tập hợp các đối tượng trong một bối cảnh cụ thể để đạt được một mục tiêu cụ thể. Tương tác có thể xác định động lực của cả một tập hợp các đối tượng và một hoạt động riêng biệt. Các yếu tố tương tác là thông điệp, chuỗi hành động (hành vi do thông báo gây ra) và kết nối (kết nối giữa các đối tượng). Như thể hiện trong hình. 9, thông báo được mô tả dưới dạng một đường dẫn có tên hoạt động của nó.



Cơ. 9. Tin nhắn

2. Máy trạng thái - hành vi xác định một chuỗi các trạng thái hoặc tương tác của đối tượng được thực hiện trong quá trình tồn tại của nó để phản hồi các sự kiện (và có tính đến trách nhiệm đối với các sự kiện này). Máy trạng thái có thể được sử dụng để xác định hành vi của một lớp riêng lẻ hoặc sự cộng tác của các lớp. Các phần tử của máy trạng thái hữu hạn là các trạng thái, chuyển tiếp (từ trạng thái này sang trạng thái khác), sự kiện (đối tượng gây ra chuyển đổi) và hành động (phản ứng với chuyển đổi). Như thể hiện trong hình. 10, một trạng thái được mô tả dưới dạng một hình chữ nhật tròn, thường bao gồm tên và các trạng thái con của nó (nếu có).



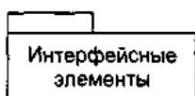
Cơ. 10. Kỳ

Hai phần tử này—tương tác và máy trạng thái—là các đối tượng hành vi cơ bản có thể được đưa vào các mô hình UML. Về mặt ngữ nghĩa, những phần tử này được liên kết với nhiều

các thành phần cấu trúc (chủ yếu là các lớp, sự cộng tác và đối tượng).

Nhóm các đối tượng là một phần tổ chức của các mô hình UML. Đây là những hộp mà mô hình có thể được đặt vào đó. Có một loại mục nhóm - một gói.

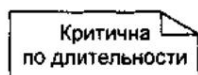
Gói là một cơ chế chung để phân phối các phần tử thành các nhóm. Túi có thể chứa các vật phẩm cấu trúc, vật phẩm hành vi và thậm chí các nhóm vật phẩm khác. Không giống như một thành phần (tồn tại trong thời gian chạy), gói là một khái niệm thuần túy mang tính khái niệm. Điều này có nghĩa là gói chỉ tồn tại trong quá trình phát triển. Như thể hiện trong hình. 11, gói được mô tả dưới dạng một thư mục có tab cho biết tên và đôi khi là nội dung của nó.



Cơ. 11. Gói hàng

Các đối tượng giải thích là phần giải thích của mô hình UML. Chúng là những ghi chú có thể được sử dụng để mô tả, giải thích và nhận xét về bất kỳ phần tử nào của mô hình. Có một loại mục giải thích - một ghi chú.

Lưu ý - Biểu tượng để hiển thị các hạn chế và nhận xét gắn liền với một phần tử hoặc tập hợp các phần tử. Như thể hiện trong hình. 12, ghi chú được mô tả dưới dạng một hình chữ nhật với một góc cong để chứa văn bản hoặc hình ảnh nhận xét.



Cơ. 12. Ghi chú

Các mối quan hệ trong UML

Có bốn loại mối quan hệ trong UML:

- 1) nghiên;
- 2) hiệp hội;
- 3) khái quát hóa;
- 4) thực hiện.

Những mối quan hệ này là nền tảng cơ bản của các mối quan hệ. Chúng được sử dụng khi viết mô hình.

1. Phụ thuộc là mối quan hệ ngữ nghĩa giữa hai đối tượng, trong đó sự thay đổi của một đối tượng (đối tượng độc lập) có thể ảnh hưởng đến ngữ nghĩa của đối tượng khác (đối tượng phụ thuộc). Như thể hiện trong hình. 13, sự phụ thuộc được mô tả dưới dạng đường chấm, có thể hướng tới một đối tượng độc lập và đôi khi có nhãn.



Cơ. 13. Sự phụ thuộc

2. Hiệp hội - một mối quan hệ cấu trúc mô tả một tập hợp các kết nối là các kết nối giữa các đối tượng. Tập hợp là một loại liên kết đặc biệt thể hiện mối quan hệ cấu trúc giữa tổng thể và các bộ phận của nó. Như thể hiện trong hình. 14, sự liên kết được mô tả như một đường liên nét, có thể định hướng, đôi khi được dán nhãn và thường bao gồm các đồ trang trí khác như quyền hạn và tên vai trò.



Cơ. 14. Hiệp hội

3. Khái quát hóa là mối quan hệ chuyên biệt hóa/khái quát hóa trong đó các đối tượng của phần tử chuyên biệt (con cháu, con) có thể thay thế các đối tượng của phần tử khái quát hóa (tổ tiên, cha mẹ). Nói cách khác, đứa trẻ chia sẻ cấu trúc và hành vi của cha mẹ. Như thể hiện trong hình. 15, phần khái quát hóa được mô tả dưới dạng một mũi tên đặc có đầu rỗng hướng về phía cha mẹ.



Cơ. 15. Khái quát hóa

4. Triển khai - ngữ nghĩa giữa các bộ phận loại, trong đó một bộ phận ^{thực thể} loại xác định một hợp đồng mà bộ phận loại khác cam kết thực hiện (các bộ phận loại bao gồm các lớp, giao diện, thành phần, các phần tử Ca sử dụng, sự hợp tác). Các mối quan hệ triển khai được sử dụng trong hai trường hợp: giữa các giao diện và các lớp (hoặc thành phần) triển khai chúng; giữa các phần tử Sử dụng

Trường hợp và sự hợp tác thực hiện chúng. Như thể hiện trong hình. 16, việc thực hiện được mô tả như một cái gì đó giữa khái quát hóa và

nghiên.



Cơ. 16. Triển khai

Sơ đồ trong UML

Sơ đồ là sự biểu diễn bằng đồ họa của một tập hợp các phần tử, thường được mô tả dưới dạng biểu đồ được kết nối của các đỉnh (đối tượng) và cung (mối quan hệ). Các sơ đồ được vẽ để trực quan hóa hệ thống từ các góc nhìn khác nhau, sau đó chúng được ánh xạ vào hệ thống. Thông thường, sơ đồ cung cấp cái nhìn không đầy đủ về các thành phần tạo nên một hệ thống. Mặc dù cùng một yếu tố có thể xuất hiện trong tất cả các biểu đồ nhưng trên thực tế nó chỉ xuất hiện trong một số biểu đồ. Về mặt lý thuyết, sơ đồ có thể chứa bất kỳ sự kết hợp nào giữa các đối tượng và mối quan hệ; trong thực tế, chúng bị giới hạn ở một số lượng nhỏ các kết hợp tương ứng với năm biểu diễn của kiến trúc PS. Vì lý do này, UML bao gồm chín loại sơ đồ:

- 1) sơ đồ lớp;
- 2) sơ đồ đối tượng;
- 3) Sơ đồ Use Case (sơ đồ tiền lệ);
- 4) sơ đồ trình tự;
- 5) sơ đồ hợp tác (hợp tác);
- 6) sơ đồ trạng thái;
- 7) sơ đồ hoạt động;
- 8) sơ đồ thành phần;
- 9) sơ đồ vị trí (triển khai).

Sơ đồ lớp hiển thị một tập hợp các lớp, giao diện, sự cộng tác và mối quan hệ của chúng. Khi mô hình hóa các hệ thống hướng đối tượng, sơ đồ lớp là sơ đồ được sử dụng phổ biến nhất. Sơ đồ lớp cung cấp một cái nhìn thiết kế tĩnh của một hệ thống. Sơ đồ lớp, bao gồm các lớp hoạt động, cung cấp một cái nhìn tĩnh về các quy trình hệ thống.

Sơ đồ đối tượng hiển thị một tập hợp các đối tượng và mối quan hệ của chúng. Sơ đồ đối tượng thể hiện một "ảnh chụp nhanh" tĩnh của các thể hiện đối tượng được tìm thấy trong sơ đồ lớp. Giống như sơ đồ lớp, các sơ đồ này cung cấp chế độ xem thiết kế tĩnh hoặc chế độ xem tĩnh về các quy trình của hệ thống.

Sơ đồ Ca sử dụng hiển thị một tập hợp các thành phần Ca sử dụng, tác nhân và mối quan hệ của chúng. Sử dụng sơ đồ ca sử dụng Chế độ xem Ca sử dụng tĩnh được tạo cho hệ thống. Những sơ đồ này đặc biệt quan trọng khi tổ chức và lập mô hình hoạt động của hệ thống cũng như xác định các yêu cầu của khách hàng đối với hệ thống.

Sơ đồ trình tự và sơ đồ cộng tác là các loại sơ đồ tương tác.

Sơ đồ tương tác thể hiện sự tương tác liên quan đến một tập hợp các đối tượng và mối quan hệ của chúng cũng như các thông điệp được gửi giữa các đối tượng. Sơ đồ tương tác cung cấp một cái nhìn năng động về một hệ thống.

Sơ đồ trình tự là sơ đồ tương tác làm nổi bật thứ tự của các thông điệp theo thời gian.

Sơ đồ cộng tác (sơ đồ hợp tác) là sơ đồ tương tác nêu bật cách tổ chức cấu trúc của các đối tượng gửi và nhận tin nhắn. Sơ đồ trình tự và sơ đồ cộng tác là các sơ đồ đẳng cấu, có nghĩa là một sơ đồ này có thể được chuyển đổi thành một sơ đồ khác.

Sơ đồ trạng thái hiển thị một máy trạng thái, biểu thị các trạng thái, chuyển tiếp, sự kiện và hành động. Sơ đồ trạng thái cung cấp một cái nhìn năng động về một hệ thống. Chúng đặc biệt quan trọng khi mô hình hóa hành vi của một giao diện, lớp hoặc sự cộng tác. Các sơ đồ này nêu bật hành vi của đối tượng hướng sự kiện, đặc biệt hữu ích khi lập mô hình các hệ thống phản ứng.

Sơ đồ hoạt động là một loại sơ đồ trạng thái đặc biệt thể hiện luồng từ hoạt động này sang hoạt động khác trong một hệ thống. Sơ đồ hoạt động cung cấp một cái nhìn năng động về một hệ thống. Chúng đặc biệt quan trọng khi mô hình hóa chức năng của hệ thống và làm nổi bật luồng điều khiển giữa các đối tượng.

Sơ đồ thành phần cho thấy cách tổ chức của một tập hợp các thành phần và sự phụ thuộc giữa các thành phần. Các sơ đồ thành phần cung cấp một sự trình bày tĩnh về việc triển khai hệ thống. Chúng liên quan đến sơ đồ lớp theo nghĩa là một thành phần thường ánh xạ một hoặc nhiều lớp, giao diện hoặc sự cộng tác.

Sơ đồ vị trí (sơ đồ triển khai) hiển thị cấu hình của các nút xử lý thời gian chạy cũng như các thành phần tồn tại trong đó. Sơ đồ bố trí cung cấp một cái nhìn tĩnh về cách bố trí của một hệ thống. Chúng liên quan đến các sơ đồ thành phần theo nghĩa là một nút thường bao gồm một

hoặc một số thành phần.

Trong một thời gian khá lâu, trong cả quá trình thiết kế cấu trúc truyền thống và hướng đối tượng, các nhà phát triển đã sử dụng các kịch bản điển hình để giúp họ hiểu rõ hơn các yêu cầu hệ thống. Những chữ viết này được diễn giải rất không chính thức - chúng hầu như luôn được sử dụng và cực kỳ hiếm khi được ghi lại. Ivar Jacobson lần đầu tiên đưa ra khái niệm "tùy chọn"

"trường hợp sử dụng" và mang lại cho nó tầm quan trọng đến mức nó trở thành yếu tố chính của việc lập kế hoạch và phát triển dự án. Trường hợp sử dụng là một chuỗi các hành động (giao dịch) được hệ thống thực hiện để đáp ứng với một sự kiện do một số đối tượng bên ngoài (tác nhân) khởi xướng. cách sử dụng mô tả sự tương tác điển hình giữa người dùng và hệ thống.

Ca sử dụng Một cấu trúc hoặc thành

phần tiêu chuẩn của ngôn ngữ UML, một ca sử dụng được sử dụng để chỉ định hành vi chung của một hệ thống hoặc bất kỳ thực thể miền nào khác mà không xem xét cấu trúc bên trong của thực thể đó. Mỗi ca sử dụng xác định một chuỗi các hành động phải được hệ thống thực hiện khi nó tương tác với tác nhân tương ứng. Sơ đồ tùy chọn có thể được bổ sung bằng văn bản giải thích cho thấy ý nghĩa hoặc ngữ nghĩa của các thành phần của nó. Văn bản giải thích này được gọi là ghi chú hoặc kịch bản.

Một trường hợp sử dụng cụ thể được biểu thị trên sơ đồ bằng một hình elip, chứa tên viết tắt hoặc tên của nó ở dạng động từ với các từ giải thích (Hình 17).



Cơ. 17 Biểu diễn đồ họa của ca sử dụng

Mục đích của ca sử dụng là xác định một khía cạnh hoặc một đoạn hành vi hoàn chỉnh của một thực thể nào đó mà không tiết lộ cấu trúc bên trong của thực thể đó. Một thực thể như vậy có thể là hệ thống nguồn hoặc bất kỳ phần tử nào khác của mô hình có hành vi riêng, như hệ thống con hoặc lớp trong mô hình hệ thống.

Mỗi trường hợp sử dụng tương ứng với một dịch vụ riêng biệt cung cấp một thực thể hoặc hệ thống được mô hình hóa theo yêu cầu của người dùng (tác nhân), tức là xác định cách sử dụng thực thể này.

Một dịch vụ được khởi tạo theo yêu cầu của người dùng là một chuỗi hành động hoàn chỉnh. Điều này có nghĩa là sau khi hệ thống xử lý xong yêu cầu của người dùng, nó phải trở về trạng thái ban đầu, sẵn sàng thực hiện các yêu cầu tiếp theo.

Các trường hợp sử dụng mô tả không chỉ các tương tác giữa người dùng và thực thể mà còn mô tả phản ứng của thực thể khi nhận được các tín hiệu riêng lẻ từ người dùng và nhận thức về những tín hiệu đó bên ngoài thực thể. Các trường hợp sử dụng có thể bao gồm mô tả về các cách cụ thể để triển khai dịch vụ và các tình huống ngoại lệ khác

nhau, chẳng hạn như xử lý đúng các lỗi hệ thống. Toàn bộ tập hợp các trường hợp sử dụng sẽ xác định tất cả các khía cạnh có thể có của hành vi dự kiến của hệ thống. Để thuận tiện, nhiều trường hợp sử dụng có thể được coi là một gói riêng biệt. Tác nhân là bất kỳ thực thể nào bên ngoài hệ thống được mô hình hóa, tương tác với hệ thống và sử dụng chức năng của nó để đạt được các mục tiêu nhất định hoặc giải quyết các vấn đề cụ thể.

Trong trường hợp này, các tác nhân đóng vai trò chỉ định một tập hợp các vai trò nhất quán mà người dùng có thể thực hiện trong quá trình tương tác với hệ thống được thiết kế. Mỗi tác nhân có thể được coi là một vai trò riêng biệt liên quan

đến một trường hợp sử dụng cụ thể. Ký hiệu đồ họa tiêu chuẩn của một diễn viên trên sơ đồ là một bức tượng nhỏ "người đàn ông nhỏ", bên dưới ghi tên cụ thể của diễn viên (Hình 18).



Cơ. 18 Chỉ định đồ họa của một diễn viên

Trong một số trường hợp, một tác nhân có thể được biểu diễn bằng một hình chữ nhật lớp với từ khóa "tác nhân" và các phần tử cấu thành thông thường lớp học.

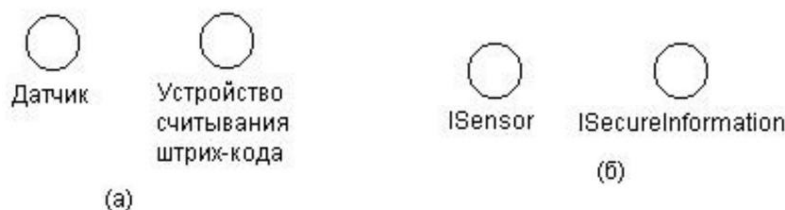
Tên diễn viên phải được viết bằng chữ in hoa và tuân theo nguyên tắc đặt tên cho các loại và lớp mô hình. Trong trường hợp này, biểu tượng của một tác nhân riêng lẻ liên kết mô tả tương ứng của tác nhân đó với một tên cụ thể. Khuyến nghị rằng tên của các tác nhân trừu tượng, giống như các thành phần trừu tượng khác của ngôn ngữ UML, nên được viết bằng chữ in nghiêng. Giao diện được sử dụng để chỉ định các tham số mô hình có thể nhìn thấy từ bên ngoài mà không chỉ định cấu trúc bên trong

của chúng. Trong UML, giao diện là một bộ phân loại và chỉ mô tả một phần giới hạn hành vi của thực thể được mô hình hóa.

Trong sơ đồ ca sử dụng, giao diện được mô tả như một vòng tròn nhỏ, bên cạnh có viết tên anh ấy (Hình 19, a).

Tên có thể là một danh từ mô tả thông tin hoặc dịch vụ liên quan (ví dụ: "cảm biến", "còi báo động", "máy quay video") nhưng thường là một dòng văn bản (ví dụ: "truy vấn cơ sở dữ liệu", "biểu mẫu đầu vào" ", tín hiệu âm thanh "thiết bị"").

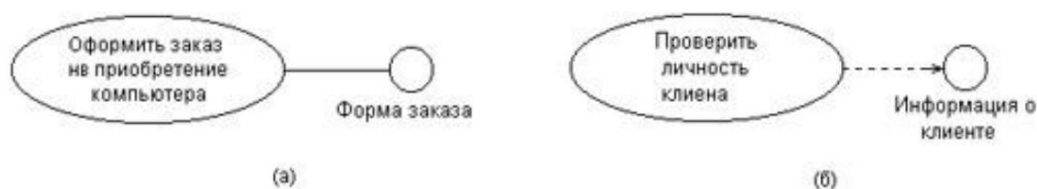
Nếu tên viết bằng tiếng Anh thì phải bắt đầu bằng chữ I in hoa, ví dụ ISecureInformation, ISensor (Hình 19, b).



Cơ. 19. Biểu diễn đồ họa các giao diện bằng sơ đồ trường hợp sử dụng

Ký hiệu đồ họa cho một giao diện cụ thể có thể được kết nối trên sơ đồ bằng một đường liền nét với trường hợp sử dụng hỗ trợ nó. Đường liền nét trong trường hợp này cho biết thực tế là trường hợp sử dụng được liên kết với giao diện phải triển khai tất cả các hoạt động cần thiết cho giao diện này và có thể hơn thế nữa (Hình 20a). Ngoài ra, các giao diện có thể được kết nối với các trường hợp sử dụng bằng đường chấm chấm có mũi tên (Hình 4.4, b), nghĩa là

Ca sử dụng nhằm mục đích chỉ định dịch vụ cần thiết để triển khai một giao diện nhất định.



См. 20. Biểu diễn đồ họa về mối quan hệ giữa các giao diện và trường hợp sử dụng

Từ quan điểm phân tích hệ thống, giao diện không chỉ tách biệt đặc điểm kỹ thuật của các hoạt động hệ thống khỏi việc triển khai chúng mà còn xác định các ranh giới chung của hệ thống được

thiết kế. Sau đó, giao diện có thể được làm rõ bằng cách chỉ ra rõ ràng những hoạt động xác định một khía cạnh cụ thể của hành vi hệ thống.

Trong trường hợp này, nó được vẽ dưới dạng hình chữ nhật lớp với từ khóa "giao diện" trong phần tên, phần thuộc tính trống và phần hoạt động không trống. Tuy nhiên, các biểu diễn đồ họa tương tự được sử dụng trong sơ đồ lớp hoặc sơ đồ mô tả

hành vi của hệ thống được mô hình hóa. Tầm quan trọng của các giao diện nằm ở chỗ chúng xác định các nút kết nối trong hệ thống được thiết kế, điều này thực sự cần thiết để tổ chức công việc tập thể trong dự án. Hơn nữa, đặc tả giao diện

tạo điều kiện thuận lợi cho việc sửa đổi hệ thống hiện có một cách "không đau đớn" khi chuyển sang các giải pháp công nghệ mới. Trong trường hợp này, chỉ có việc thực hiện các hoạt động có thể thay đổi chứ không phải chức năng của chính hệ thống. Và điều này đảm bảo tính tương thích của các phiên bản tiếp theo của chương trình với phiên bản đầu tiên sử dụng công nghệ xoắn ốc để phát triển hệ thống phần mềm. Các ghi chú bằng ngôn ngữ UML nhằm mục đích đưa vào mô hình thông tin văn bản tùy ý có liên quan trực tiếp đến bối cảnh của dự

án đang được phát triển.

Thông tin này có thể bao gồm nhận xét của nhà phát triển (ví dụ: ngày và phiên bản của sơ đồ hoặc các thành phần riêng lẻ của nó), các hạn chế (ví dụ: về giá trị của các mối quan hệ riêng lẻ hoặc phiên bản thực thể) và các giá trị được gán nhãn. Trong sơ đồ ca sử dụng, ghi chú có thể là thông tin rất chung chung liên quan đến bối cảnh tổng thể.

hệ thống.

Về mặt đồ họa, các ghi chú được biểu thị bằng một hình chữ nhật có góc trên bên phải "cong" (Hình 21). Hình chữ nhật chứa văn bản của ghi chú.

Ghi chú có thể áp dụng cho bất kỳ phần tử nào của sơ đồ, trong trường hợp đó chúng được kết nối bằng một đường chấm. Nếu một ghi chú đề cập đến một số phần tử thì sẽ có nhiều dòng được rút ra từ ghi chú đó.

Tất nhiên, ghi chú có thể hiện diện không chỉ trên sơ đồ các trường hợp sử dụng mà còn trong các sơ đồ chuẩn khác.



Cơ. 21. Ví dụ về ghi chú trong ngôn ngữ UML

Các mối quan hệ trong sơ đồ ca sử dụng Có thể tồn tại nhiều mối quan hệ khác nhau giữa các thành phần của sơ đồ ca sử dụng mô tả cách các thể hiện của một số tác nhân và ca sử dụng tương tác với các thể hiện của các tác nhân và trường hợp khác. Một tác nhân duy nhất có thể tương tác với nhiều trường hợp sử dụng. Trong trường hợp này, tác nhân này truy cập một số dịch vụ của một hệ thống nhất định.

Đổi lại, một ca sử dụng có thể tương tác với một số tác nhân, cung cấp dịch vụ của bạn cho tất cả họ.

Cần lưu ý rằng hai trường hợp sử dụng được xác định cho cùng một thực thể không thể tương tác với nhau, vì mỗi trường hợp sử dụng đó mô tả độc lập một trường hợp sử dụng hoàn chỉnh cho thực thể đó. Hơn nữa, các trường hợp sử dụng luôn bao gồm một số tín hiệu hoặc thông báo khi tương tác với các tác nhân bên ngoài hệ thống. Đồng thời, có thể xác

định các cách khác để tương tác với các phần tử trong hệ thống. Ngôn ngữ UML có một số loại mối quan hệ tiêu chuẩn giữa

Tác nhân và trường hợp sử dụng:

- mối quan hệ liên kết

- Mở rộng mối quan hệ Khái quát hóa mối quan hệ Bao gồm mối
- quan hệ
-

Trong trường hợp này, các thuộc tính chung của các trường hợp sử dụng có thể được biểu diễn theo ba cách khác nhau, cụ thể là sử dụng các quan hệ mở rộng, khái quát hóa và bao hàm.

Quan hệ kết hợp Quan hệ kết

hợp là một trong những khái niệm cơ bản trong ngôn ngữ UML và được sử dụng ở mức độ này hay mức độ khác khi xây dựng tất cả các mô hình đồ họa của hệ thống dưới dạng sơ đồ chính tắc.

Khi được áp dụng cho sơ đồ ca sử dụng, nó dùng để chỉ ra vai trò cụ thể của một tác nhân trong một ca sử dụng cụ thể.

Nói cách khác, sự liên kết chỉ rõ các đặc điểm ngữ nghĩa của sự tương tác giữa các tác nhân và trường hợp sử dụng trong mô hình đồ họa của hệ thống. Do đó, mỗi quan hệ này thiết lập vai trò cụ thể của một tác nhân khi tương tác với một cá

thể ca sử dụng. Trong sơ đồ ca sử dụng, giống như trong các sơ đồ khác, mỗi quan hệ liên kết được biểu thị bằng một đường liền nét giữa tác nhân và ca sử dụng. Dòng này có thể có các ký hiệu bổ sung, chẳng hạn như tên và bội số (Hình 22).



См. 22. Một ví dụ về biểu diễn đồ họa của mối quan hệ liên kết giữa tác nhân và ca sử dụng

Tính bội số của liên kết được biểu thị bên cạnh việc chỉ định thành phần sơ đồ là thành viên của liên kết này. Tính bội số đặc trưng cho tổng số trường hợp cụ thể của một thành phần nhất định có thể hoạt động như các phần tử của một thành phần nhất định

hiệp hội.

Khi áp dụng cho sơ đồ ca sử dụng, bội số có ký hiệu đặc biệt ở dạng một hoặc nhiều số và dấu "*" (dấu hoa thị). Có thể là một ký tự đặc biệt Đối với các sơ đồ ca sử dụng, bốn dạng cơ bản phổ biến nhất để viết bội số của kết là: Một số nguyên không âm (bao gồm cả chữ một mối quan hệ liên số 0). Nhằm mục đích biểu thị tính đa dạng, được cố định chặt chẽ cho

-

phần tử của liên kết tương ứng. Trong trường hợp này, số lượng thực thể tác nhân hoặc ca sử dụng có thể đóng vai trò là các phần tử của mối quan hệ liên kết chính xác bằng số lượng đã chỉ định. Một ví dụ về hình thức ghi lại bội số của một liên kết này là chỉ ra bội số "1" cho tác nhân "Khách hàng Ngân hàng" (Hình

22). Mục nhập này có nghĩa là mỗi phiên bản của trường hợp sử dụng "Đăng ký khoản vay cho khách hàng ngân hàng" có thể có thành phần là một phiên bản duy nhất của tác nhân "Khách hàng ngân hàng".

Nói cách khác, khi đăng ký vay ngân hàng, bạn phải lưu ý rằng mỗi khoản vay cụ thể được cấp cho một khách hàng duy nhất của ngân hàng này.

- Hai số nguyên không âm cách nhau bởi hai dấu chấm và được viết là: "số thứ nhất .. số thứ hai". Ký hiệu UML này tương ứng với ký hiệu cho một tập hợp hoặc phạm vi số nguyên, được sử dụng trong một số ngôn ngữ lập trình để biểu thị ranh giới của một mảng các phần tử. Mục này nên được hiểu là một tập hợp các số nguyên không âm, theo thứ tự tăng dần:

$\{\text{số_đầu tiên}, \text{số_đầu tiên}+1, \text{số_đầu tiên}+2, \dots, \text{số_thứ hai}\}$. Rõ ràng, số thứ nhất phải nhỏ hơn số thứ hai theo nghĩa số học, trong khi số thứ nhất có thể bằng 0.

Một ví dụ về hình thức ghi bội số của một kết hợp này là "1..5". Ký hiệu này có nghĩa là số lượng phiên bản riêng lẻ của một thành phần nhất định có thể đóng vai trò là phần tử của một liên kết nhất định bằng với một số chưa biết trước đó từ tập hợp số nguyên $\{1, 2, 3, 4, 5\}$. Tình huống này có thể xảy ra, ví dụ, nếu tác nhân được coi là khách hàng của ngân hàng và ca sử dụng là

thủ tục mở tài khoản ngân hàng.

Trong trường hợp này, số lượng tài khoản riêng biệt cho mỗi khách hàng trong một ngân hàng nhất định, dựa trên một số cân nhắc bổ sung, có thể không quá 5.

Những cân nhắc bổ sung này chính xác là các yêu cầu bên ngoài liên quan đến hệ thống được thiết kế và được khách hàng xác định ở giai đoạn đầu của OOAP.

- Hai ký tự cách nhau bởi hai dấu chấm. Hơn nữa, số đầu tiên trong số chúng là số nguyên không âm hoặc 0, và số thứ hai là ký tự đặc biệt "*". Ở đây ký hiệu "*" biểu thị một số nguyên không âm hữu hạn tùy ý, giá trị của số này không xác định được tại thời điểm quan hệ liên kết tương ứng được chỉ định. Một ví dụ về hình thức ghi bội số của một kết hợp này là "2..*". Mục nhập có nghĩa là số lượng phiên bản riêng lẻ của một thành phần nhất định có thể hoạt động như các phần tử của một liên kết nhất định bằng

tới một số chưa biết trước đó từ tập hợp con các số tự nhiên: {2, 3, 4}. Ký tự duy nhất là "*", là viết tắt của khoảng

- "0..*". Trong trường hợp này, số lượng các thể hiện riêng biệt của một thành phần quan hệ kết hợp nhất định có thể là số nguyên không âm bất kỳ. Hơn nữa, 0 có nghĩa là đối với một số trường hợp của thành phần tương ứng, mối quan hệ liên kết này

có thể không hề diễn ra.

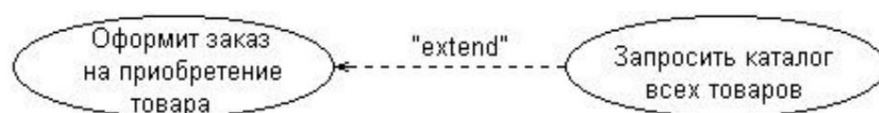
Một ví dụ về mục này là bội số của mối quan hệ liên kết cho trường hợp sử dụng "Đăng ký khoản vay cho khách hàng ngân hàng" (Hình 22). Ở đây, tính bội số có nghĩa là mỗi khách hàng cá nhân của ngân hàng có thể nhận được một số khoản vay cho mình, trong khi tổng số khoản vay của họ không được biết trước và không bị giới hạn dưới bất kỳ hình thức nào. Đồng thời, một số khách hàng có thể không có khoản vay nào được phát hành dưới tên của họ (tùy chọn giá trị 0). Nếu bội số quan hệ liên kết không được chỉ định thì theo mặc định

giá trị của nó được lấy là 1.

Mối quan hệ mở rộng Mối quan hệ mở

rộng xác định mối quan hệ giữa các phiên bản của một trường hợp sử dụng cụ thể và một trường hợp sử dụng tổng quát hơn có các thuộc tính được xác định dựa trên cách các phiên bản này được kết hợp với nhau. Trong siêu mô hình, mối quan hệ mở rộng mang tính định hướng và chỉ rõ rằng, đối với các trường hợp riêng lẻ của một trường hợp sử dụng, các điều kiện cụ thể được xác định cho trường hợp sử dụng đó được mở rộng phải được đáp ứng. Do đó, nếu có mối quan hệ mở rộng từ ca sử dụng A đến ca sử dụng B, thì điều này có nghĩa là các thuộc tính của một thể hiện của ca sử dụng B có thể được mở rộng do sự hiện diện của các thuộc tính trong ca sử dụng mở rộng A. Mối quan hệ mở rộng giữa các

trường hợp sử dụng được biểu thị bằng một đường chấm chấm có mũi tên (mối quan hệ phụ thuộc trường hợp) hướng tới một trường hợp sử dụng là phần mở rộng của trường hợp sử dụng ban đầu. Dòng mũi tên này được đánh dấu bằng từ khóa "mở rộng", như trong Hình. 23.



См. 23. Một ví dụ về biểu diễn đồ họa của mối quan hệ mở rộng giữa các trường hợp sử dụng

Mối quan hệ mở rộng đánh dấu thực tế là một trong các ca sử dụng có thể đính kèm một số

hành vi bổ sung được xác định cho tùy chọn khác
sử dụng.

Mối quan hệ này bao gồm một số điều kiện và tham chiếu đến các điểm mở rộng trong trường hợp sử dụng cơ sở. Để việc
mở rộng diễn ra, phải đáp ứng một điều kiện nhất định.

mối quan hệ.

Tham chiếu điểm mở rộng xác định những vị trí trong trường hợp sử dụng cơ sở nơi phần mở rộng tương ứng sẽ được
đặt khi điều kiện được đáp ứng.

Một trường hợp sử dụng có thể là phần mở rộng cho một số tùy chọn cơ bản và cũng có một số tùy chọn khác làm phần
mở rộng của riêng nó. Ngoài ra, trường hợp sử dụng cơ bản có thể không phụ thuộc vào các tiện ích mở rộng của nó theo bất kỳ
cách nào.

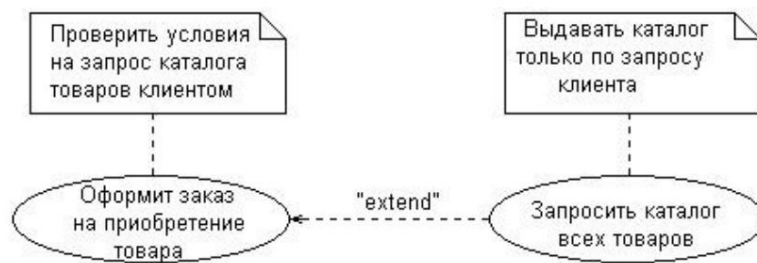
Ngữ nghĩa của quan hệ mở rộng được định nghĩa như sau. Nếu một phiên bản của một ca sử dụng thực hiện một số
chuỗi hành động xác định hành vi của nó và có một điểm mở rộng cho một phiên bản của một ca sử dụng khác là điểm
đầu tiên trong số các điểm mở rộng của ca sử dụng ban đầu, thì điều kiện của trường hợp này mối quan hệ được kiểm tra. Nếu
điều kiện được đáp ứng, chuỗi hành động ban đầu sẽ được mở rộng để bao gồm các hành động của một phiên bản của một biến thể
khác

sử dụng.

Lưu ý rằng điều kiện quan hệ mở rộng chỉ được kiểm tra một lần, lần đầu tiên điểm mở rộng được tham chiếu và nếu
nó được thỏa mãn thì tất cả các trường hợp sử dụng mở rộng sẽ được chèn vào trường hợp cơ sở.

Trong ví dụ được trình bày ở trên (Hình 23), khi đặt hàng mua hàng, chỉ trong một số trường hợp, có thể cần phải
cung cấp cho khách hàng danh mục tất cả hàng hóa. Trong trường hợp này, điều kiện mở rộng là yêu cầu từ khách hàng để nhận
danh mục sản phẩm. Rõ ràng, sau khi nhận được danh mục, khách hàng cần một
thời gian để nghiên cứu nó, trong thời gian đó đơn hàng sẽ bị tạm dừng.

Sau khi xem xét danh mục, khách hàng quyết định chọn một sản phẩm cụ thể hoặc không mua gì cả. Dịch vụ hoặc tùy chọn
sử dụng “Đặt hàng để mua hàng” có thể đáp ứng lựa chọn của khách hàng sau khi khách hàng nhận được danh mục sản phẩm để xem
xét.



См. 24. Biểu diễn đồ họa của mối quan hệ mở rộng với
lưu ý về điều kiện thực hiện ca sử dụng

Quan hệ tổng quát hóa Quan hệ

tổng quát hóa được sử dụng để chỉ ra thực tế là một số ca sử dụng A có thể được khái quát hóa thành trường hợp sử dụng B. Trong trường hợp này, tùy chọn A sẽ là một chuyên biệt hóa của tùy chọn B.

Trong trường hợp này, B được gọi là tổ tiên hoặc cha mẹ của A, và lựa chọn A là con trong trường hợp sử dụng B. Cần nhấn mạnh rằng con kế thừa tất cả các đặc tính và hành vi của cha mẹ nó và cũng có thể được bổ sung các đặc tính và hành vi mới. Về mặt đồ họa, mối quan hệ này được

biểu thị bằng một đường liền nét có mũi tên có hình tam giác mở, biểu thị trường hợp sử dụng chính (Hình 25). Dòng có mũi tên này có một tên đặc biệt - mũi tên "khái quát hóa".



См. 25. Một ví dụ về biểu diễn đồ họa của mối quan hệ khái quát hóa
giữa các trường hợp sử dụng

Mối quan hệ khái quát hóa giữa các ca sử dụng được sử dụng khi cần lưu ý rằng các ca sử dụng con có tất cả các thuộc tính và hành vi của ca sử dụng cha. Trong trường hợp này, các ca sử dụng con tham gia vào tất cả các mối quan hệ của các ca sử dụng cha. Đổi lại, các tùy chọn con có thể được cung cấp các thuộc tính hành vi mới không có trong

các trường hợp sử dụng gốc, cũng như tinh chỉnh hoặc sửa đổi các thuộc tính hành vi được kế thừa từ chúng.

Đối với một mối quan hệ nhất định, một ca sử dụng có thể có nhiều trường hợp con. Trong trường hợp này, sự kế thừa đa dạng các thuộc tính và hành vi của mối quan hệ tổ tiên được thực hiện.

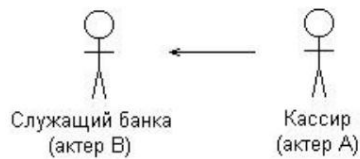
Mặt khác, một ca sử dụng có thể là tổ tiên của một số trường hợp con, tương ứng với bản chất phân loại của mối quan hệ khái quát hóa.

Một mối quan hệ khái quát hóa cũng có thể tồn tại giữa các tác nhân riêng lẻ. Mối quan hệ này mang tính định hướng và chỉ ra thực tế về sự chuyên môn hóa của một số tác nhân so với những tác nhân khác.

Ví dụ, mối quan hệ tổng quát từ tác nhân A đến tác nhân B đánh dấu một thực tế là mọi thể hiện của tác nhân A vừa là một thể hiện của tác nhân B vừa có tất cả các thuộc tính của nó.

Trong trường hợp này, diễn viên B là cha mẹ của diễn viên A và diễn viên A là con của diễn viên B. Diễn viên A có khả năng đóng nhiều vai giống như diễn viên B.

Về mặt đồ họa, mối quan hệ này cũng được biểu thị bằng một mũi tên khái quát, tức là một đường liền nét với một mũi tên có hình tam giác mở, chỉ vào tác nhân chính (Hình 26).



См. 26. Một ví dụ về biểu diễn đồ họa của mối quan hệ khái quát hóa giữa các diễn viên

Mối quan hệ bao gồm

Mối quan hệ bao hàm giữa hai trường hợp sử dụng chỉ ra rằng một số hành vi được chỉ định cho một trường hợp sử dụng

việc sử dụng được đưa vào như một thành phần không thể thiếu trong

trình tự hành vi của một ca sử dụng khác.

Mối quan hệ này là mối quan hệ nhị phân có hướng theo nghĩa là một cặp trường hợp sử dụng luôn được sắp xếp theo thứ tự bao gồm.

Ngữ nghĩa của mối quan hệ này được định nghĩa như sau. Khi phiên bản ca sử dụng đầu tiên đạt đến điểm được đưa vào chuỗi hành vi của phiên bản ca sử dụng thứ hai trong quá trình thực thi, phiên bản ca sử dụng đầu tiên sẽ thực hiện một chuỗi hành động xác định hành vi của phiên bản ca sử dụng thứ hai, sau đó tiếp tục thực hiện các hành động thuộc hành vi của nó.

Điều này giả định rằng mặc dù một phiên bản của trường hợp sử dụng đầu tiên có thể có nhiều phiên bản của các trường hợp khác được bao gồm trong đó, nhưng các hành động mà chúng thực hiện phải kết thúc tại một thời điểm nào đó, sau đó các hành động bị gián đoạn của phiên bản trường hợp sử dụng đầu tiên phải tiếp tục thực thi trong

phù hợp với hành vi được chỉ định cho nó.

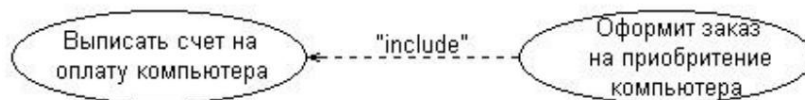
Một trường hợp sử dụng có thể được bao gồm trong một số trường hợp khác các tùy chọn và cũng bao gồm các tùy chọn khác.

Một trường hợp sử dụng đi kèm có thể độc lập với trường hợp cơ sở theo nghĩa là nó cung cấp cho trường hợp cơ sở một số hành vi được đóng gói, các chi tiết triển khai của chúng được ẩn khỏi trường hợp cơ sở.

cái sau và có thể dễ dàng phân phối lại giữa nhiều trường hợp sử dụng được bao gồm. Hơn nữa, một trường hợp cơ sở có thể chỉ phụ thuộc vào kết quả thực hiện hành vi mà nó bao gồm chứ không phụ thuộc vào cấu trúc của các biến thể mà nó bao gồm.

Mối quan hệ bao hàm hướng từ ca sử dụng A đến ca sử dụng B chỉ ra rằng mỗi phiên bản của ca sử dụng A bao gồm các thuộc tính chức năng được chỉ định cho ca sử dụng B. Các thuộc tính này chuyên biệt hóa hành vi của ca sử dụng A tương ứng trong một sơ đồ nhất định. Về mặt đồ họa, mối quan hệ này được biểu thị bằng đường chấm chấm có mũi tên (biến thể của mối quan hệ phụ thuộc) hướng từ trường hợp sử dụng cơ sở đến trường hợp sử dụng được bao gồm.

Trong trường hợp này, dòng có mũi tên này được đánh dấu bằng từ khóa "include", như trong Hình. 27.



Cơ. 27. Một ví dụ về biểu diễn đồ họa của mối quan hệ bao hàm giữa các trường hợp sử dụng

Ví dụ về xây dựng sơ đồ các ca sử dụng Ví dụ: chúng ta hãy xem xét quy trình mô hình hóa một hệ thống bán hàng thông qua danh mục, danh mục này có thể được sử dụng để tạo ra các hệ thống thông tin tương ứng. Hai chủ thể có thể đóng vai trò là tác nhân trong hệ thống này, một chủ thể là người bán và chủ thể kia là người mua. Mỗi tác nhân này tương tác với hệ thống bán hàng theo danh mục được đề cập và là người dùng của nó, tức là cả hai đều truy cập vào dịch vụ "Đặt hàng để mua hàng" tương ứng.

Các giá trị của bội số được chỉ ra trong sơ đồ này phản ánh các quy tắc hoặc logic chung để đặt hàng mua hàng. Theo các quy tắc này, một người bán có thể tham gia đặt một số đơn đặt hàng, đồng thời, mỗi đơn hàng chỉ có thể được đặt bởi một người bán, người này chịu trách nhiệm về tính đúng đắn của việc thực hiện và về vấn đề này, sẽ có thù lao của đại lý cho việc thực hiện nó. Mặt khác, mỗi người mua có thể đặt một số đơn hàng cho mình, nhưng đồng thời, mỗi đơn hàng phải đứng tên một người mua duy nhất, người sẽ chuyển quyền sở hữu hàng hóa sau khi thanh toán. Trường hợp sử dụng "Đặt hàng để mua sản phẩm" có thể được cải tiến bằng cách xem xét bốn trường hợp sử dụng bổ sung. Điều này xuất phát từ một phân tích chi tiết hơn về quá trình bán hàng, cho phép chúng tôi phân biệt các dịch vụ riêng biệt

những hành động như cung cấp cho người mua thông tin về sản phẩm, thỏa thuận các điều khoản thanh toán cho sản phẩm và đặt hàng sản phẩm từ kho.

Khả rõ ràng rằng những hành động này bộc lộ hành vi của ca sử dụng ban đầu theo nghĩa khởi tạo của nó và do đó sẽ có mối quan hệ bao hàm giữa chúng. Mặt khác, việc bán hàng thông qua danh mục giả định trước sự hiện diện của một đối tượng thông tin độc lập - danh mục hàng hóa, theo một nghĩa nào đó, không phụ thuộc vào việc

thực hiện dịch vụ khách hàng. Trong trường hợp của chúng tôi, người mua hoặc người bán có thể yêu cầu một danh mục sản phẩm nếu cần thiết để chọn một sản phẩm và làm rõ các chi tiết về việc bán sản phẩm đó. Sẽ khá hợp lý khi trình bày dịch vụ “Yêu cầu danh mục sản phẩm” như một trường hợp sử dụng độc lập.

Một mặt, việc chi tiết hóa có thể được thực hiện trên cơ sở thiết lập các mối quan hệ bổ sung như mối quan hệ “tổng quát hóa-chuyên môn hóa” cho các thành phần hiện có của sơ đồ biến thể.

sử dụng.

Do đó, trong khuôn khổ hệ thống đang được xem xét để bán hàng hóa, một loại hàng hóa riêng biệt-máy tính-có thể có ý nghĩa độc lập và các tính năng cụ thể.

Trong trường hợp này, sơ đồ có thể được bổ sung với trường hợp sử dụng “Đặt hàng mua máy tính” và các tác nhân “Người mua máy tính” và “Người bán máy tính”, được kết nối với các thành phần tương ứng của sơ đồ bằng cách khái quát hóa. mối quan hệ (Hình 4.14). Sơ đồ ca sử dụng được tinh chỉnh theo cách này chứa một tính năng quan trọng cần được

ghi chú.

Cụ thể là, mặc dù sơ đồ này (Hình 28) không hiển thị các mối quan hệ liên kết giữa tác nhân “Người bán máy tính” và trường hợp sử dụng “Đặt hàng mua máy tính”, cũng như giữa “Người mua máy tính”. “ tác nhân và máy tính ca sử dụng “Đặt đơn hàng”, sự hiện diện của mối quan hệ khái quát hóa giữa các thành phần tương ứng cho phép chúng kế thừa mối quan hệ liên kết từ tổ tiên của chúng. Vì nguyên tắc kế thừa là một trong những nguyên tắc cơ bản của lập trình hướng đối tượng, nên trong ví dụ của chúng tôi, chúng tôi có thể tự tin nói rằng các đường liên kết này với bộ số tương ứng hiện diện trong sơ đồ này ở dạng ẩn.



См. 28. Một trong những lựa chọn để hoàn thiện hơn nữa sơ đồ trường hợp sử dụng cho một ví dụ về hệ thống bán hàng được đề cập

Lập sơ đồ ca sử dụng là bước đầu tiên trong quá trình phân tích và thiết kế hướng đối tượng, mục đích của nó là thể hiện tập hợp các yêu cầu đối với hành vi của hệ thống được thiết kế.

Đặc tả các yêu cầu đối với hệ thống được thiết kế dưới dạng sơ đồ ca sử dụng là một mô hình độc lập, trong ngôn ngữ UML được gọi là mô hình ca sử dụng và có tên tiêu chuẩn đặc biệt hoặc khuôn mẫu riêng là "useCaseModel".