





PHẦN II: QUY TRÌNH THIẾT KẾ HỆ TƯƠNG TÁC

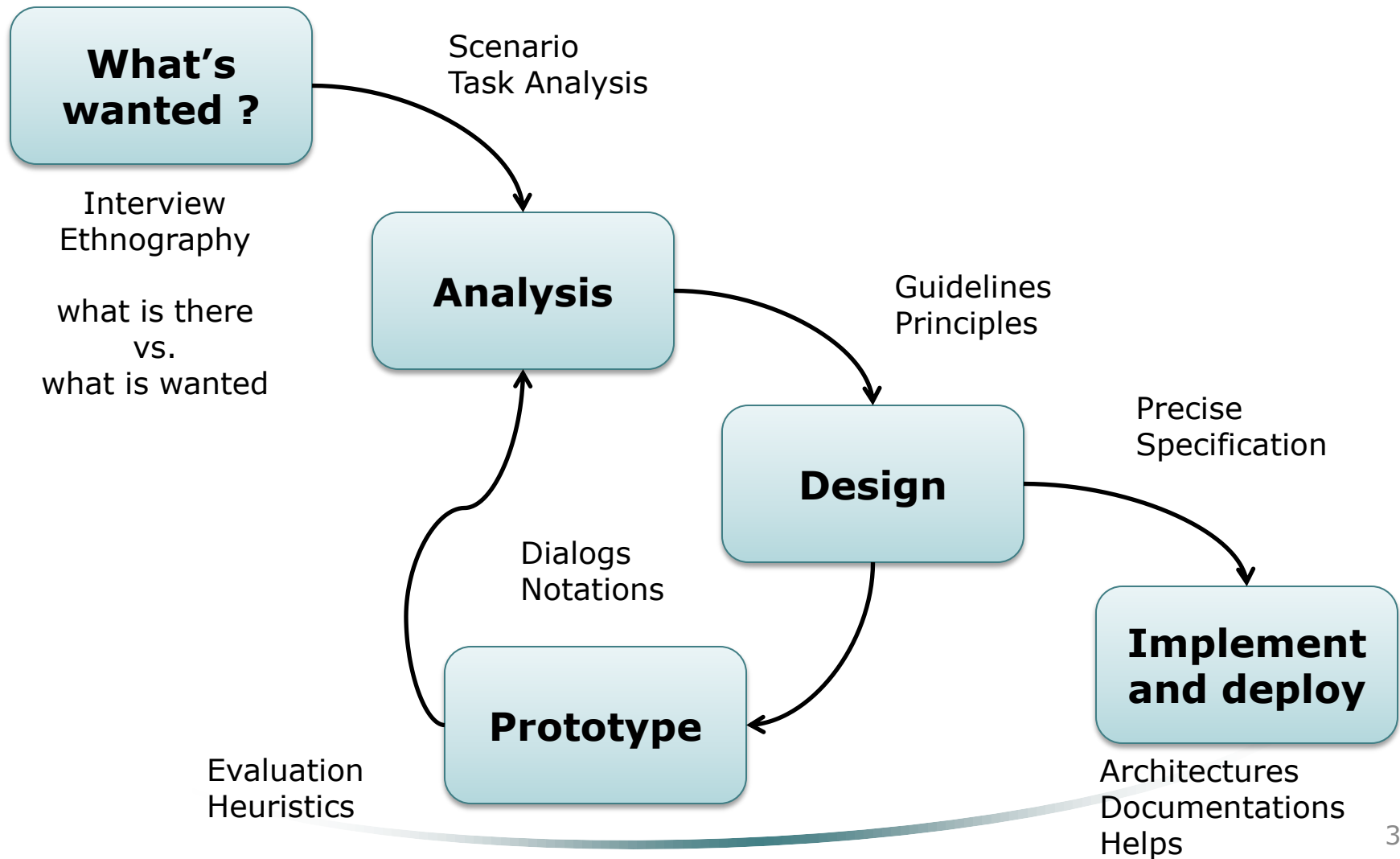
- 
- I. Giới thiệu chung
 - II. Đặc tả yêu cầu và phân tích nhiệm vụ
 - III. Thiết kế tương tác người dùng máy tính
 - IV. Kiểm thử tính dùng được và đánh giá hệ thống
 - V. Quản lý hệ thống tương tác



Nhắc lại một số khái niệm


- Thế nào là một thiết kế tốt và một thiết kế tồi ?
 - Thiết kế tương tác là gì ?
 - Quy trình thiết kế tương tác có đặc điểm gì khác biệt so với các quy trình thiết kế phần mềm nói chung ?
- 

Quy trình thiết kế hệ tương tác

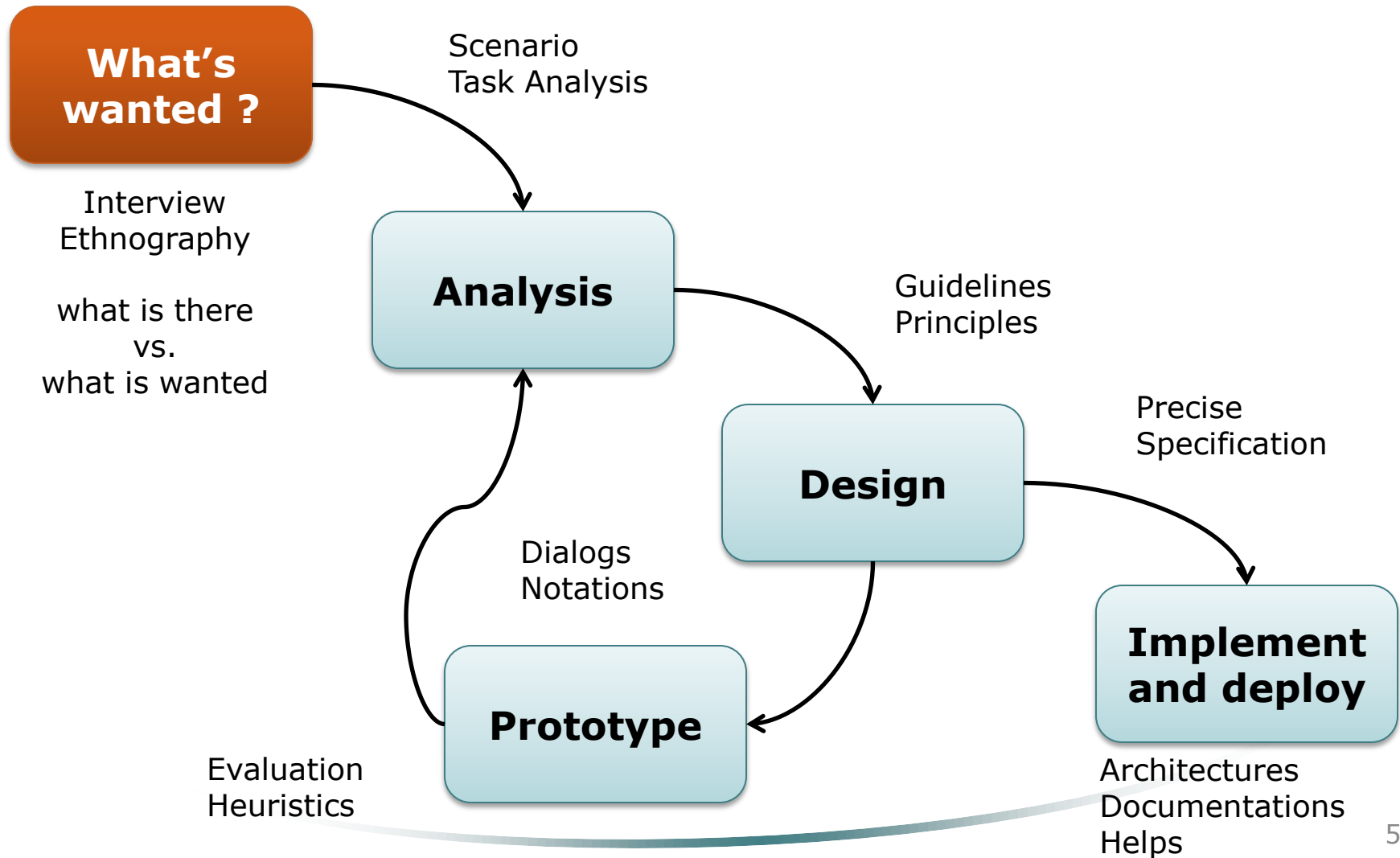




CHƯƠNG II: ĐẶC TẢ YÊU CẦU NGƯỜI DÙNG

- 
- I. Khái niệm**
 - II. Mô hình người dùng
 - III. Các kiểu đặc tả

Quy trình thiết kế hệ tương tác





Đặc tả yêu cầu người dùng

- Quá trình xác định các yêu cầu của khách hàng đối với hệ thống mà ta cần phát triển
 - Người dùng là ai
 - Mục đích của họ là gì
 - Nhiệm vụ nào họ muốn hoàn thành

- Đầu vào:
 - Khách hàng cung cấp một mô tả về yêu cầu của họ, cái mà họ mong muốn hệ thống cung cấp bằng thuật ngữ của họ
- Xử lý: bản mô tả do khách hàng cung cấp còn chung chung và cần phải xác định
 - Những cái không cần thiết
 - Những cái không thể thực hiện được hay không chắc chắn thực hiện được
 - Những cái còn thiếu, nhập nhằng hay mâu thuẫn
- Đầu ra:
 - Biểu diễn bài toán với hệ thống hiện tại
 - Biểu diễn các yêu cầu của hệ thống mới



Cách tiếp cận

- Mô hình hoá yêu cầu người dùng
 - Các kỹ thuật sử dụng
 - Phỏng vấn, bảng câu hỏi
 - Quan sát
 - Phân tích tài liệu
- Diễn giải lại bằng ngôn ngữ chuyên ngành IT
 - Các kiểu đặc tả:
 - Đặc tả chức năng
 - Đặc tả dữ liệu
 - Đặc tả tính dùng được (HCI)



CHƯƠNG II: ĐẶC TẢ YÊU CẦU NGƯỜI DÙNG

I. Khái niệm

II. Mô hình người dùng

1. Mô hình hóa yêu cầu người dùng
2. OSTA
3. USTA
4. Đa cách nhìn
5. Mô hình nhận thức

III. Các kiểu đặc tả



Mở đầu

- Nhận biết và hiểu người dùng hệ thống: cần gì, có thể làm gì
 - Người dùng tương tác với máy tính như thế nào
 - Các nhân tố con người ảnh hưởng đến thiết kế hệ thống
 - Mức độ hiểu biết và kinh nghiệm của người dùng
 - Các đặc trưng về nhu cầu, công việc và nhiệm vụ của người dùng
 - Đặc trưng tâm sinh lý của người dùng
 - Đặc trưng vật lý của người dùng
 - Cách thức người dùng trau dồi kiến thức

1. Mô hình hóa yêu cầu người dùng

- Thiết kế giao tiếp người dùng - máy tính thường được mô tả bằng tài liệu: văn bản, tranh, sơ đồ, nhằm giảm thiểu yêu cầu/ cơ hội cho cài đặt.
 - Mô hình hình thức
 - Mô hình phi hình thức:
 - triết lý thiết kế với các thành phần như đối tượng, hành động;
 - mô tả chi tiết về ngữ nghĩa các chức năng.
- Cung cấp đầu vào cho hệ thống quản lý các giao tiếp người dùng - UIMS, trao đổi với các nhóm khác.

1. Mô hình hóa yêu cầu người dùng

- Nhằm mô tả các khía cạnh khác nhau của người dùng: hiểu biết, chú ý và xử lý
- Các dạng chung: năng lực và hiệu suất
- Các mô hình:
 - **Phân tích hệ thống mở (Open System Task Analysis- OSTA)**
 - **Phân tích kỹ năng và nhiệm vụ người dùng (User Skills and Task Analysis)**
 - Mô hình hệ thống mềm (Soft System methodology)
 - **Mô hình đa cách nhìn (multiview)**
 - Mô hình dự đoán: **GOMS**, KEYSTROKE



2. Mô hình kỹ thuật xã hội OSTA

- Cách thức làm việc với người dùng trong quá trình thiết kế: thiết kế thành viên và thiết kế xã hội.
 - Thiết kế thành viên: người dùng tham gia vào các công đoạn phân tích yêu cầu, lập kế hoạch
 - Thiết kế xã hội: tập trung phát triển đầy đủ và nhất quán hệ thống
- Nhiệm vụ chính: xác định
 - Yêu cầu công việc: nhiệm vụ cho từng nhóm, đầu vào nhiệm vụ, môi trường bên ngoài
 - Hệ thống thực thi công việc: hệ thống xã hội, hệ thống kỹ thuật
 - Các đặc tính khác: mức độ thỏa mãn về hiệu năng, chức năng, tính dùng được, tính chấp nhận được



Các bước thực hiện theo OSTA

- Liệt kê các nhiệm vụ chính
- Xác định đầu vào của các nhiệm vụ (bên ngoài hệ thống)
- Thiết lập môi trường bên ngoài
- Mô tả quá trình biến đổi từ đầu vào thành đầu ra
- Phân tích hệ thống xã hội: vai trò, đặc tính, chất lượng
- Phân tích hệ thống kỹ thuật: cũ và mới, hiệu quả làm việc
- Đặc tả yêu cầu về mức độ hiệu năng thỏa mãn
- Đặc tả yêu cầu về chức năng, tính dùng được, tính chấp nhận được cho hệ thống kỹ thuật mới

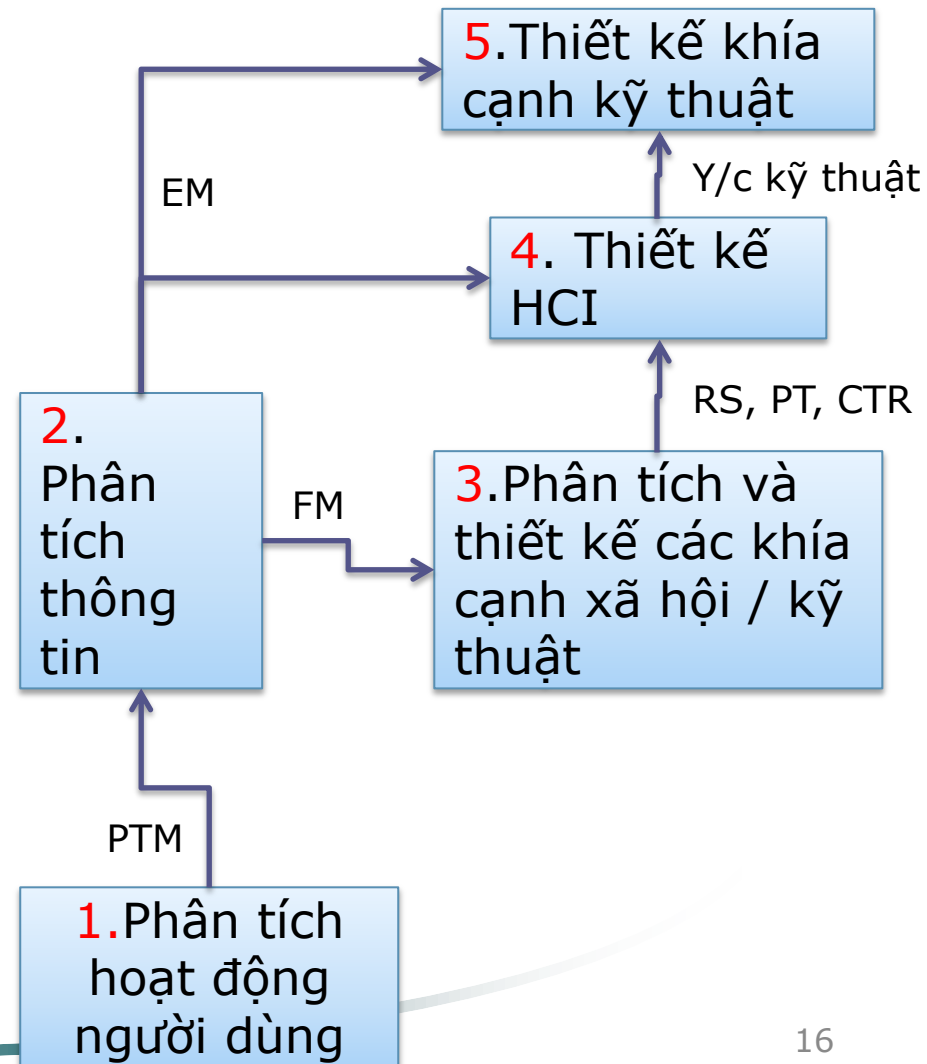


3. USTA

- Mô tả yêu cầu của mọi người có quyền lợi và nghĩa vụ liên quan đến hệ thống cần phát triển
 - Người dùng hệ thống.
 - Người không sử dụng trực tiếp hệ thống song có nhận thông tin từ đầu ra hệ thống
 - Không thuộc hai loại trên song có chịu tác động từ sự thành công hay thất bại của hệ thống.
 - Người tham gia vào quá trình thiết kế, phát triển và bảo trì hệ thống.
- Lập bảng câu hỏi sao cho câu trả lời của người dùng luôn nằm trong tập các câu trả lời được định nghĩa sẵn

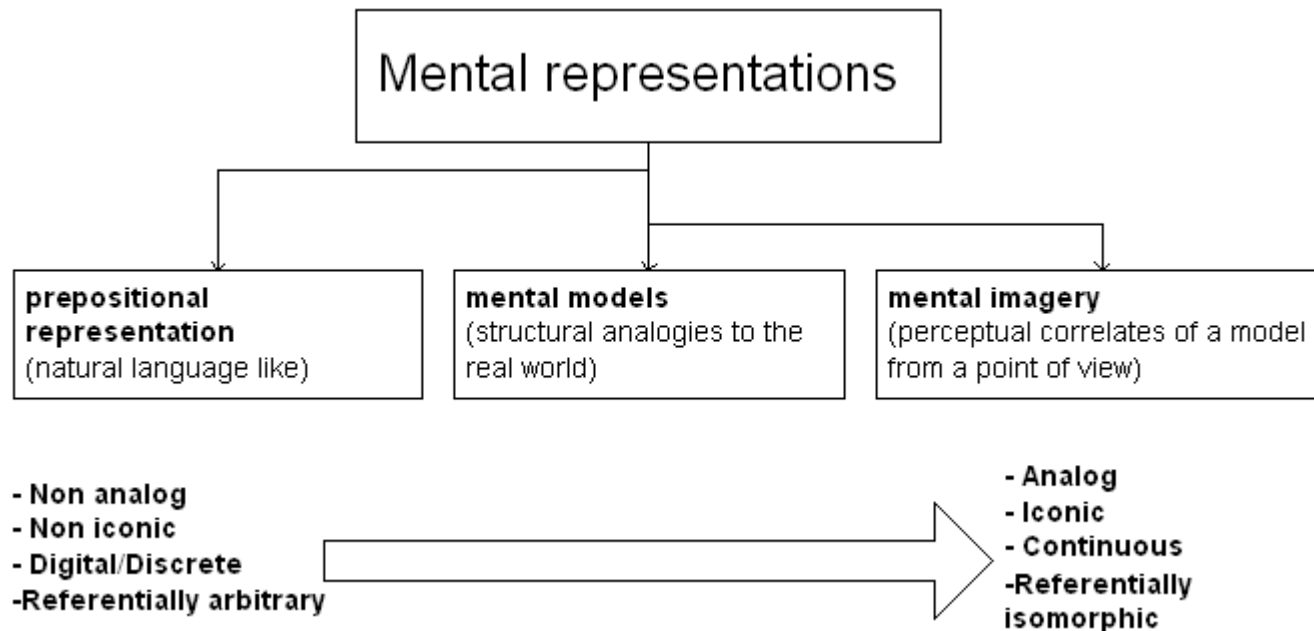
4. Mô hình đa cách nhìn

- Là một cách tiếp cận tổ hợp nhiều cách tiếp cận trong 1 giai đoạn, có phương pháp kiểm tra.
 - PTM: Mô hình các nhiệm vụ chính
 - FM: Mô hình chức năng
 - EM: Mô hình thực thể (mô hình khái niệm)
 - RS: các vai trò
 - PT: các nhiệm vụ của người dùng
 - CTR: Yêu cầu các nhiệm vụ của máy tính
- Tiếp cận đa cách nhìn nhấn mạnh vào thứ tự hoạt động => không thích hợp.

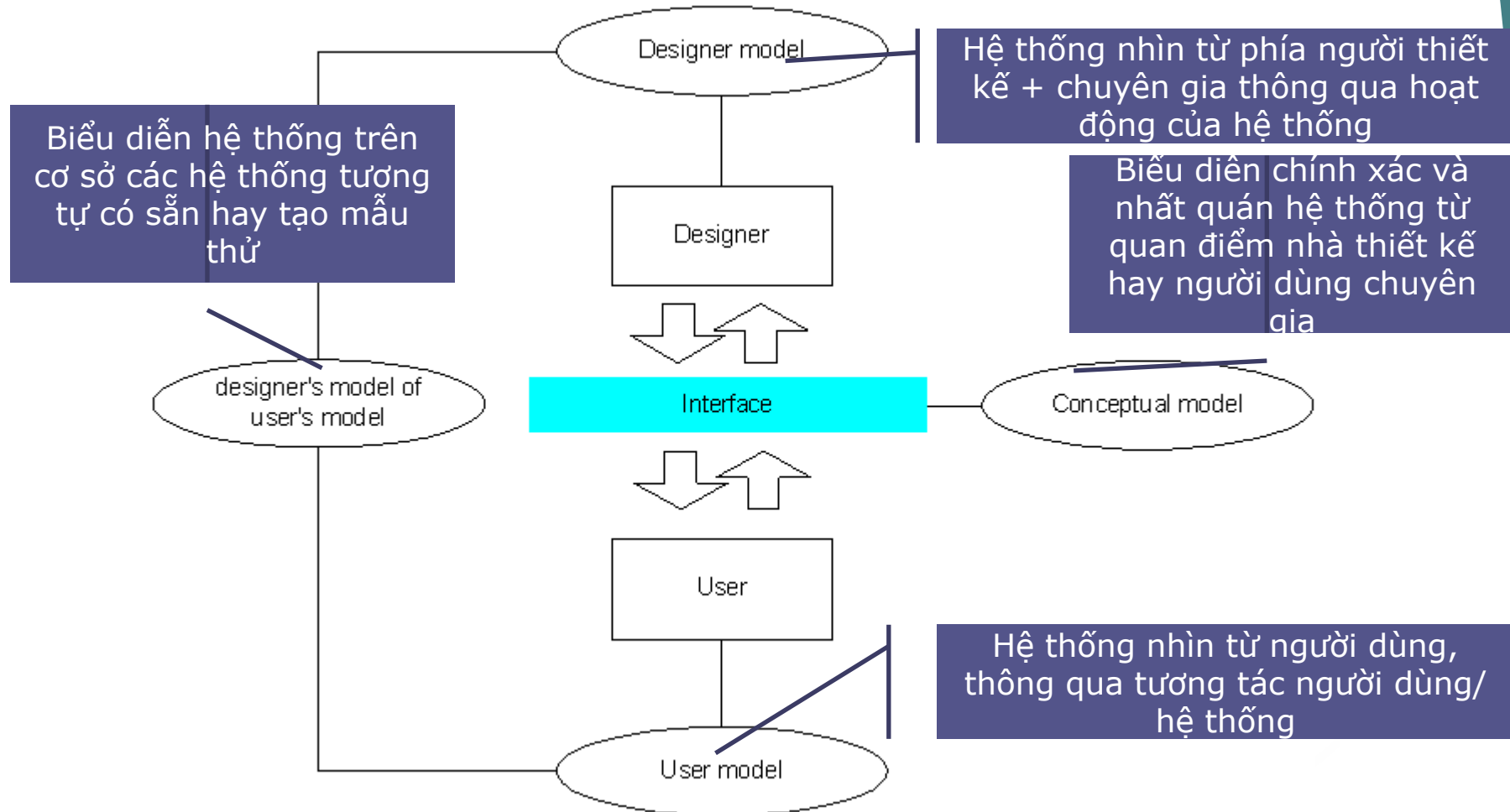


5. Mô hình nhận thức

- Biểu diễn các tình huống thực tế hoặc tưởng tượng
- Xây dựng từ nhận thức, trí tưởng tượng hay cách phiên dịch các hội thoại của người dùng
- Đại diện cho những gì được cho là đúng, không đại diện cho những gì được cho là sai



Mô hình nhận thức trong HCI






Bài tập lớn

- Hỏi ý kiến giáo viên để chốt đề cương bài tập lớn:
 - Mục tiêu
 - Công việc cần thực hiện
 - Kết quả cần đạt
- Chốt kế hoạch làm bài tập lớn và nộp báo cáo:
 - GĐ I: 4 tuần (hạn nộp báo cáo 26/10 23h59)
 - GĐ II : 2 tuần (hạn nộp báo cáo 9/11 23h59)
- Lưu ý: Báo cáo phải chỉ rõ công việc của từng người trong nhóm
- Bảo vệ vào 3 tuần cuối của học kỳ (17/11, 24/11 và 1/12)



CHƯƠNG II: ĐẶC TẢ YÊU CẦU NGƯỜI DÙNG

- 
- I. Khái niệm
 - II. Mô hình người dùng
 - III. Các kiểu đặc tả**

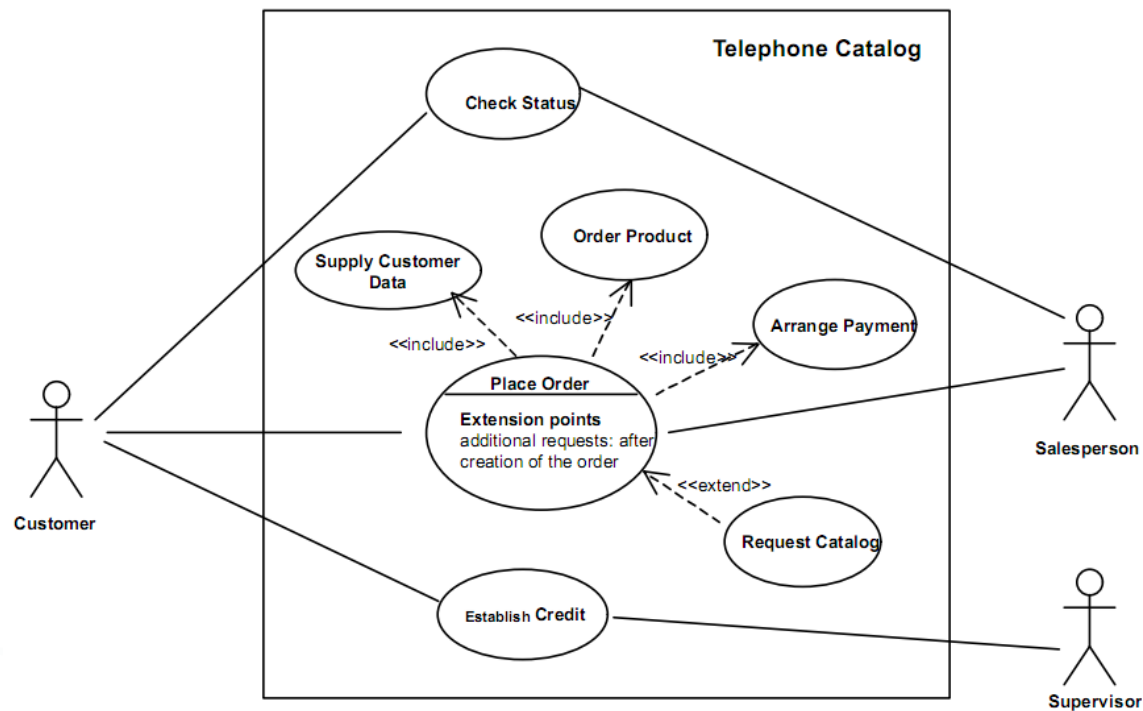


1. Đặc tả chức năng

- Là đặc tả cái mà hệ thống (con người, máy tính, ...) phải làm
 - Việc quyết định hành động nào sẽ được thực hiện bởi con người hay bởi máy tính sẽ được thực hiện ở pha phân tích nhiệm vụ
- Đặc tả chức năng bao gồm đặc tả các ràng buộc mà chức năng khi thực hiện phải tính đến
 - Việc đặc tả thường được chia thành nhiều module có phân cấp để dễ điều khiển và cho phép xử lý riêng biệt: đi từ mức trừu tượng đến mức cụ thể
- Quan trọng: thu thập các yêu cầu không thể thực hiện được và chỉ đặc tả các yêu cầu này một lần

Công cụ đặc tả chức năng

- Sơ đồ luồng dữ liệu (Data Flow Diagram - DFD)
- Máy với trạng thái hữu hạn
- Ngôn ngữ tự nhiên có cấu trúc



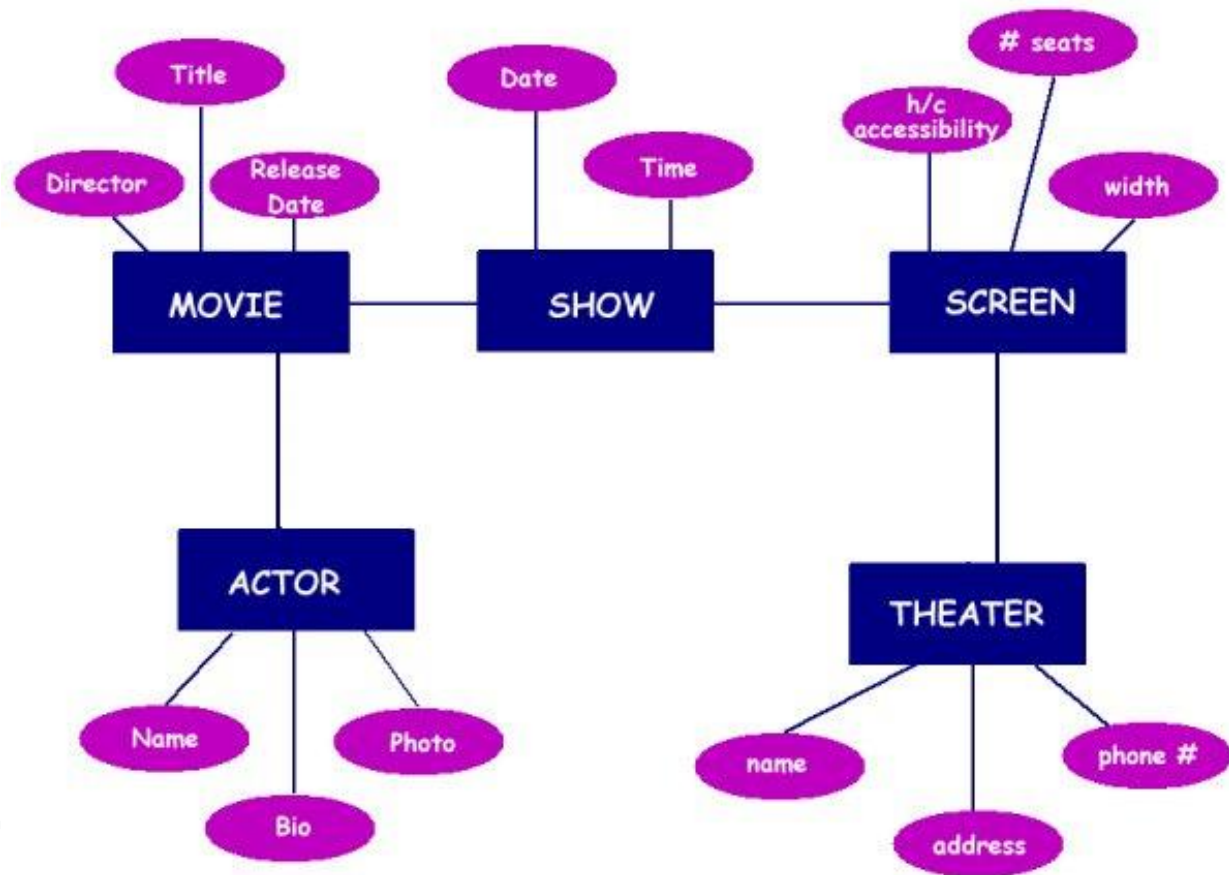


2. Đặc tả dữ liệu

- Là biểu diễn luồng dữ liệu, ngữ nghĩa, cấu trúc chính của dữ liệu đáp ứng yêu cầu của người dùng
- Đặc tả dữ liệu khác với đặc tả chức năng: nó chỉ tập trung vào cấu trúc dữ liệu
- Đặc tả dữ liệu được liệt kê nhờ các kỹ thuật
 - Quan sát
 - Phân tích tài liệu
 - Phỏng vấn
- Quan trọng: cần phải hiểu và định nghĩa một cách chính xác các phần tử dữ liệu

Công cụ đặc tả dữ liệu

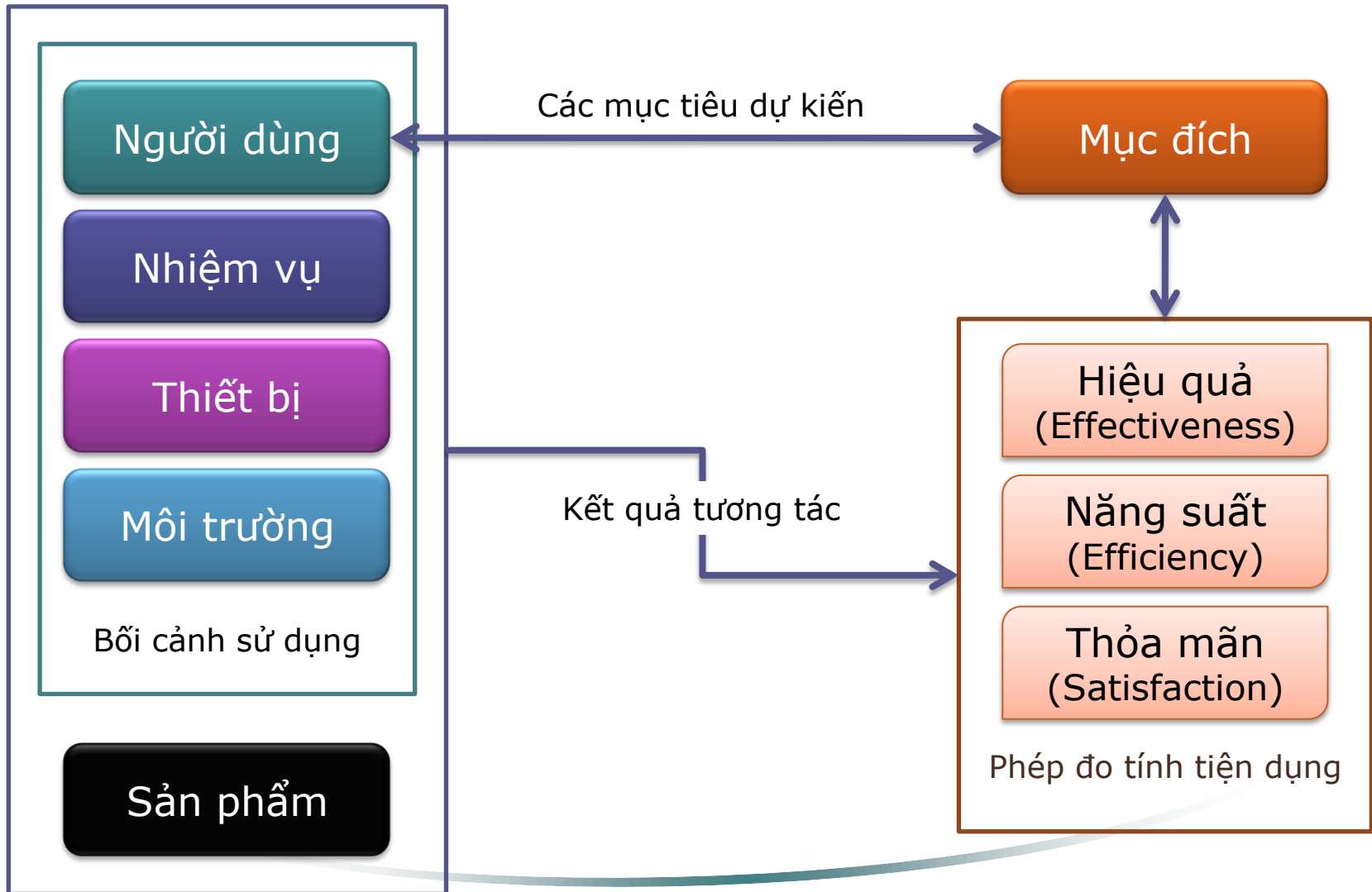
- Lưu đồ thực thể liên kết - Entity Relationship Diagram- ERD



3. Đặc tả tính dùng được

- Trong việc xây dựng một hệ thống thông tin ta mới chỉ quan tâm đến hai loại đặc tả:
 - Đặc tả chức năng
 - Đặc tả dữ liệu
- Trong giao tiếp người dùng - máy tính, cần đặc tả thêm tính dùng được:
 - Việc đặc tả tính dùng được được tiến hành đồng thời với đặc tả chức năng và đặc tả dữ liệu với cùng một kỹ thuật (quan sát, phỏng vấn, phân tích tài liệu).
 - Hoạt động đặc tả tính dùng được có liên quan đến đánh giá quy trình thiết kế

Khung tính dùng được ISO 9241-11





CHƯƠNG III: PHÂN TÍCH NHIỆM VỤ



I. Khái niệm

- II. GOMS: mô hình phân cấp mục đích và nhiệm vụ
- III. Mô hình ngôn ngữ
- IV. Phân tích công việc



1. Phân tích nhiệm vụ

- Quá trình phân tích cách thức người dùng thực hiện công việc để đạt được mục đích của mình.
- Phân tích nhiệm vụ tập trung vào:
 - Các hành động của người dùng (actions)
 - Đối tượng mà người dùng tác động vào (objects)
 - Những tri thức mà người dùng cần có để thực thi nhiệm vụ nhằm đạt được mục đích mong muốn(knowledge)

2. Ví dụ về phân tích nhiệm vụ “Hút bụi”

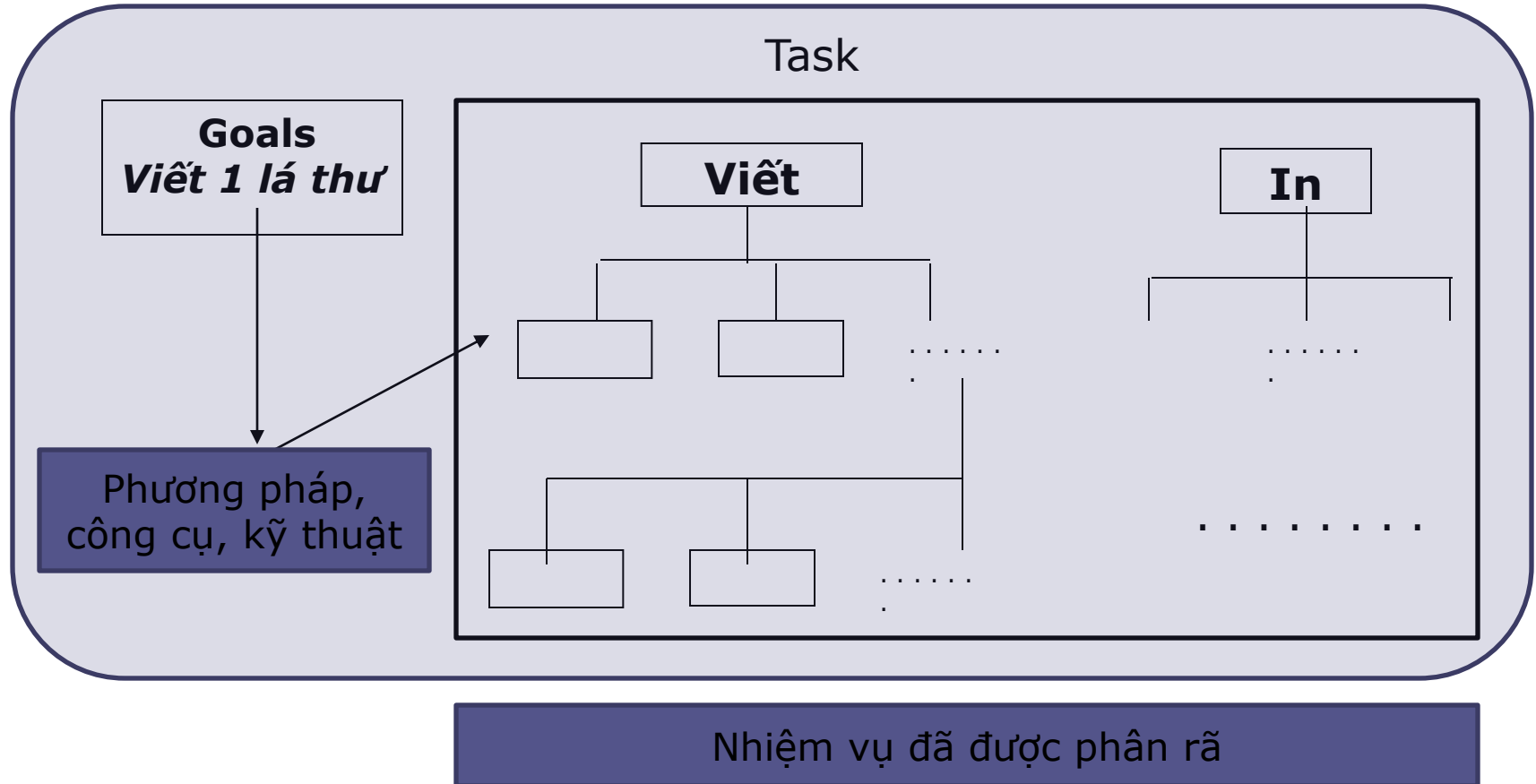
- Mục đích: “Hút bụi trong nhà”
- Các công việc cần làm:
 - Lấy máy hút bụi
 - Lắp các phụ tùng cần thiết
 - Thực hiện hút bụi
- Một số các điều kiện
 - Khi hộp rác đã đầy: tháo bỏ rác và lắp lại
 - Khi hút xong: tháo các phụ tùng và cất máy
- Tri thức cần có
 - Sử dụng máy hút bụi như thế nào
 - Việc tháo lắp các chi tiết ra sao
 - Trình tự hút ở các phòng như thế nào



3. Thuật ngữ

- Mục đích (Goal)
 - Trạng thái của hệ thống mà người dùng muốn hoàn thành
 - Một đích có thể được thực hiện bởi một số công cụ, phương pháp, tác nhân, kỹ thuật, thiết bị có thể làm thay đổi trạng thái của hệ thống
 - Ví dụ: mục đích là viết thư thì có thể dùng các phương tiện như bút, giấy, máy soạn thảo văn bản, v.v
- Nhiệm vụ (Task)
 - Là cái người dùng cần làm để thực hiện mục đích đề ra
- Hành động (Action)
 - Là một nhiệm vụ mà bản thân nó không bao hàm việc giải quyết vấn đề hay là một thành phần của cấu trúc điều khiển

Ví dụ: “Viết thư”



Kết quả cần đạt: Hiểu được các chức năng nghiệp vụ của hệ tương tác

- Định nghĩa quy trình nghiệp vụ và phân tích yêu cầu người dùng
- Xác định các chức năng nghiệp vụ cơ bản của hệ thống
- Mô tả lại các hành động của người dùng thông qua bước phân tích nhiệm vụ
- Hình thành mô hình hệ thống ở mức khái niệm
- Hình thành các chuẩn thiết kế hoặc chỉ dẫn phong cách thiết kế (nếu chưa có)
- Thiết lập các mục đích về tính dùng được
- Xác định các tài liệu cần cung cấp cho người dùng cũng như phương thức hướng dẫn người dùng sử dụng hệ thống

Giới thiệu mô hình GOMS

I. Khái niệm

II. GOMS: mô hình phân cấp mục tiêu và nhiệm vụ

- Mục tiêu của mô hình:
 - Mô tả phản ứng của con người ở nhiều cấp độ trừu tượng, từ nhiệm vụ tới các hành động vật lý
 - Tạo ra tính tương thích với chủ thể con người
- Đánh giá theo 2 hướng: phân tích nhiệm vụ và hình dung phản ứng của người dùng khi hoàn thành nhiệm vụ.
 - Các kỹ thuật đánh giá tương tự:
 - CCT (lý thuyết độ phức tạp nhận thức)
 - Phân tích nhiệm vụ phân cấp (Hierachial Task Analysis – HTA)



Goal-Operator-Methods-Selection

- Goal: mục đích mà người dùng muốn thực hiện.
 - Trạng thái mong muốn, bao gồm nhiều đích con (mục tiêu cơ sở).
 - Các mục đích được phân cấp tạo nên một cây mà các lá là các thao tác nhằm đạt được mục tiêu cơ sở
 - Operator: các thao tác cơ bản của ND như: nhấn phím, rê chuột, suy nghĩ, v.v. nhằm thay đổi trạng thái (trạng thái tâm lý của ND hay trạng thái môi trường).
 - Đặc trưng của mỗi thao tác: bắt đầu, kết thúc, cách giải quyết và nhiệm vụ cơ sở
 - Một thao tác được đánh giá qua các toán hạng vào, ra và thời gian cần thiết để thực hiện.
 - Thao tác có thể là cơ chế tâm lý hay đặc thù của môi trường.
- Dễ dàng xung đột vì có nhiều cách để đạt mục đích



Goal-Operator-Methods-Selection

- Method: mô tả cách thức để đạt mục đích.
 - phân rã mục đích thành các mục đích con/thao tác con, lưu trong bộ nhớ ngắn hạn dưới dạng chuỗi có điều kiện.
 - Nó không phải là kế hoạch hành động để hoàn thành nhiệm vụ mà là kết quả của kinh nghiệm được tích lũy.
- Selection: quy tắc lựa chọn các phương thức
 - “Nếu điều kiện C thì chọn cách thức M”



Ví dụ: dịch chuyển con trỏ trong một hệ soạn thảo văn bản

- Người dùng có thể dùng chuột hay bàn phím. Giả sử có 2 cách thức M1 và M2. M2 dùng khi khoảng cách lớn và thường dùng chuột, ngược lại khi khoảng cách nhỏ dùng M1 với bàn phím.
 - M1: Di chuột đến vị trí đích rồi chọn
 - M2: chừng nào con trỏ chưa đúng hàng nhấn \uparrow , chừng nào con trỏ chưa đúng vị trí nhấn \leftarrow (\rightarrow)
- Hai nguyên tắc chọn R1 và R2:
 - R1: Nếu vị trí cần đặt ở xa thì dùng M1
 - R2: Nếu vị trí cần đặt ở gần thì dùng M2

Giới thiệu

I. Khái niệm

II. GOMS: mô hình phân cấp mục tiêu và nhiệm vụ

III. Mô hình ngôn ngữ

- Là các mô hình hình thức được phát triển dựa trên các thuật ngữ của một ngôn ngữ.
- Mục đích: hiểu được hành vi của người dùng và phân tích các khó khăn về nhận thức của tương tác.
- Các mô hình chính:
 - Ký pháp BNF
 - Văn phạm nhiệm vụ hành động (TAG)

1. Ký pháp BNF

- BNF = Backus Naus Form: luật để mô tả văn phạm đối thoại
tên ::= <biểu thức>;
dấu ::= hiểu là “được định nghĩa”
- Chỉ liên quan đến cú pháp, bỏ qua ngữ nghĩa của ngôn ngữ.
 - Ký hiệu kết thúc viết bằng chữ in hoa
 - Ký hiệu không kết thúc viết bằng chữ thường→
- BNF được sử dụng khá rộng rãi để đặc tả cú pháp của các ngôn ngữ lập trình
- Chỉ biểu diễn hành động ND mà không đề cập đến cảm nhận của ND về sự đáp ứng của hệ thống

Ví dụ: chức năng vẽ đường của một ứng dụng đồ họa

- Có thể vẽ nhiều đoạn thẳng (polyline) nối giữa 2 điểm: chọn một điểm bằng cách nhấn chuột trong vùng vẽ và chỉ ra điểm cuối cùng bằng cách nhấn kép.
- Cú pháp:

```
vẽ đường ::= <chọn đường><chọn điểm><chọn điểm cuối>  
chọn đường ::= <định vị con trỏ><Nhấn phím đơn>  
chọn điểm ::= <chọn 1điểm>/<chọn điểm><chọn 1điểm>  
chọn 1điểm ::= <định vị con trỏ><Nhấn phím đơn>  
chọn điểm cuối ::= <định vị con trỏ ><Nhấn phím kép>  
định vị con trỏ ::= <rỗng ><di chuyển><định vị con trỏ>
```

2. Văn phạm nhiệm vụ hành động (Task Action Grammar- TAG)

- Đưa vào một số phần tử như luật văn phạm tham số để nhấn mạnh tính nhất quán và mã hoá tri thức của người dùng.

Ví dụ: Phân tích cú pháp một số lệnh của UNIX:
“cp”: sao chép, “mv”: chuyển và “ln”: liên kết

- BNF

```
copy ::= "cp" + filename + filename/  
      "cp" + filename + directory  
  
move  ::= "mv" + filename + filename/  
      "mv" + filename + directory  
  
link  ::= "ln" + filename + filename/  
      "ln" + filename + directory
```

- TAG

```
Fileop [op] := command[op] + filename + filename/  
            command[op] + filename + directory  
command[op=copy] := 'cp'  
command[op=move]  := 'mv'  
command[op=link]  := 'ln'
```

- không thể phân biệt tính nhất quán của lệnh và một biến thể không nhất quán nếu “ln” nhận thư mục là đối số thứ nhất.



CHƯƠNG III: PHÂN TÍCH NHIỆM VỤ

- I. Khái niệm
- II. GOMS: mô hình phân cấp mục đích và nhiệm vụ
- III. Mô hình ngôn ngữ
- IV. Phân tích công việc**
 - 1. Phân chia nhiệm vụ
 - 2. Sơ đồ quan hệ phân cấp các nhiệm vụ



1. Phân chia nhiệm vụ

- Chia nhiệm vụ thành các nhiệm vụ con
- Kỹ thuật dựa vào tri thức: người dùng hiểu gì về nhiệm vụ và nó được tổ chức ra sao?
- Phân tích dựa vào mô hình quan hệ thực thể: mối quan hệ giữa các thực thể, hành động và người dùng trong quá trình thực hiện.



a. Kỹ thuật phân chia nhiệm vụ (Task decomposition)

- Mục đích: mô tả cái mà người dùng phải thực hiện thành các nhiệm vụ con và thứ tự của các nhiệm vụ con
- Biểu diễn dưới dạng sơ đồ hay văn bản các mức thao tác và các kế hoạch.
- Các mức thao tác không theo thứ tự; kế hoạch chỉ ra thứ tự.



b. Phân tích nhiệm vụ theo nhận thức

- Là kỹ thuật phân tích theo biểu diễn tri thức mà người dùng có hoặc cần phải có để hoàn thành mục đích
- Cơ sở: dựa vào lý thuyết nhận thức các hành động có tính vật lý, ví dụ như nhấn phím, di chuyển chuột, trao đổi có tính suy nghĩ hay hành động nhận thức
- Mô hình: mô hình xử lý thông tin con người, lý thuyết phức tạp nhận thức.

c. Phân tích nhiệm vụ theo mô hình tri thức (How to do it)

- Cơ sở: dựa vào ánh xạ nhiệm vụ - hành động. Nhiệm vụ được dùng, hành động thực hiện => GOMS là mô hình thích hợp nhất
- Việc thực hiện GOMS được chia thành nhiều mức trừu tượng khác nhau
- 3 tính chất của GOMS:
 - GOMS mô tả một tập các phương pháp để thực hiện nhiệm vụ
 - Mức độ nhiệm vụ cơ sở (unit task)
 - Mức độ hành động (Keystroke)



Ví dụ: Quản lý tệp của PC-MSDOS và MACINTOSH

- Users goal
 1. Xóa 1 tệp
 2. Xóa 1 thư mục
 3. Chuyển 1 tệp
 4. Chuyển 1 thư mục



PC-MSDOS

File manipulation methods - 1 (tiếp)

- Phương pháp thực hiện mục đích xóa 1 tệp
 - Step 1
 - Step 2
 - Step 3
 - Step 4
- Phương pháp thực hiện mục đích chuyển 1 tệp
 - Step 1
 - Step 2
 - Step 3
- Phương pháp thực hiện mục đích sao 1 tệp
 - Step 1
 - Step 2
 - Step 3
 - Step 4
 - Step 5



PC-MSDOS

File manipulation methods - 2

- Phương pháp thực hiện mục đích xóa 1 thư mục
 - Step 1 Xóa tất cả các tệp trong TM
 - Step 2 Xóa TM
 - Step 3 Kết thúc
- Phương pháp thực hiện mục đích xóa mọi tệp
 - Step 1 Chọn lệnh ERASE từ tệp lệnh
 - Step 2 Nghĩ tên TM
 - Step 3 Nhập tên TM: *.*
 - Step 4 Nhập lệnh và thực hiện lệnh
 - Step 5 Kết thúc
- Phương pháp thực hiện mục đích chuyển thư mục
 - Step 1 Chọn lệnh RD từ tệp lệnh
 - Step 2 Nghĩ tên TM và nhập
 - Step 3 Nhập lệnh và thực hiện lệnh
 - Step 4 Kết thúc



PC-MSDOS

File manipulation methods - 3

- Phương pháp thực hiện mục đích chuyển 1 thư mục
 - Step 1 Thực hiện sao TM
 - Step 2 Thực hiện xóa TM
 - Step 3 Kết thúc
- Phương pháp thực hiện mục đích sao 1 thư mục
 - Step 1 Tạo TM mới
 - Step 2 Sao mọi tệp sang TM mới
 - Step 3 Kết thúc
- Phương pháp thực hiện mục đích tạo 1 thư mục
 - Step 1 Chọn lệnh MD trong tập lệnh
 - Step 2 Nghĩ và nhập tên TM
 - Step 3 Thực hiện lệnh
 - Step 4 Kết thúc

PC-MSDOS

File manipulation methods - 3 (tiếp)

- Phương pháp thực hiện mục đích sao mọi tệp
 - Step 1 Chọn lệnh COPY từ tệp lệnh
 - Step 2 Nghĩ tên TM nguồn
 - Step 3 Nhập *.* trong vùng nguồn
 - Step 4 Nghĩ tên TM đích
 - Step 5 Nhập *.* trong vùng đích
 - Step 6 Thực hiện lệnh
 - Step 7 Kết thúc



Macintosh

Specific file manipulation methods

- Phương pháp thực hiện mục đích xoá 1 tệp
 - Step 1 Gấp tệp bỏ vào thùng rác
 - Step 2 Kết thúc
- Phương pháp thực hiện mục đích chuyển 1 tệp
 - Step 1 Gấp tệp bỏ vào nơi đến
 - Step 2 Kết thúc
- Phương pháp thực hiện mục đích xoá 1 TM
 - Step 1 Gấp TM bỏ vào thùng rác
 - Step 2 Kết thúc
- Phương pháp thực hiện mục đích chuyển 1 TM
 - Step 1 Gấp TM bỏ vào nơi đến
 - Step 2 Kết thúc



Macintosh

Generalize file manipulation methods

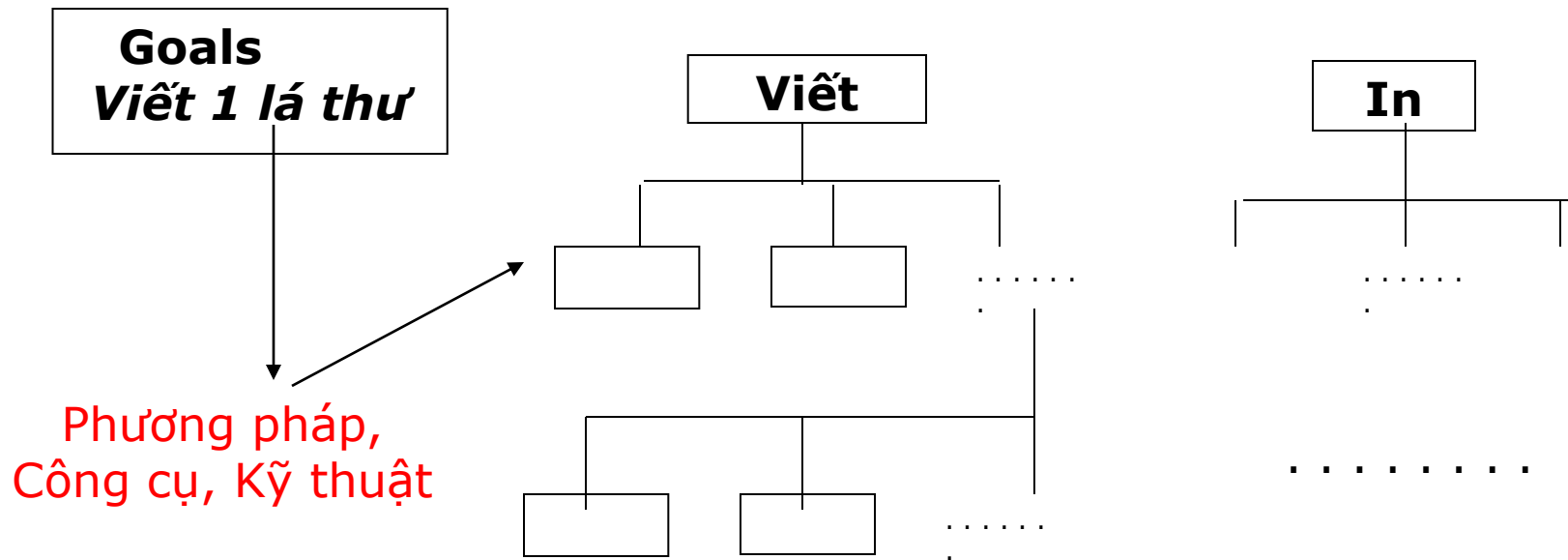
- Phương pháp thực hiện xóa 1 đối tượng
 - Step 1 Gấp đối tượng bỏ vào thùng rác
 - Step 2 Kết thúc
- Phương pháp thực hiện chuyển 1 đối tượng
 - Step 1 Gấp đối tượng bỏ vào nơi đến
 - Step 2 Kết thúc



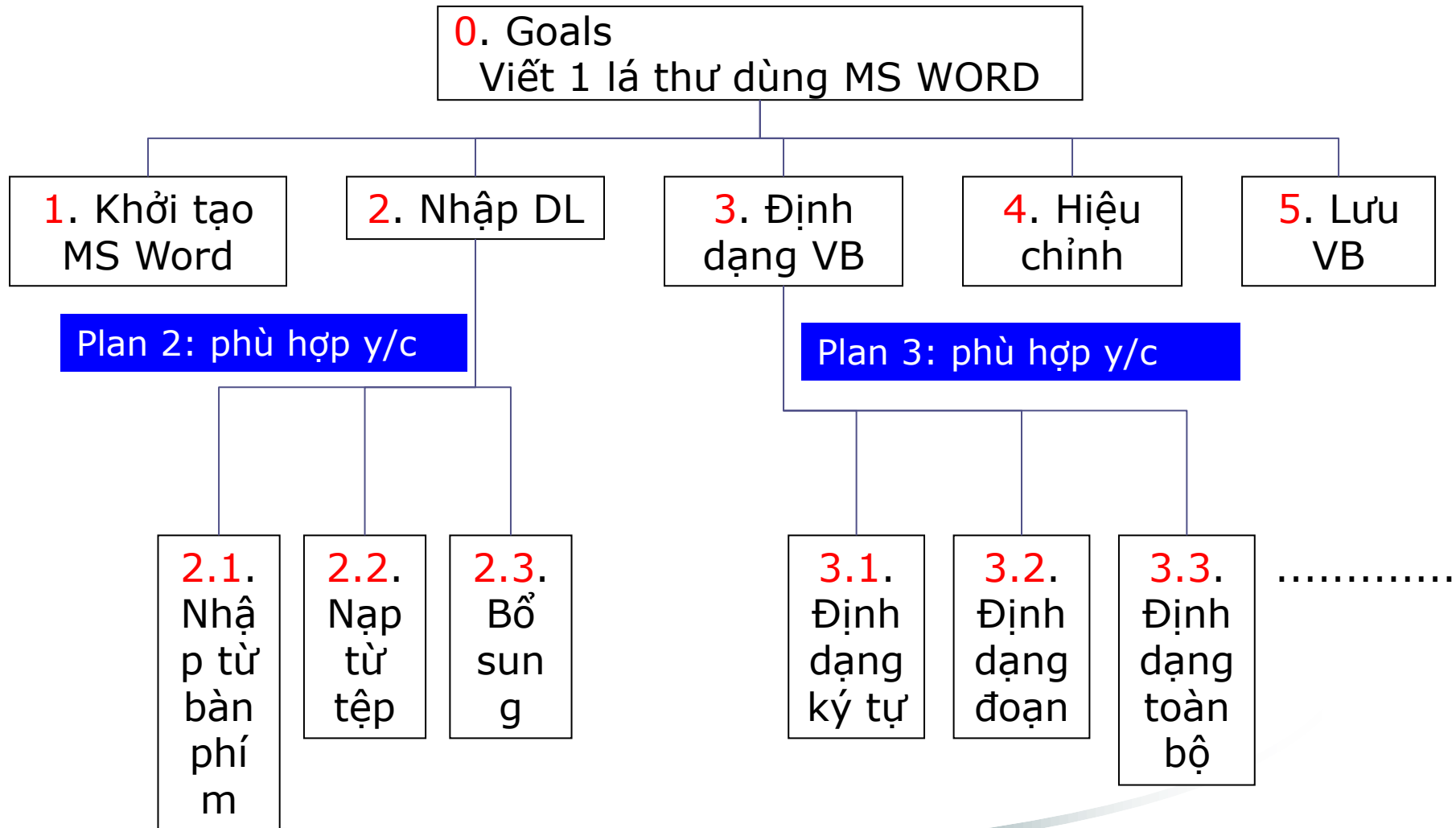
2. Sơ đồ quan hệ phân cấp các nhiệm vụ (HTA)

- HTA phân rã nhiệm vụ thành các nhiệm vụ con và được miêu tả dưới dạng văn bản hay lưu đồ.

Ví dụ: các công việc để viết một bức thư với MS WORD



Biểu diễn HTA dạng lưu đồ cho soạn thảo 1 văn bản





Biểu diễn HTA dạng văn bản

- Miêu tả phân cấp:

Mức 0:

Mức 1

Mức 2

Mức 3

Mức 3.1

Mức 3.2

.

Mức 4

.



Biểu diễn dạng văn bản(tiếp)

- Miêu tả kế hoạch

Plan 0: do 1 – 2 – 3 – 5 in that order.
when the dust bag gets full do 4

Plan 3: do any of 3.1, 3.2 or 3.3 in any order
depending on which rooms need cleaning

- Luật kết thúc: khi nào kết thúc công việc?

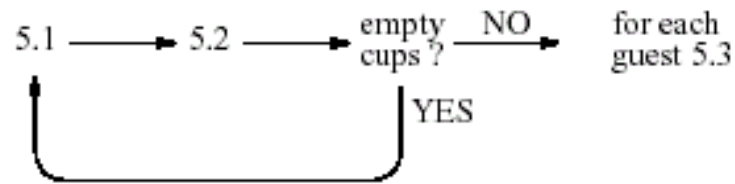


Hiệu chỉnh

- Tinh chỉnh: Cho mô tả ban đầu (Text/Diagram)
=> kiểm thử/ tăng cường?
- Nguyên tắc: dùng Heuristics
 - dựa vào cặp hành động
 - cấu trúc lại
 - cân bằng
 - khái quát hoá

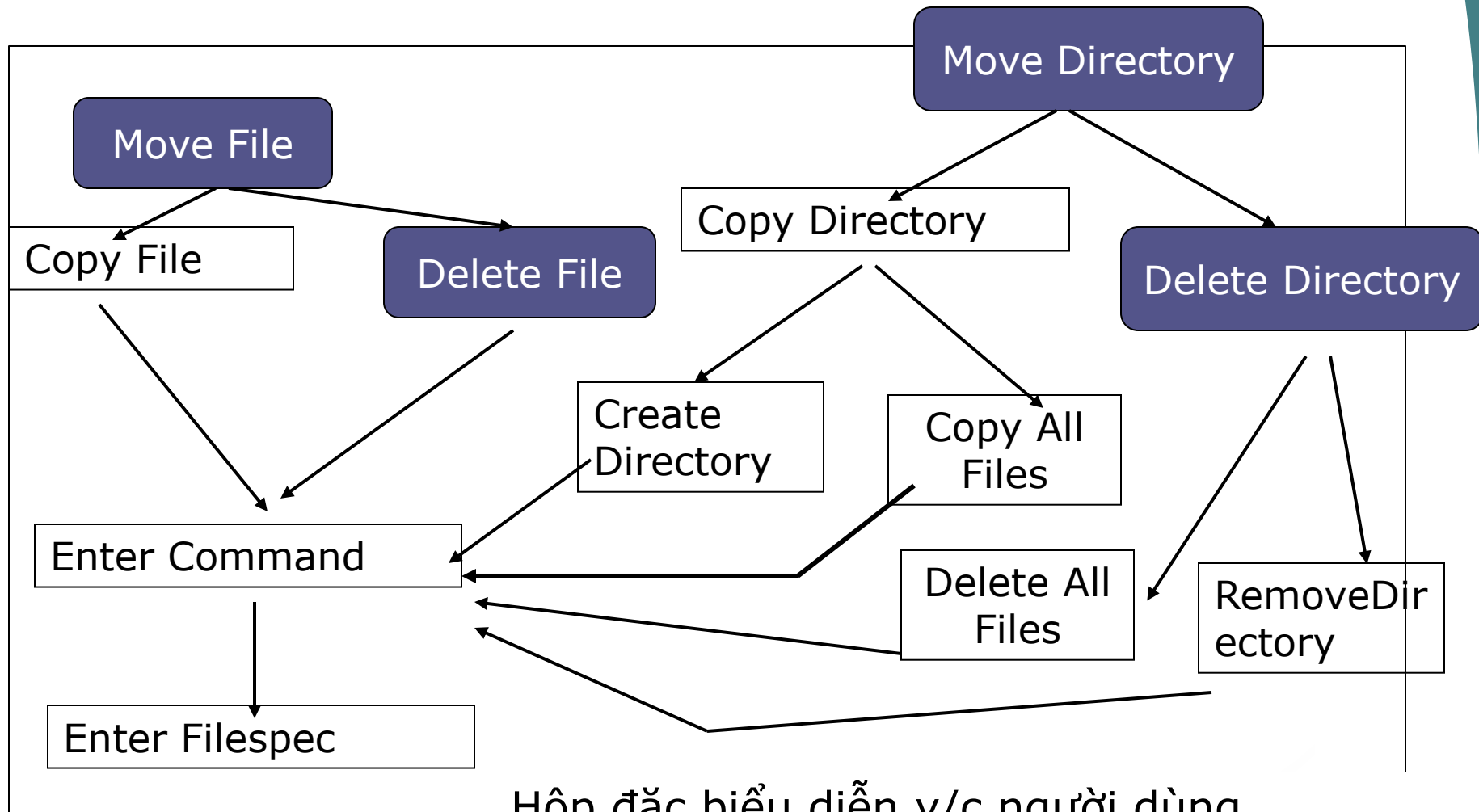
Các kiểu kế hoạch (plan)

- Chuỗi cố định: ví dụ 1.1{1.2{1.3
- Lựa chọn: ví dụ if the pot is full 2
- Đợi sự kiện: ví dụ when kettle boils 1.4
- Chu trình



- Thời gian phân chia do 1; at the same time : : :
- Trộn

Cấu trúc mục đích của PC-DOS



Hộp đặc biểu diễn y/c người dùng

Hộp rỗng là nhiệm vụ con



Cấu trúc mục đích của Macintosh

Move
File

Delete
File


Delete
File

Delete
Directory

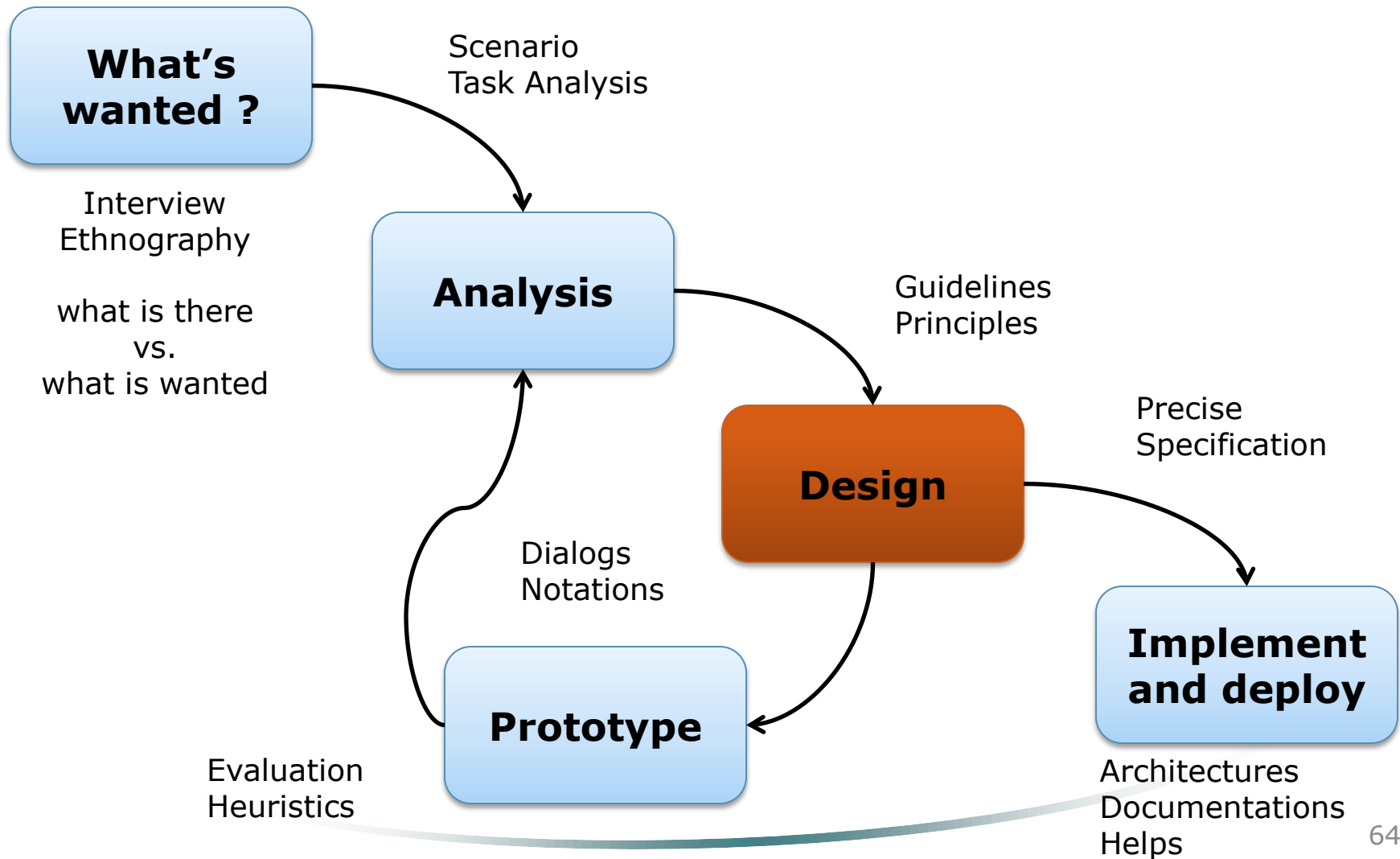
Drag Item to Destination



CHƯƠNG IV: THIẾT KẾ TƯƠNG TÁC NGƯỜI DÙNG MÁY TÍNH

- 
- I. Giới thiệu chung
 - 1. Đặc trưng
 - 2. Mục tiêu thiết kế
 - II. Mô hình thoại
 - III. Mô hình tương tác
 - IV. Thiết kế lặp
 - V. Mẫu thử

Quy trình thiết kế hệ tương tác





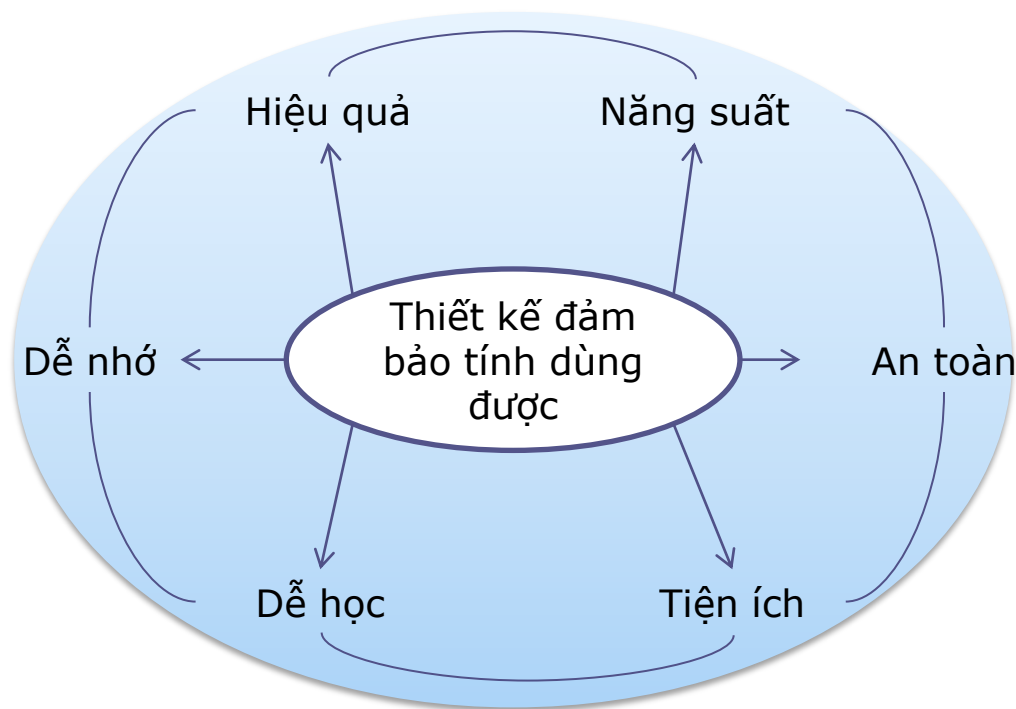
1. Đặc trưng của quá trình thiết kế tương tác

I. Giới thiệu chung

- Người dùng thường xuyên can thiệp trong quá trình thiết kế và phát triển sản phẩm mẫu.
- Các yêu cầu, mục tiêu của tính dùng được phải được định nghĩa rõ ràng, thống nhất từ khi bắt đầu của dự án
- Việc lập các bước trong quá trình thiết kế điều không thể tránh khỏi


2. Mục tiêu của thiết kế tương tác

- Thiết kế phù hợp với kinh nghiệm của người dùng
- Thiết kế đảm bảo tính dùng được






3. Cách tiếp cận

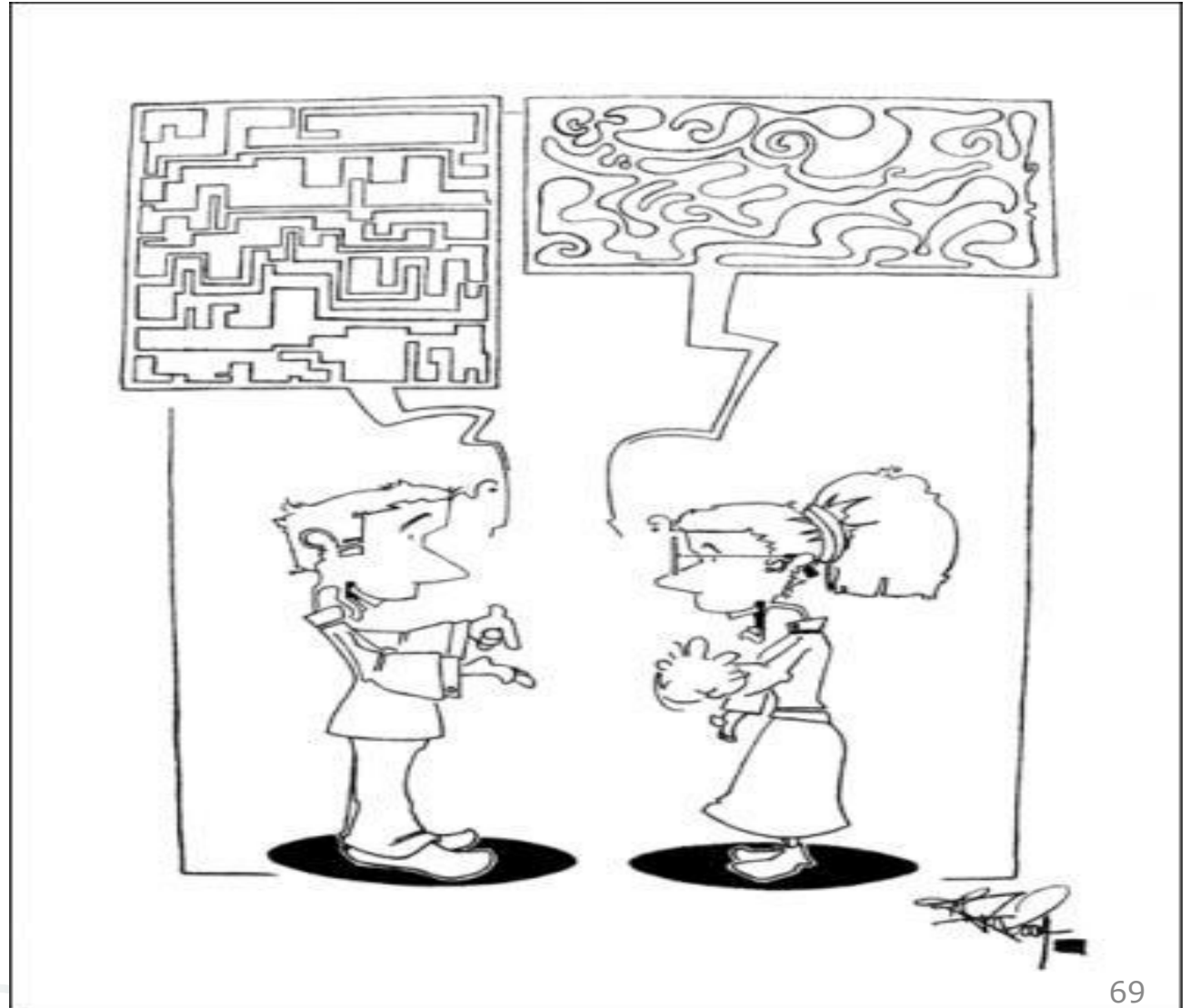
- Thiết kế hội thoại người dùng - máy tính
 - Thiết kế tương tác người dùng – máy tính
 - Thiết kế lặp và mẫu thử
- 



II. MÔ HÌNH THOẠI

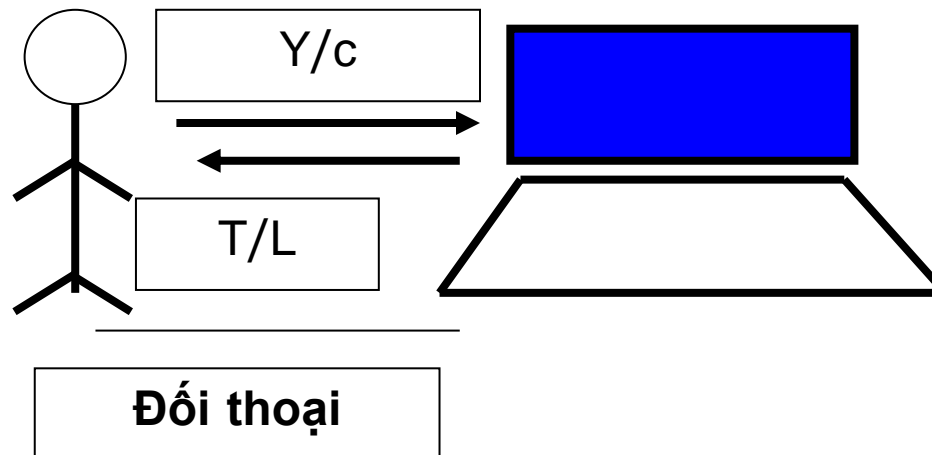
- 
1. Khái niệm
 2. Ký pháp đồ họa
 3. Ký pháp văn bản
 4. Ngữ nghĩa thoại
 5. Phân tích thiết kế thoại

1. Khái niệm



Đối thoại

- Đối thoại ngược với độc thoại, đó là sự trao đổi giữa 2 bên
- Trong thiết kế tương tác người-máy, khái niệm đối thoại tham chiếu đến cấu trúc và ngữ nghĩa của trao đổi giữa người dùng và hệ tương tác.
- *Đối thoại khá giống với **lời thoại** của 1 vở diễn, vì thế nó có thể có khá nhiều lựa chọn.*





Đối thoại người dùng có cấu trúc

- Đối thoại với MT thường có cấu trúc và bị ràng buộc.
- Các thành viên có thể trả lời những câu đã xác định trước. Tuy nhiên cũng có thể phụ thuộc các tình huống khác nhau, không lường trước.
- Chỉ có trong phim Star Trek là người dùng có thể chat một cách thoải mái với máy tính

Ví dụ về tính ràng buộc của đối thoại

- **At a marriage's service**

Minister: Do you *man's name*, take this woman

Man: I do

Minister: Do you *woman's name*, take this man ...

Women: I do

Man: With this ring, I thee wed ... (Places ring on woman's finger)

Women: With this ring, I thee wed ... (Places ring on woman's finger)

Minister: I now pronounce you husband and wife

Ký pháp biểu diễn đối thoại

- Ký pháp đối thoại: ký pháp sử dụng để mô tả đối thoại
- Một số các kỹ sư máy tính khá quen thuộc với một số ký pháp.



NNLT với các cấu trúc không đủ để mô tả đối thoại
Cần tách riêng chức năng giao tiếp và chức năng tính toán của HTTP
Có thể thay đổi kiểu giao diện và thiết kế hội thoại trước khi lập trình

Ký pháp đối thoại

- Phân loại ký pháp đối thoại:
 - Lưu đồ (diagrammatic): dễ dàng lĩnh hội
 - Văn bản (textual): dễ dàng cho việc phân tích hình thức
- Đối thoại liên kết với:
 - Ngữ nghĩa của hệ thống: cái mà nó thực hiện
 - Biểu diễn của hệ thống: đáng vẽ như thế nào

Cấp độ ngôn ngữ	Ý nghĩa	Theo ngôn ngữ tự nhiên
Từ vựng (Lexical)	Là mức độ thấp nhất. Đó là hình dạng, biểu tượng, phím nhấn	Âm thanh và đánh vần của các từ
Cú pháp (Syntactic)	Thứ tự và cấu trúc của đầu vào, đầu ra	Ngữ pháp xây dựng câu
Ngữ nghĩa (Semantic)	Ý nghĩa của hội thoại tác động lên cấu trúc dữ liệu bên trong máy tính hay với thế giới bên ngoài	Ngữ nghĩa được diễn tả bởi các đối tượng tham gia hội thoại



2. Ký pháp đồ họa

- Ký pháp đồ họa: Sử dụng các ký hiệu, biểu tượng để mô tả đối thoại
- Ưu điểm: trực quan
- Nhược điểm: chưa phải là ký pháp tốt nhất để mô tả đối thoại
- Các kiểu ký pháp đồ họa điển hình
 - Mạng dịch chuyển trạng thái (STN)
 - Mạng dịch chuyển trạng thái phân cấp (HSTN)
 - Đối thoại tương tranh và bùng nổ tổ hợp
 - Lưu đồ luồng (Flow Chart)
 - Lưu đồ JSD (Jackson Structured Design)

a. Mạng dịch chuyển trạng thái (State transition networks - STN)

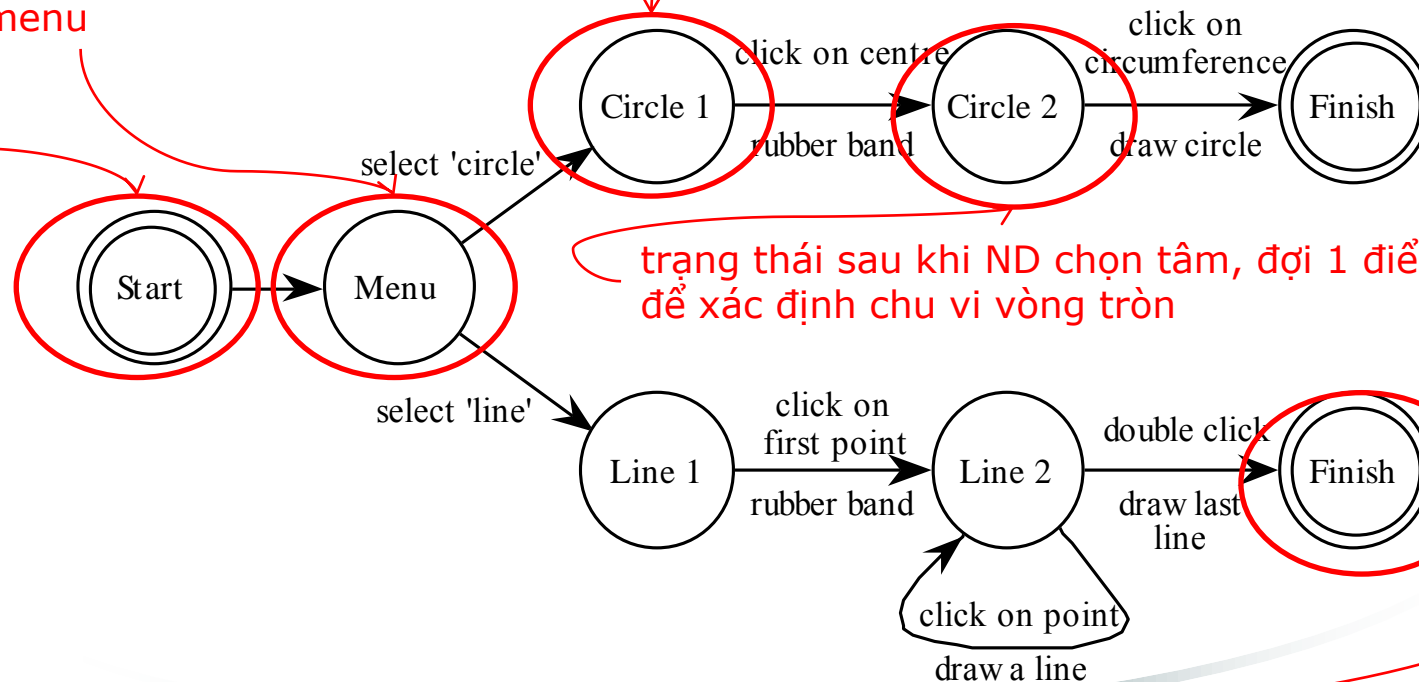
- Mạng dịch chuyển trạng thái đã được sử dụng từ rất sớm để mô tả đối thoại (1960)
- Dùng 2 đối tượng để mô tả:
 - Hình tròn: mô tả 1 trạng thái của hệ thống
 - Mũi tên: mô tả dịch chuyển trạng thái - hành động hay sự kiện.
 - Mũi tên, vòng tròn có thể có nhãn.

Ví dụ: Mạng dịch chuyển trạng thái biểu diễn công cụ vẽ

- Giao diện kiểu thực đơn, gồm 2 lựa chọn: vẽ vòng tròn và vẽ đường thẳng
- Trạng thái: hình tròn
- Sự kiện, hành động: mũi tên

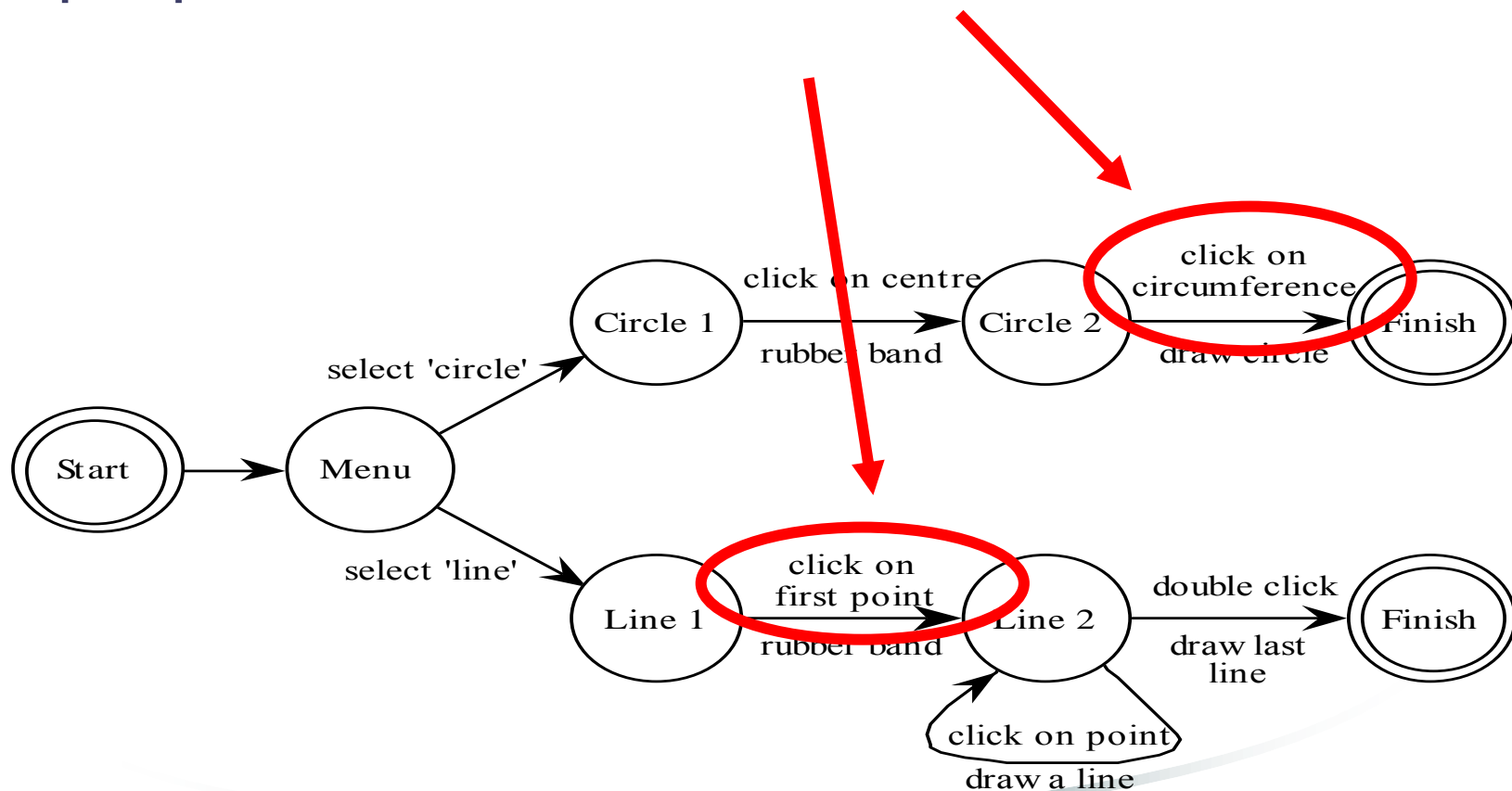
HT đợi ND nhấn "circle" hay "line" từ menu

HT đợi ND chọn tâm vòng tròn



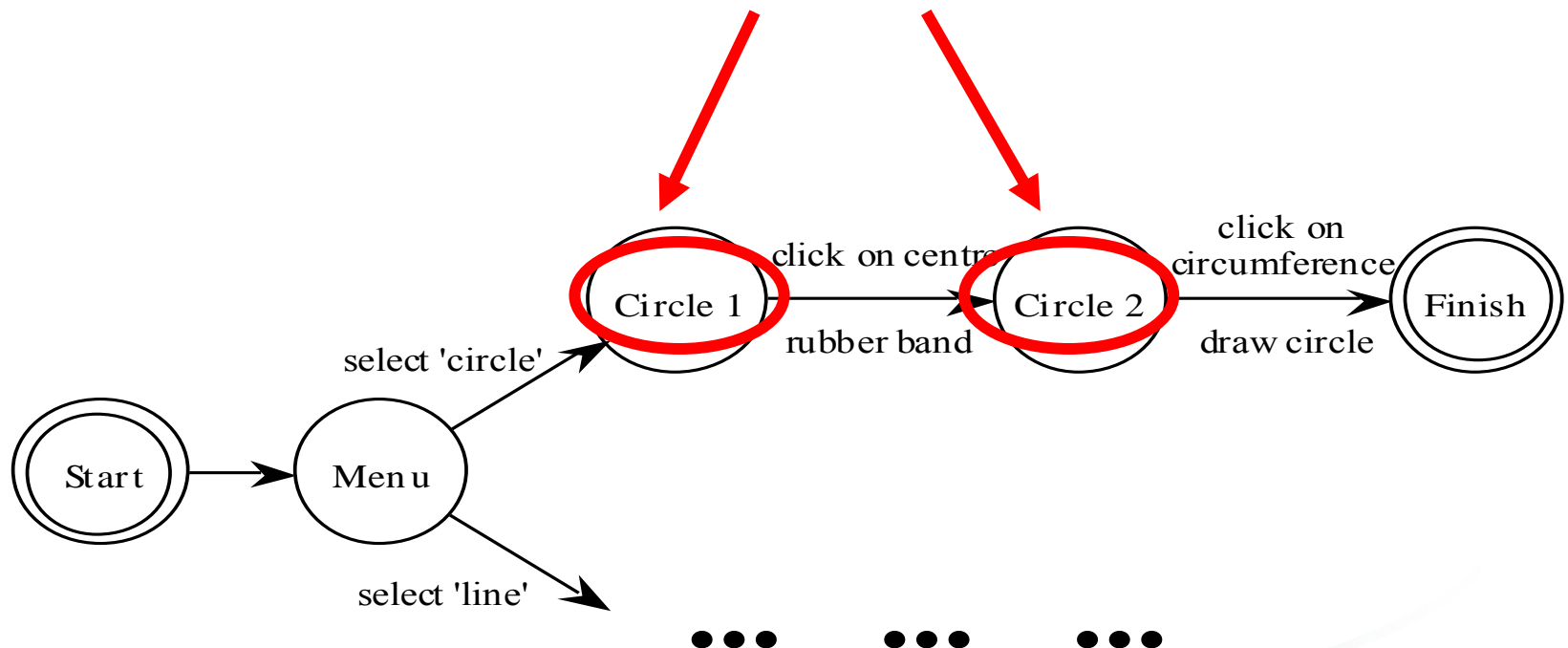
STN: sự kiện

- Nhãn của mũi tên thường tương đối nặng vì các sự kiện đòi hỏi cần chi tiết

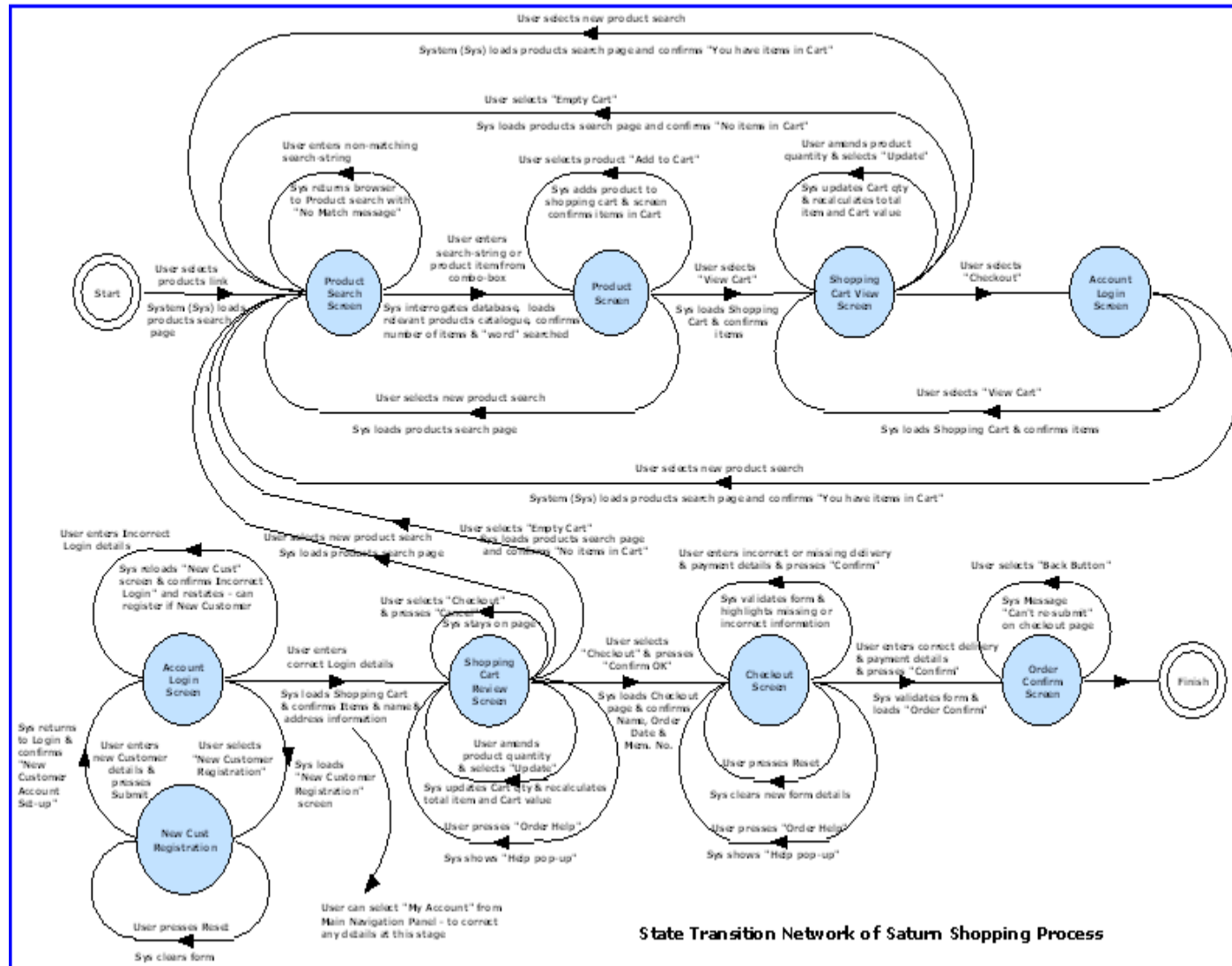


STN: trạng thái

- Nhấn trong các đường tròn thì thường ít thông tin vì trạng thái thường khó đặt tên nhưng lại dễ hiển thị

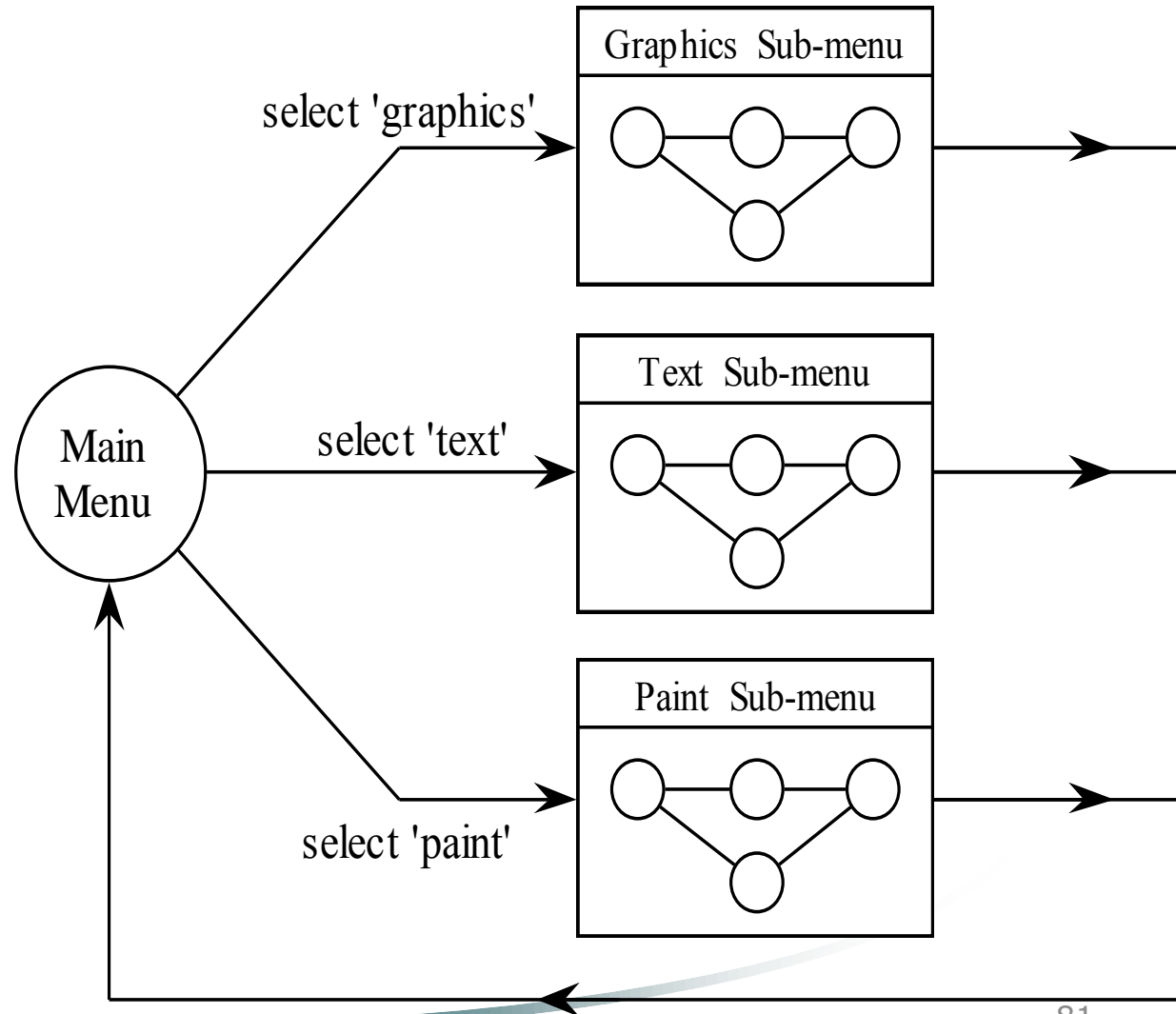


STN thường khá phức tạp



b. Mạng dịch chuyển trạng thái phân cấp - HSTN

- Được sử dụng khi đối thoại khá phức tạp.
- Người ta chia đối thoại thành các đối thoại nhỏ (sub-dialog)
- Ví dụ: Mạng dịch chuyển trạng thái phân cấp cho 1 thực đơn chính gồm 3 thực đơn con





Sử dụng STN phân cấp

- Ưu điểm
 - Là điểm khởi đầu tốt để tạo mẫu thử.
 - Có thể duyệt toàn bộ kịch bản với ND hay khách hàng => giải thích nhờ STN
 - Biểu diễn tốt đối thoại tuần tự, chọn hay lặp.
 - Vẽ tay dễ dàng nếu đối thoại đơn giản
 - Có nhiều phần mềm hỗ trợ vẽ STN: Hypercard, Macromedia Director, v.v.
- Nhược điểm:
 - Kém tác dụng nếu phải biểu diễn đối thoại tương tranh.

c. Đối thoại tương tranh

- Ví dụ: Giao diện thực đơn gồm 3 lựa chọn (dạng phím Toggle). Mỗi phím tương ứng với một kiểu định dạng: đậm, nghiêng hay gạch chân.
- Một đoạn văn bản có thể là nghiêng, đậm hay gạch chân và cũng có thể là bất kỳ tổ hợp nào của 3 thuộc tính trên.

Text Style

☐ **bold**

example ☒ *italic*

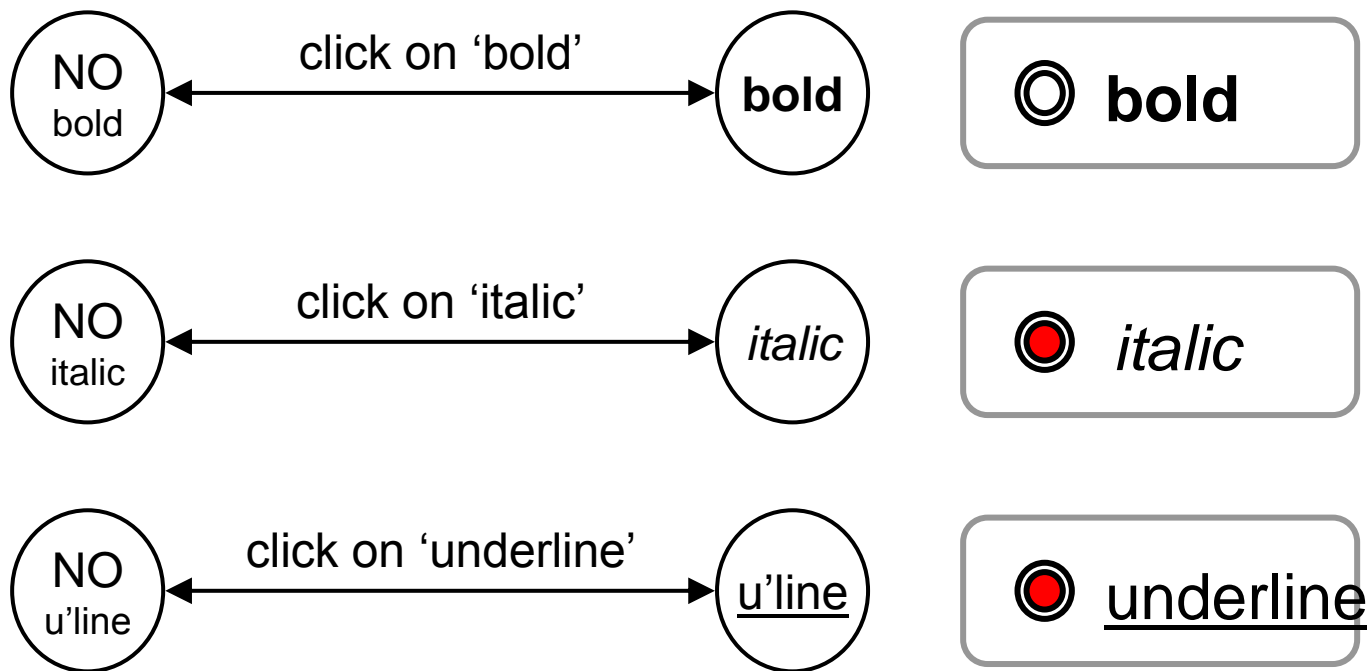
☒ underline

Nếu quan sát từng phím, có 1 STN 2 trạng thái.

Nếu muốn biểu diễn tổ hợp trạng thái, cần tổ hợp lưu đồ.

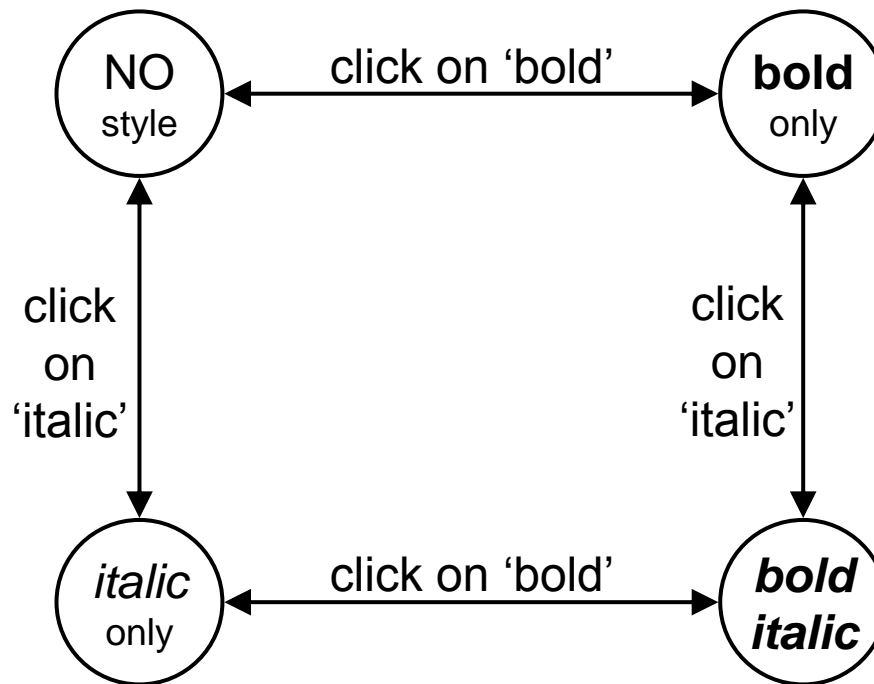
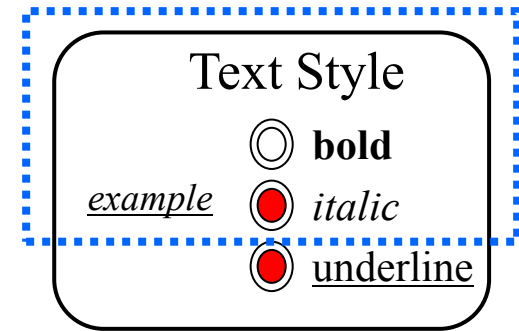
Đối thoại tương tranh

1 STN per toggles



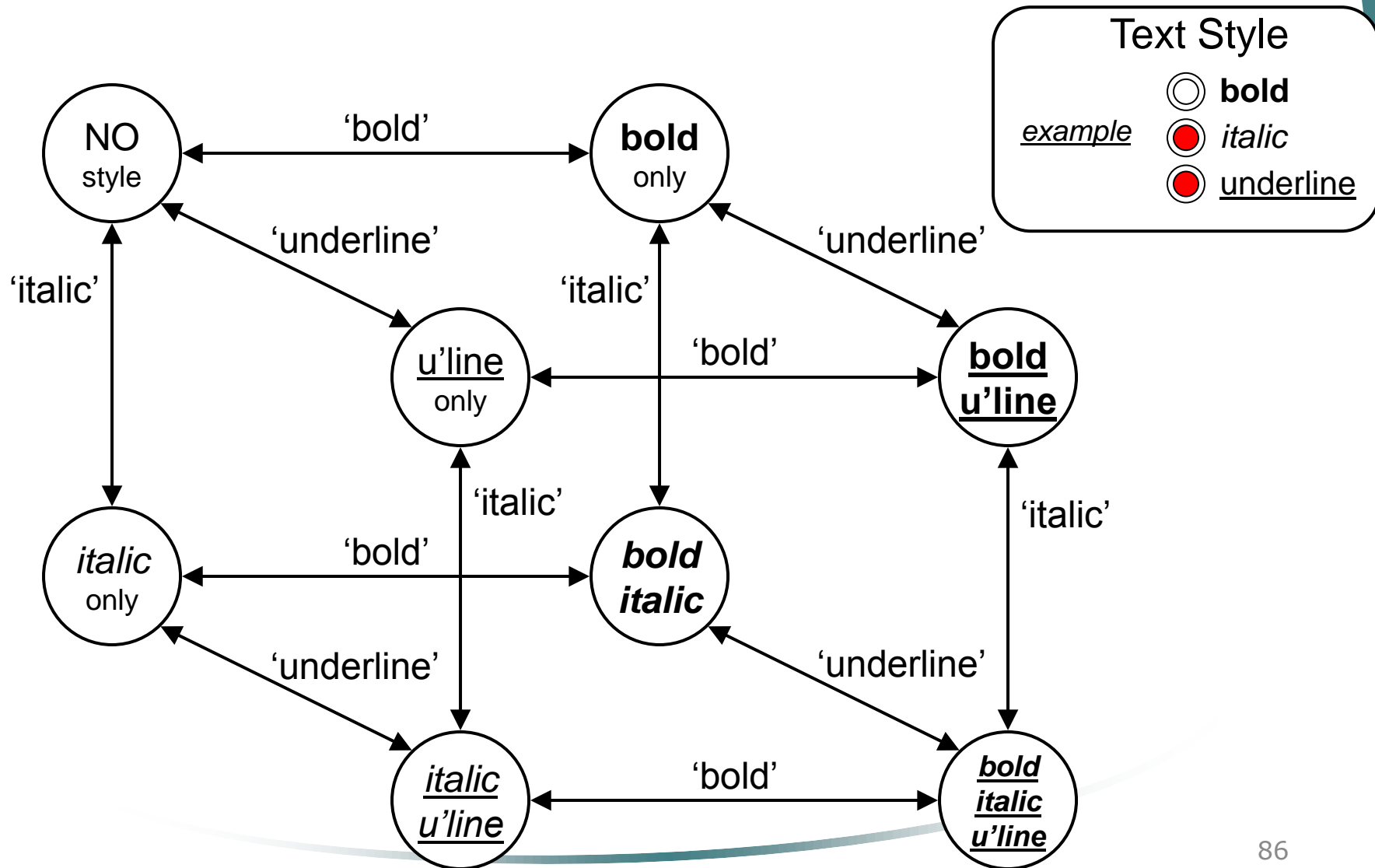
Đối thoại tương tranh

Tổ hợp nghiêng – đậm



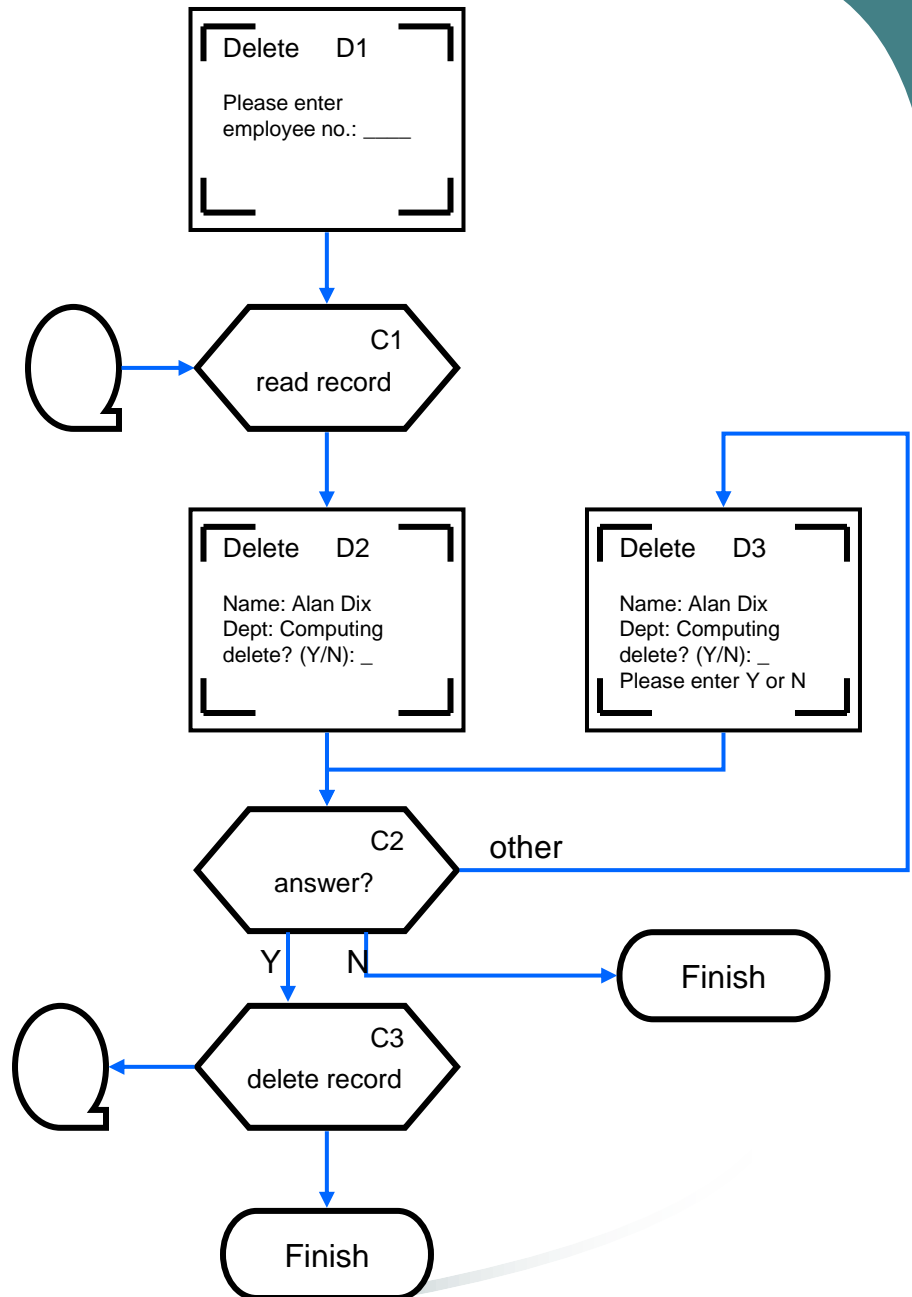
Đối thoại tương tranh

Bùng nổ tổ hợp đậm – nghiêng – gạch chân



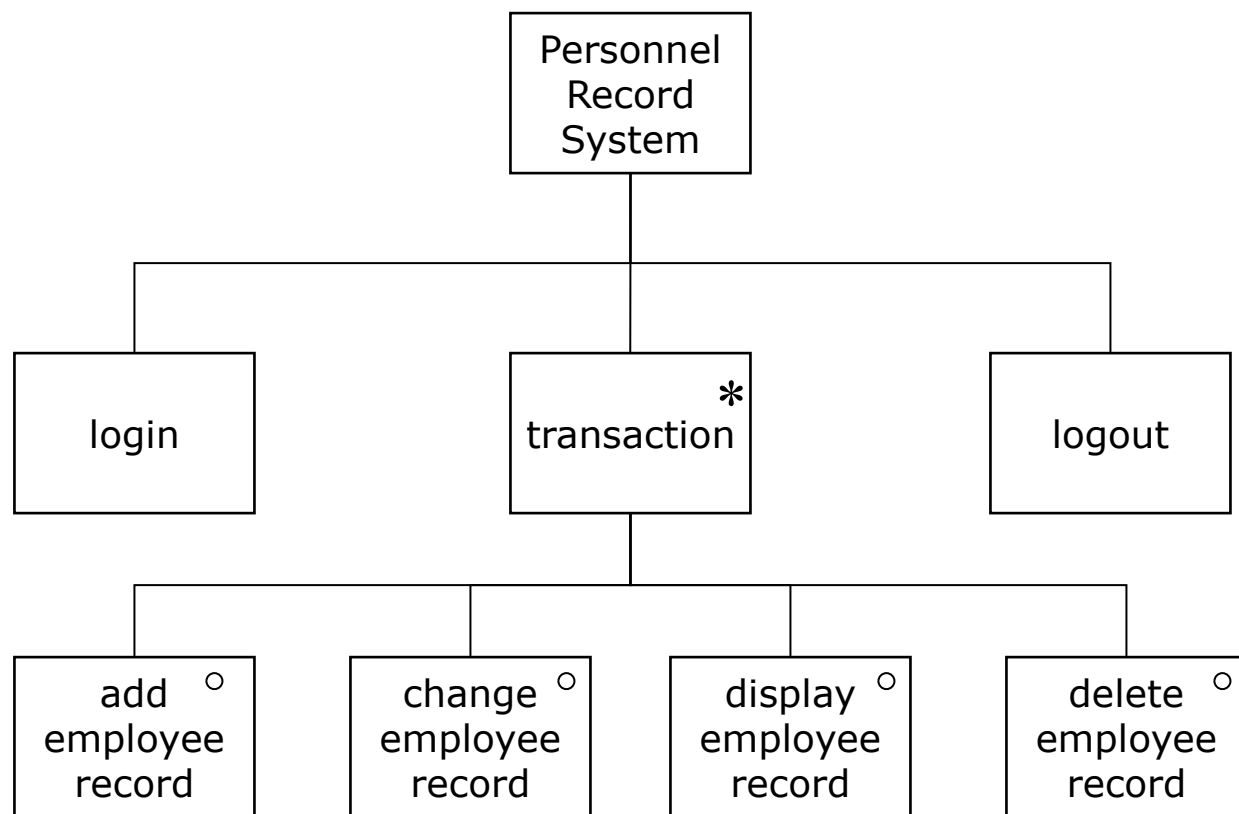
d. Lưu đồ luồng (Flow charts)

- Sử dụng nhiều loại hình khối khác nhau để biểu diễn các hoạt động khác nhau
 - Hình khối biểu diễn quy trình hay sự kiện
 - Hình khối không biểu diễn trạng thái
- Quen thuộc với lập trình viên, biểu diễn đối thoại trên quan điểm của lập trình viên hơn là quan điểm của người dùng



Lưu đồ JSD

- Ra đời sau Flow Charts
- Thích hợp với cấu trúc đối thoại hình cây
- Khả năng diễn tả ngữ nghĩa kém, nhưng diễn tả cấu trúc rõ ràng.
- Ví dụ: HT cho phép cập nhật thông tin về nhân sự: bổ sung, hiện, xoá,...





3. Ký pháp văn bản

- Được sử dụng song song với ký pháp đồ họa
- Các ký pháp văn bản tiêu biểu
 - Văn phạm
 - Luật sản xuất
 - CSP (Communicating Sequential Process)



a. Văn phạm (Textual grammars)

- Văn phạm hình thức được dùng phổ biến như một ký pháp văn phạm, ví dụ BNF
- Ưu điểm: đa dạng hơn so với biểu thức chính quy hay STN
- Nhược điểm: Không có biểu diễn tương tranh

b. Luật sản xuất: Production rules

- Là một chuỗi các lệnh không có trật tự:

if condition then action

- Các điều kiện dựa trên trạng thái hoặc sự kiện đang treo
- Hệ thống luật sản xuất có thể là hướng sự kiện hoặc hướng trạng thái hoặc cả hai
- Ưu điểm: Tốt để biểu diễn tương tranh
- Nhược điểm: Không thích hợp cho tuần tự hay trạng thái

Ví dụ về luật sản xuất

```
sel-line → first  
C-point first → rest  
C-point rest → rest  
D-point rest → <draw line>
```

- Sự kiện người dùng: Bắt đầu với chữ hoa: Sel-line
- Sự kiện trong: Bắt đầu với chữ thường. Dùng trong đối thoại để lưu vết của trạng thái đối thoại. Ví dụ rest là trạng thái sau khi điểm đầu tiên được chọn
- Đáp ứng của hệ thống: thường nằm trong cặp <>. Đó là hiệu ứng nhìn thấy hoặc nghe thấy của hệ thống.

Ví dụ về luật sản xuất (tt)

Sel-line → first

C-point first → rest

C-point rest → rest

D-point rest → <draw line>

- Một luật được áp dụng nếu:
 - Mọi sự kiện trong phần điều kiện của nó hiện diện trong bộ nhớ
 - Và mọi tương tác của người dùng được thực hiện ngay lập tức bởi sự kiện này
 - Ví dụ: Sự kiện người dùng nhấn chuột được bổ sung vào bộ nhớ và đáp ứng của hệ thống là đường được vẽ (<draw line>)
- Khi một luật được áp dụng:
 - Mọi sự kiện trong điều kiện được loại khỏi bộ nhớ hệ thống
 - Các sự kiện trong phần hành động sẽ được bổ sung vào bộ nhớ
 - Ví dụ:
Khi người dùng chọn Sel-line từ menu, bộ nhớ của hệ thống sẽ chứa Sel-line.
Khi luật thứ nhất được áp dụng, Sel-line bị loại bỏ khỏi hệ thống và thay thế vào đó là first

Luật sản xuất hướng trạng thái

- Các thuộc tính

Mouse: { mouse-off, select-line, click-point, double-click }
Line-state: { menu, first, rest }

- Các luật

select-line → mouse-off first,
start-line,
high-light line

click-point start line → mouse-off,
rest-line,
rubber-band on

click-point rest line → mouse-off,
draw line

double-click rest line → mouse-off,
menu,
draw line,
rubber-band off

c. Quy trình giao tiếp tuần tự và đại số quy trình

- STN không phù hợp với mô tả tương tranh
- Luật sản xuất thì không phù hợp với mô tả tuần tự, trạng thái
- Việc xử lý tương tranh + tuần tự đặt ra trong nhiều bài toán: truyền thông, điều khiển tương tranh
- Đại số quy trình (Process Algebra) là một ký pháp hình thức được phát triển cho các quá trình như thế
- CSP (Communicating Sequential Process): là một lớp con được phát triển cho đặc tả đối thoại cả tuần tự lẫn tương tranh

Quy trình giao tiếp tuần tự CSP

- Quá trình tương tranh được dùng như 1 cách thức để tổ chức cấu trúc trong của giao tiếp.
 - Ví dụ: khi lựa chọn, ND có thể dùng chuột hay dùng phím nóng => mỗi lựa chọn một quá trình.
- Quá trình chuột đơn giản là đợi ND chọn 1 mục trên menu và tiếp sau là 1 sự kiện trong phụ thuộc vào sự lựa chọn.
- Quá trình bàn phím điều khiển bởi phím Alt và tiếp sau cũng là 1 sự kiện trong .

Quy trình giao tiếp tuần tự CSP

- CSP được sử dụng vì:
 - đặc tả cho cả tuần tự và tương tranh
 - dễ hiểu

- Ví dụ:

Hành động của người dùng

Draw-menu = (select - circle ? -> Do-circle
Lựa chọn ← [] select-line ? -> Do-line)

Do-circle = click ? -> set centre -> Sự kiện tuần tự
click ? Draw-circle -> Skip

Do-line = Start-line ; Quá trình tuần tự
Rest-line

Start-line = click ? -> first-point -> Skip

Rest-line = (click ? -> next-point -> Rest-line
[] double-click ? -> last-point -> Skip)

Quy trình giao tiếp tuần tự CSP

- Do-circle là hoàn toàn tuần tự. Khi HT thực hiện Do-circle, trước tiên cần ND nhấn phím chuột, tiếp sau là 1 sự kiện trong "set centre" để xác định vị trí con trỏ. Tiếp theo nhận 1 lần nhấn chuột rồi vẽ và kết thúc bởi Skip.
- Do-line cũng là tuần tự. Dấu ";" để chỉ quá trình tuần tự, cái xảy ra giữa 2 quá trình. Dấu "->" chỉ dùng sau 1 sự kiện.
- []: chỉ ra sự lựa chọn như dòng 1: ND có thể chọn circle hay chọn line.

Hành động của người dùng

Draw-menu = (select - circle ? -> Do-circle
Lựa chọn [select-line ? -> Do-line)

Do-circle = click ? -> set centre -> Sự kiện tuần tự
click ? Draw-circle -> Skip

Do-line = Start-line ; Quá trình tuần tự
Rest-line

Start-line = click ? -> first-point -> Skip

Rest-line = (click ? -> next-point -> Rest-line
[] double-click ? -> last-point -> Skip)

4. Ngữ nghĩa đối thoại

- Mục đích của miêu tả đối thoại chỉ là: giao tiếp giữa các nhà thiết kế hay như là công cụ trong thiết kế.
- Miêu tả đối thoại dùng để đặc tả hình thức, có thể là 1 phần của hợp đồng hay thực hiện một nguyên mẫu
=> cần có miêu tả một cách hình thức ngữ nghĩa của đối thoại.
- Có 2 sắc thái của ngữ nghĩa đối thoại:
 - Phía trong của UD
 - Phía ngoài của biểu diễn.
- Ba cách tiếp cận:
 - Ngữ nghĩa đặc tả ký hiệu
 - Liên kết với ngôn ngữ lập trình
 - Liên kết với ký pháp đặc tả



a. Ngữ nghĩa đặc tả ký hiệu

- Mục đích của mô tả đối thoại:
 - thông báo giữa các nhà thiết kế
 - Là một công cụ để suy nghĩ
- Khi đó cần:
 - Chú thích cho đối thoại hình thức với chủ ý của hành động
 - Hoặc để cho độc giả tự suy luận về ngữ nghĩa
- Có hai khía cạnh nên được xem xét
 - Đối với ứng dụng
 - Đối với việc trình diễn



Augmented Transition NetWorks (ATN)

- ATN là 1 dạng của mạng dịch chuyển trạng thái.
- HT giả định là 1 tập các thanh ghi lưu trữ vị trí mà mạng dịch chuyển có thể thiết lập và kiểm thử.
 - Mỗi cung trong mạng ngoài việc có thể có nhãn, nó có thể có ràng buộc (như 1 sự kiện) và chỉ có cung nếu ràng buộc là thoả và sự kiện xảy ra.
 - Luật sản xuất cũng có nhiều cách thức và ngữ ngữ của nó cũng thay đổi tương tự.



b. Liên kết với NNLT

- Các ký pháp thường gắn với 1 NNLT nào đó.
- Ví dụ: ký pháp về biểu thức chính qui thường dùng C để biểu diễn ngữ nghĩa đối thoại

Ví dụ: Công cụ để đọc 1 số, viết trong C:

- Dùng ký pháp của biểu thức chính quy có bổ sung các phép toán
 - ‘?’ : chỉ tuần tự, ‘+’: chỉ sự lựa chọn
 - ‘key : | điều kiện| ‘: biểu thức chỉ sánh được nếu điều kiện là thỏa.
- Đặc tả biểu thức ‘input tool’ như sau: Tool chỉ ra một công cụ mới, tương tự như ký tự không kết thúc của BNF và biểu thức là nằm trong “input”.
- Các tool được bố trí kiểu phân cấp => *digit*, *sign* và *return tools* là riêng của *tool number*.
- Echo: trả lại ký tự cho ND

```
Tool number
{ char buf [80];
  int index, positive;
  input {( digit * + sign; digit; digit*)
        return)
  tool digit
  { input {key:| key_c >='0' && key_c <='9' | }
    if (index < 79) /* noi ky tu vao xau *
    { buf [index] = key_c; index = index + 1;
      echo (key_c); }
    }
  tool sign
  | . . . . .
  tool return { input {key: | key_c=='\n'}
    . . . . }
  ... }
```



c. Liên kết với đặc tả hình thức

- SPI (Specifying and Prototyping Interaction) bao gồm 2 phần:
 - event CSP: ký pháp tương tự như CSP
 - event ISL: mô tả ngữ nghĩa đối thoại
- Phần CSP được mô tả như trong CSP, tuy nhiên, mỗi sự kiện của nó ứng với 1 sự kiện trong ISL
- ISL: chuẩn cục bộ và phụ thuộc cục bộ vào ngôn ngữ chủ.

Ví dụ: Event CSP mô tả quá trình login

```
Login  = login-mess -> get-name -> Passwd
Passwd = passwd-mess -> (  invalid ->Login
                        [] valid Session)
Session = (  logout -> Login
           [] command -> execute -> Session)
```

- Một đăng nhập bị lỗi có thể mô tả như sau:
 - login: toto
 - passwd: b9fkG
 - Sorry bad user-id/password

Ví dụ: Event ISL mô tả 3 sự kiện
login-mess, get-name và valid

```
event login-mess =  
  prompt: true  
  out    : "login"  
event get-name =  
  uses: input  
  set user-id = input  
event: valid  
  uses: input, user-id, passwd-db  
  wgen: passwd-id = passwd-db(user-id)
```



5. Phân tích và thiết kế đối thoại

- Các cách thức mà đối thoại có thể được phân tích nhằm phát hiện tính tiện dụng tiềm năng bằng cách xem xét các nguyên lý thiết kế giao diện
- Trước tiên tập trung **vào hành động** của ND, tiếp theo **là trạng thái** của đối thoại. Cuối cùng là xem xét cách **biểu diễn và từ vựng**.

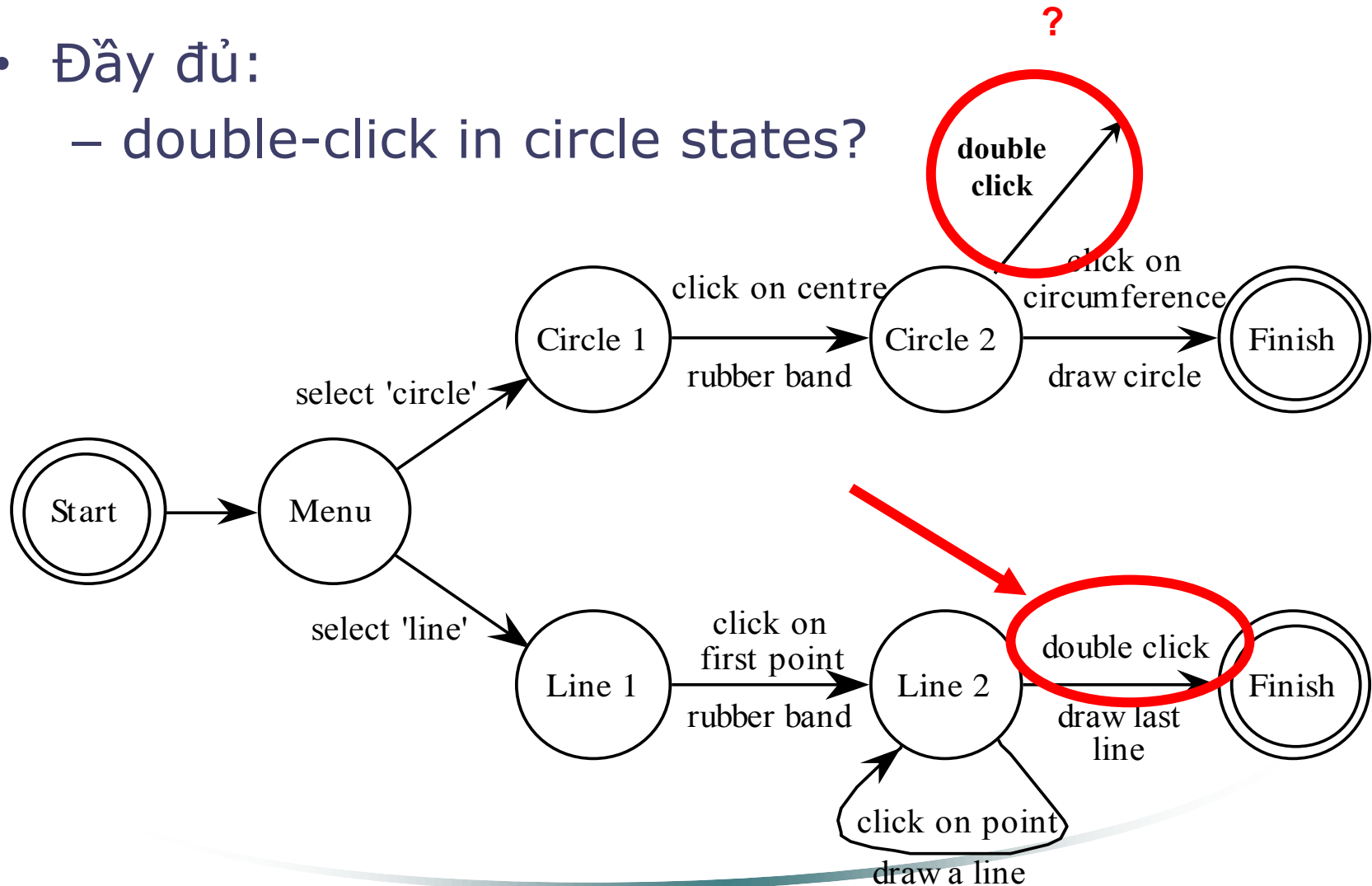


a. Tính chất của hành động

- Hành động có hợp lý không ?
- Đầy đủ: completeness
 - Các cung bị thiếu: missed arcs
 - Các trường hợp bất khả kháng: unforeseen circumstances
- Xác định: determinism
 - Nhiều cung cho một hành động
 - Cung cấp quyết định ứng dụng
 - Chú ý: luật sản xuất
 - Thoát nhiều mức lồng nhau
- Nhất quán
 - Cùng hành động, cùng hiệu quả
 - Thể thức và tính quan sát được


Tính chất của hành động

- Đầy đủ:
 - double-click in circle states?



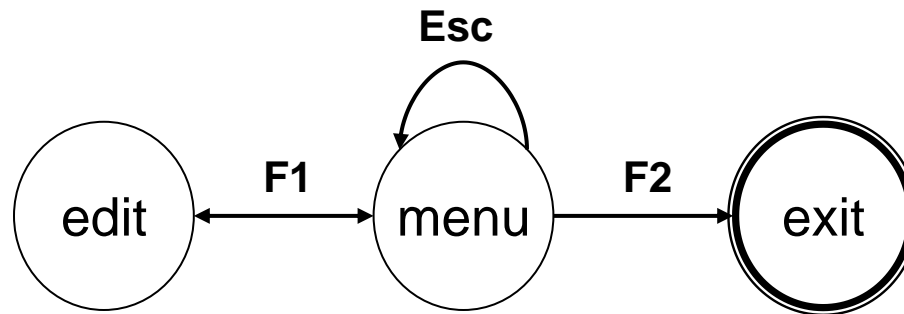


b. Tính chất của trạng thái

- Người dùng có đạt tới trạng thái mong muốn không ?
 - Tính đạt tới được
 - Nhận được mọi thứ từ bất kỳ vị trí nào
 - Dễ dàng
 - Tính thuận nghịch
 - Có thể nhận được trạng thái trước?
 - Nếu không: Undo
 - Các trạng thái nguy hiểm
 - Các trạng thái không muốn xảy ra
- 

Ví dụ: Trạng thái nguy hiểm của bộ xử lý văn bản

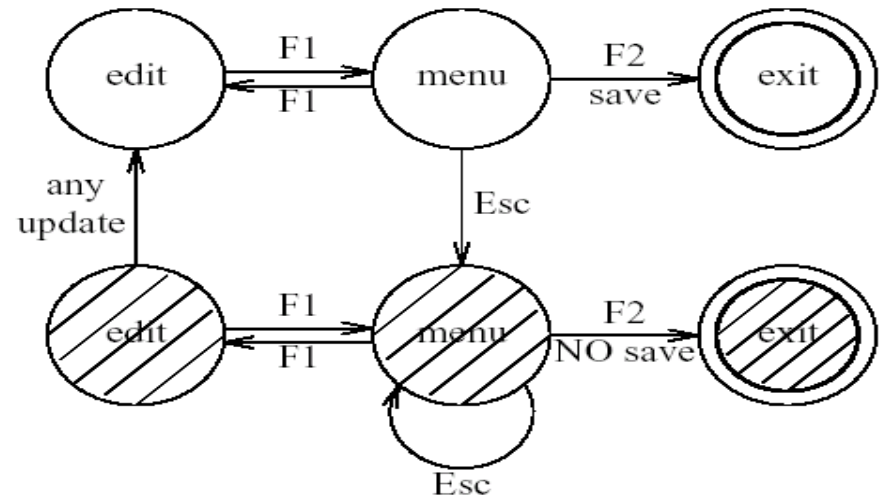
- Có 2 chế độ và thoát
 - F1 - thay đổi chế độ
 - F2 - thoát (và tự động ghi nội dung)
 - Esc - không thay đổi chế độ



- Nhưng ... Esc không tự động ghi lại nội dung

Ví dụ: Trạng thái nguy hiểm của bộ xử lý văn bản

- Thoát có/không tự động ghi là các trạng thái nguy hiểm
- Cần phân biệt rõ 2 trạng thái



F1-F2 – exit with save

F1-Esc-F2 – exit *no* save



c. Tính chất của biểu diễn và từ ngữ

- Hình thức và chức năng của hội thoại ?
- Tính trừu tượng: Biểu diễn trừu tượng của hội thoại.
 - Ví dụ, nhập tọa độ một điểm từ bàn phím hay click chuột lên bề mặt đối tượng
- Các chế độ nhận: Tập trung vào tính dễ nhìn, dễ quan sát và dễ dự đoán của các nhãn biểu diễn
- Tính phù hợp của kiểu hội thoại: Sử dụng kiểu hội thoại phù hợp cho từng loại giao diện.
 - Ví dụ, giao diện dòng lệnh khác với giao diện WIMP.

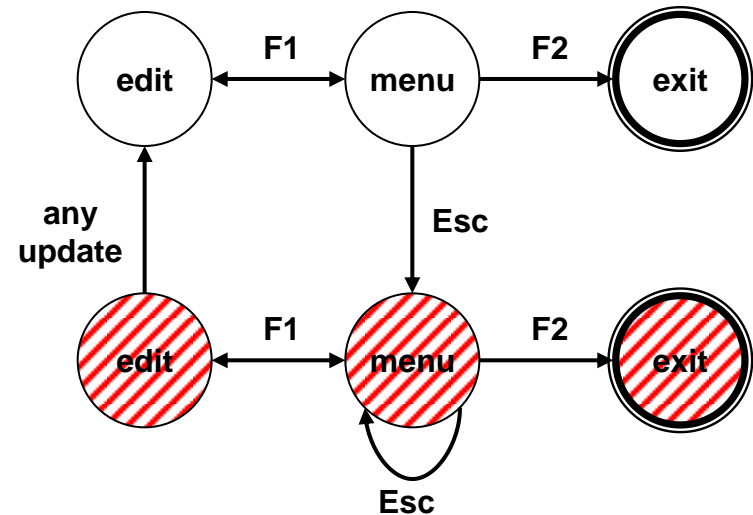
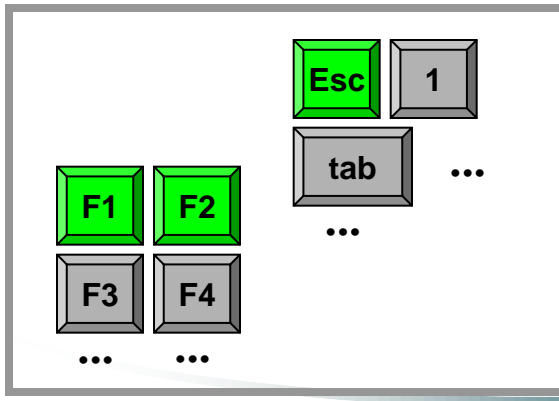


Ví dụ: Mô thức từ vựng của bộ xử lý văn bản

- Trực quan
 - Phân biệt được các chế độ và trạng thái
 - Ký pháp thoại
- Kiểu từ vựng
 - Danh từ chỉ việc thực hiện các lệnh (command - verb noun)
 - Động từ chỉ các thao tác với chuột (mouse based - noun verb)
- Hiện thị

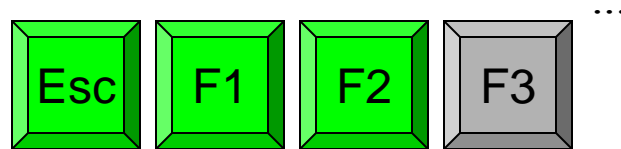
Ví dụ: hiện thị trạng thái nguy hiểm của bộ xử lý văn bản

- old keyboard - OK



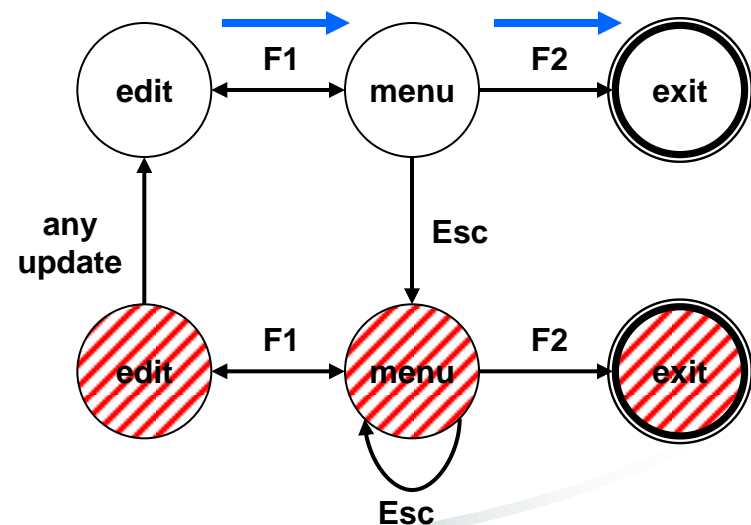
Ví dụ: hiện thị trạng thái nguy hiểm của bộ xử lý văn bản

- new keyboard layout



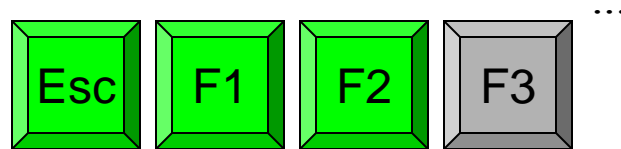
intend F1-F2 (save)

finger catches Esc



Ví dụ: hiện thị trạng thái nguy hiểm của bộ xử lý văn bản

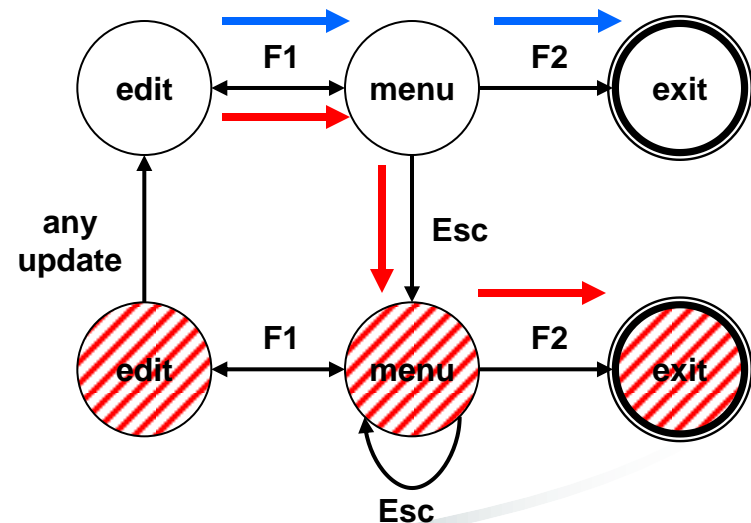
- new keyboard layout



intend F1-F2 (save)

finger catches Esc

F1-Esc-F2 - disaster!





III. MÔ HÌNH TƯƠNG TÁC

- 
1. Mô hình PIE
 2. Phân tích trạng thái-sự kiện

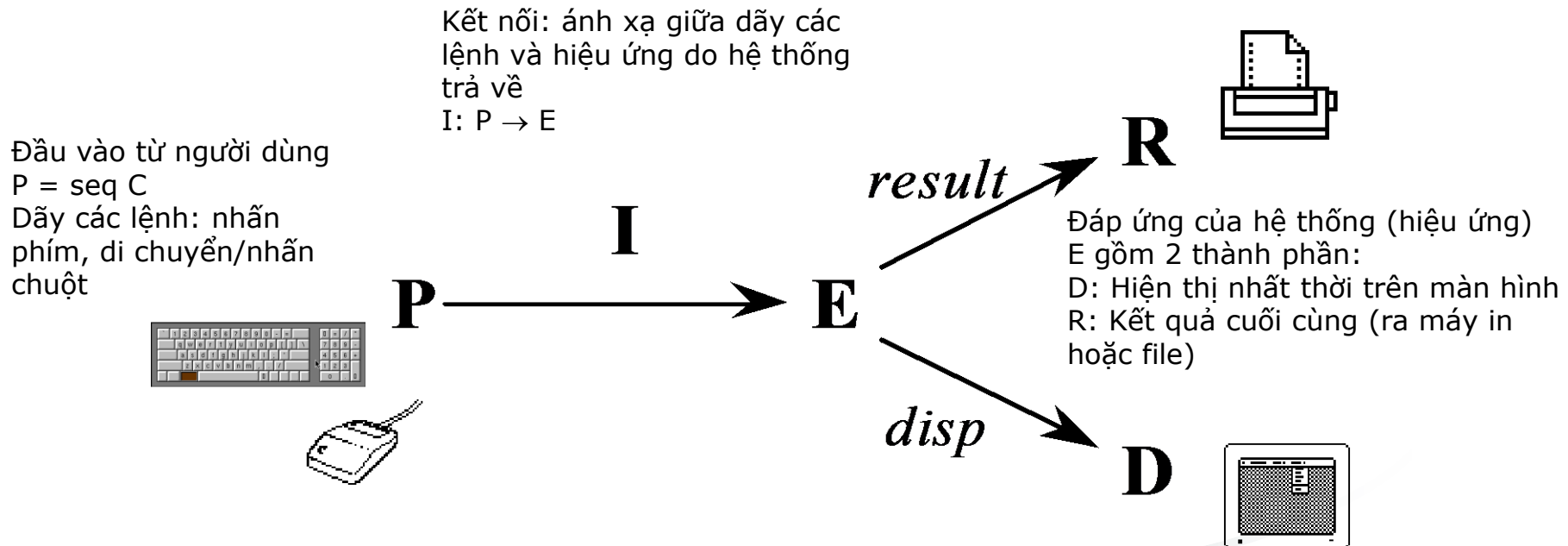


Mô hình tương tác

- Hệ thống thiết kế theo các mô hình phần mềm thông dụng thường ít quan tâm đến người dùng
- Cần dung hòa giữa mô hình phần mềm và mô hình tương tác.
 - Hình thức
 - **Mô hình PIE:** diễn tả các đặc tính tương tác tổng quát hỗ trợ tính dùng được
 - Phi hình thức
 - Kiến trúc tương tác (MVC, PAC, ALV) cho phép phân tách và mô đun hóa phần chức năng và phần trình diễn của ứng dụng
 - Bán hình thức
 - **Phân tích trạng thái-sự kiện** để quan sát một phần của hệ tương tác trên nhiều lớp.

1. Mô hình PIE

- Hộp đen tối thiểu của hệ tương tác
- Tập trung vào các khía cạnh tương tác quan sát được từ bên ngoài






2. Phân tích trạng thái – sự kiện

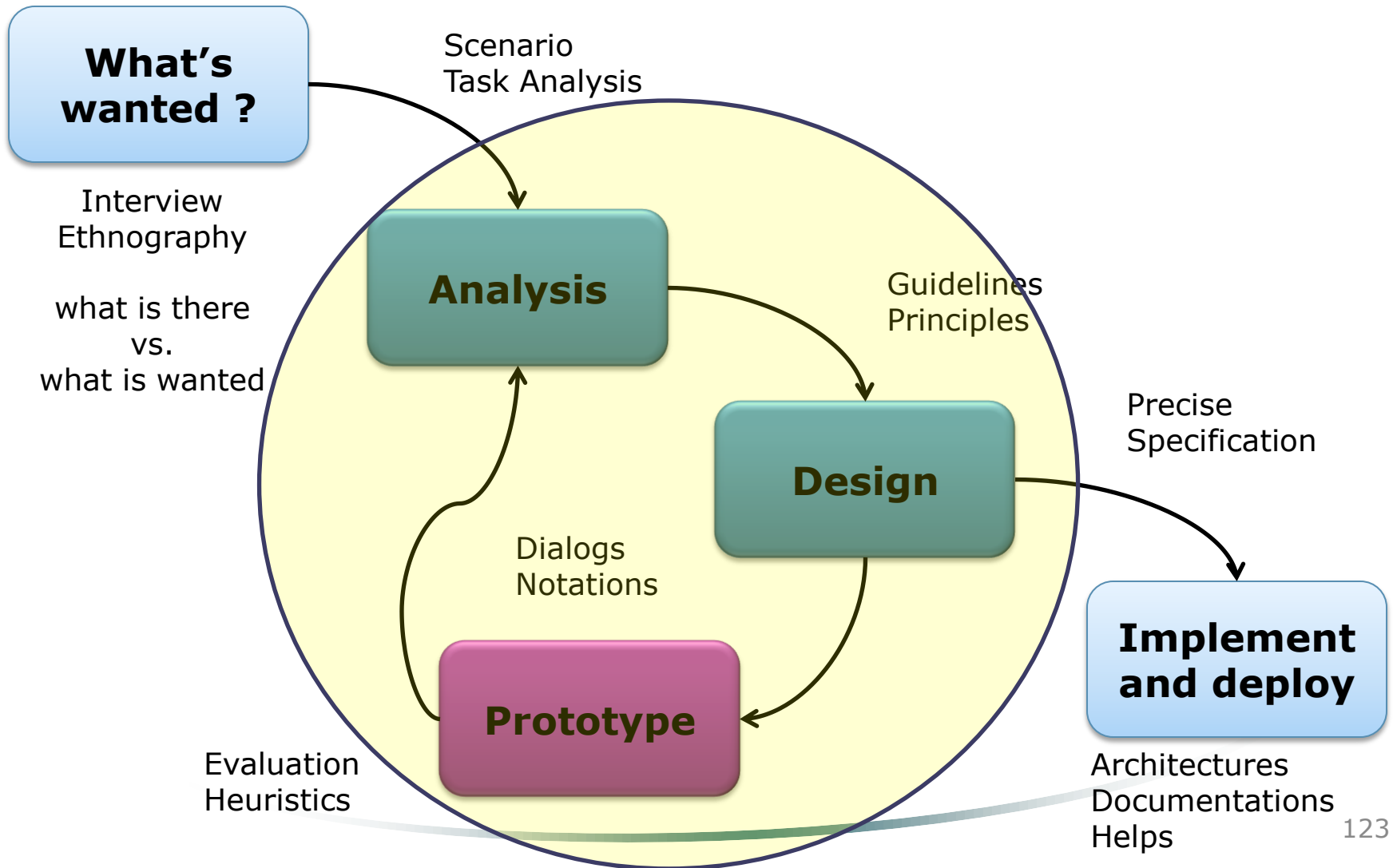
- Dùng để mô hình hóa các tương tác phức tạp
- Sự kiện: xảy ra tại thời điểm cụ thể, có thể làm thay đổi trạng thái
 - chuông reo, nhấn phím
- Trạng thái: giá trị nhất định trong một khoảng thời gian, sự thay đổi trạng thái có thể là sự kiện có ý nghĩa
 - hiện thị trên màn hình, vị trí chuột
- Cho phép phân tích người dùng và hệ thống dưới góc độ giống nhau



CHƯƠNG III: THIẾT KẾ TƯƠNG TÁC NGƯỜI DÙNG MÁY TÍNH

- 
- I. Giới thiệu chung
 - II. Mô hình thoại
 - III. Mô hình tương tác
 - IV. Thiết kế lặp
 - V. Mẫu thử

Thiết kế lặp trong quy trình thiết kế tương tác





Tại sao cần thiết kế lặp ?

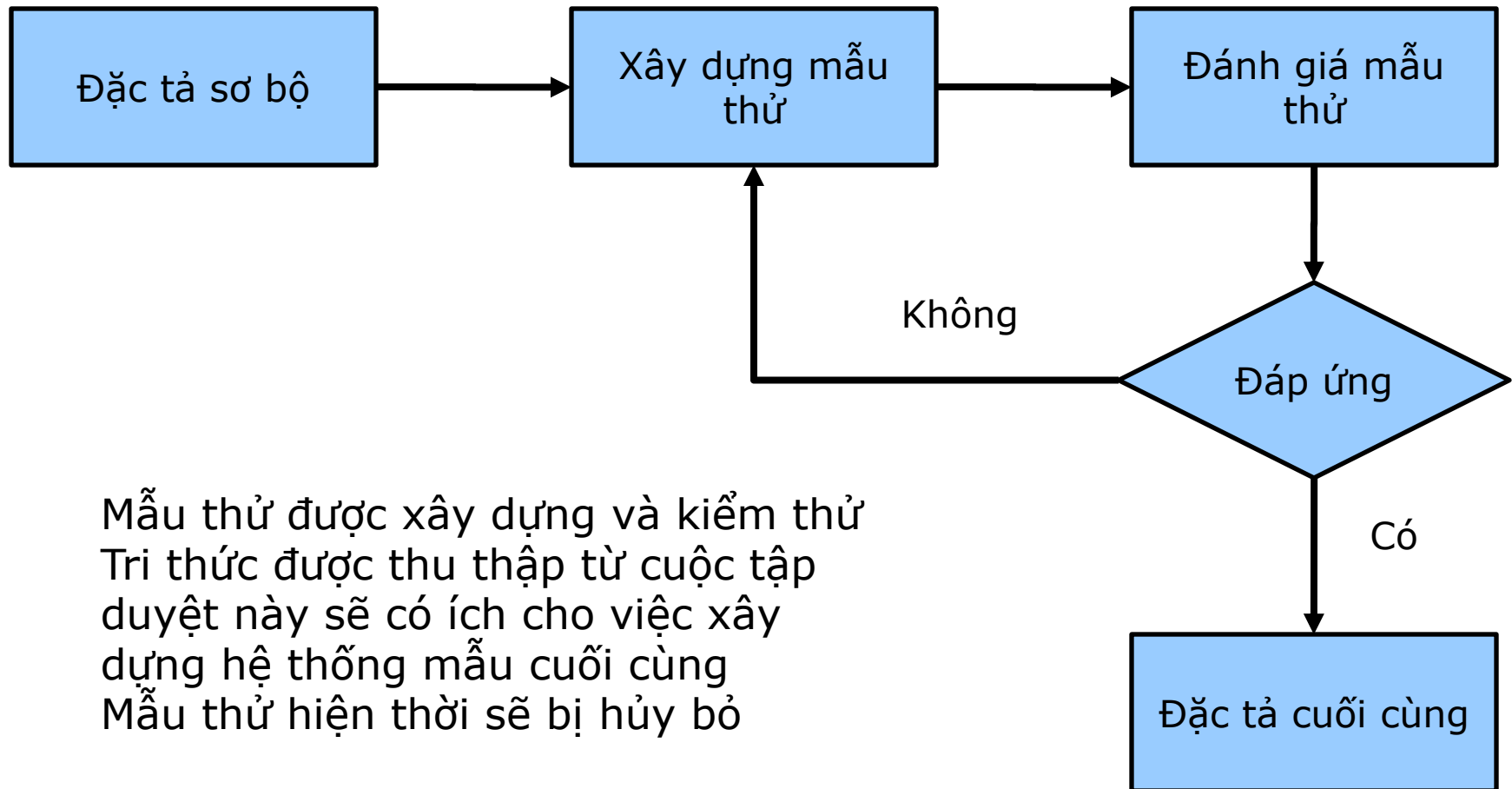
- Đặc tả yêu cầu người dùng thường hiếm khi đầy đủ
- Quá trình đặc tả yêu cầu thường diễn ra ở giai đoạn đầu nên phải được hiệu chỉnh trong lúc thiết kế
- Để đảm bảo các đặc trưng của thiết kế phải
 - Xây dựng
 - Kiểm thử
 - Đánh giá } Với người dùng thực
- Thiết kế cần phải được hiệu chỉnh để sửa các lỗi phát hiện được trong lúc kiểm thử



V. Mẫu thử

- Mẫu thử: Là sự bắt chước hay mô phỏng một số chức năng đặc trưng chứ không phải của một hệ thống đầy đủ (hệ thống có thể chưa tồn tại)
- Có 3 kỹ thuật mẫu thử:
 - Tung ra (Throw away)
 - Gia tăng (Incremental)
 - Tiến hóa (Evolutionary)

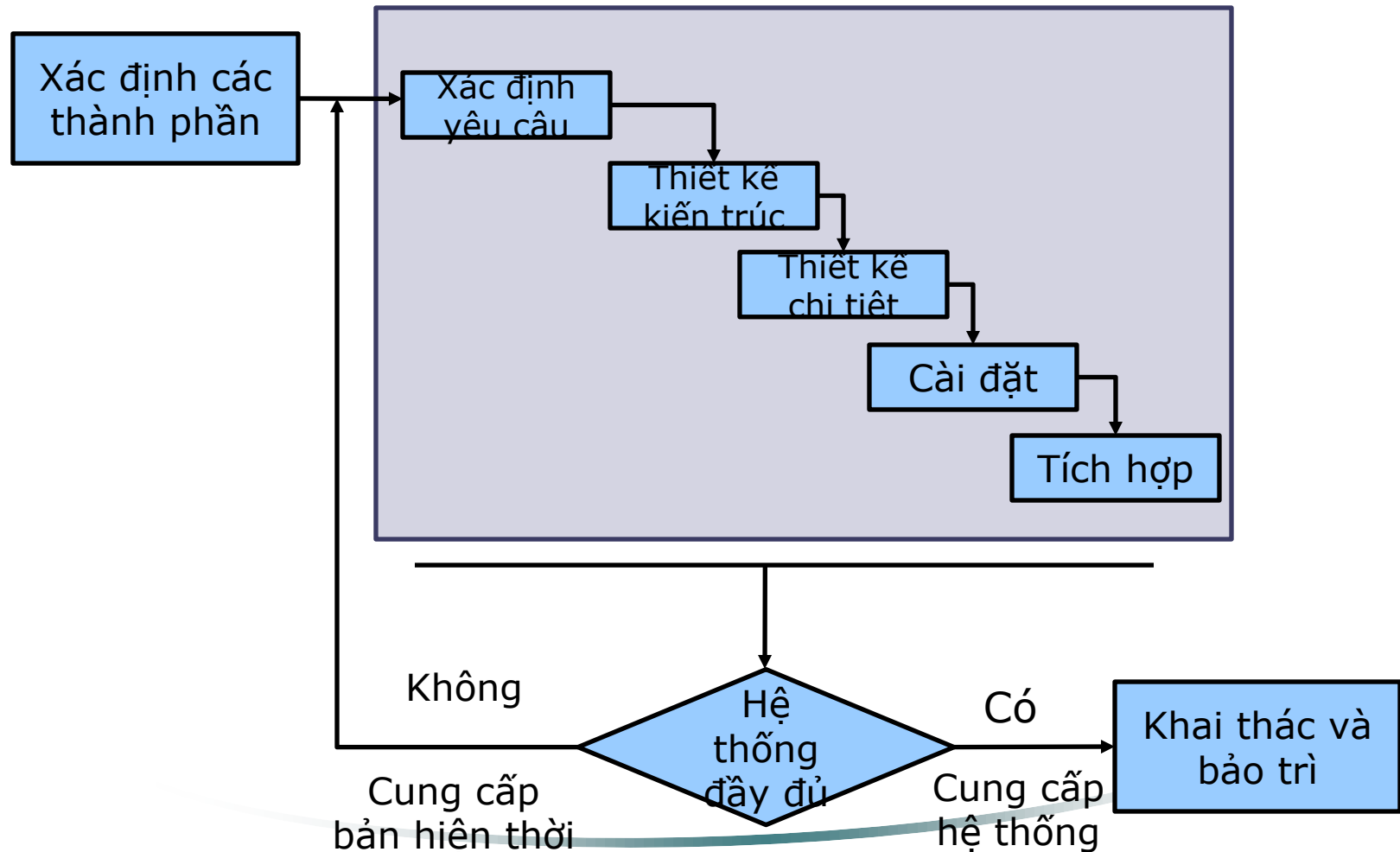
1. Mẫu thử tung ra



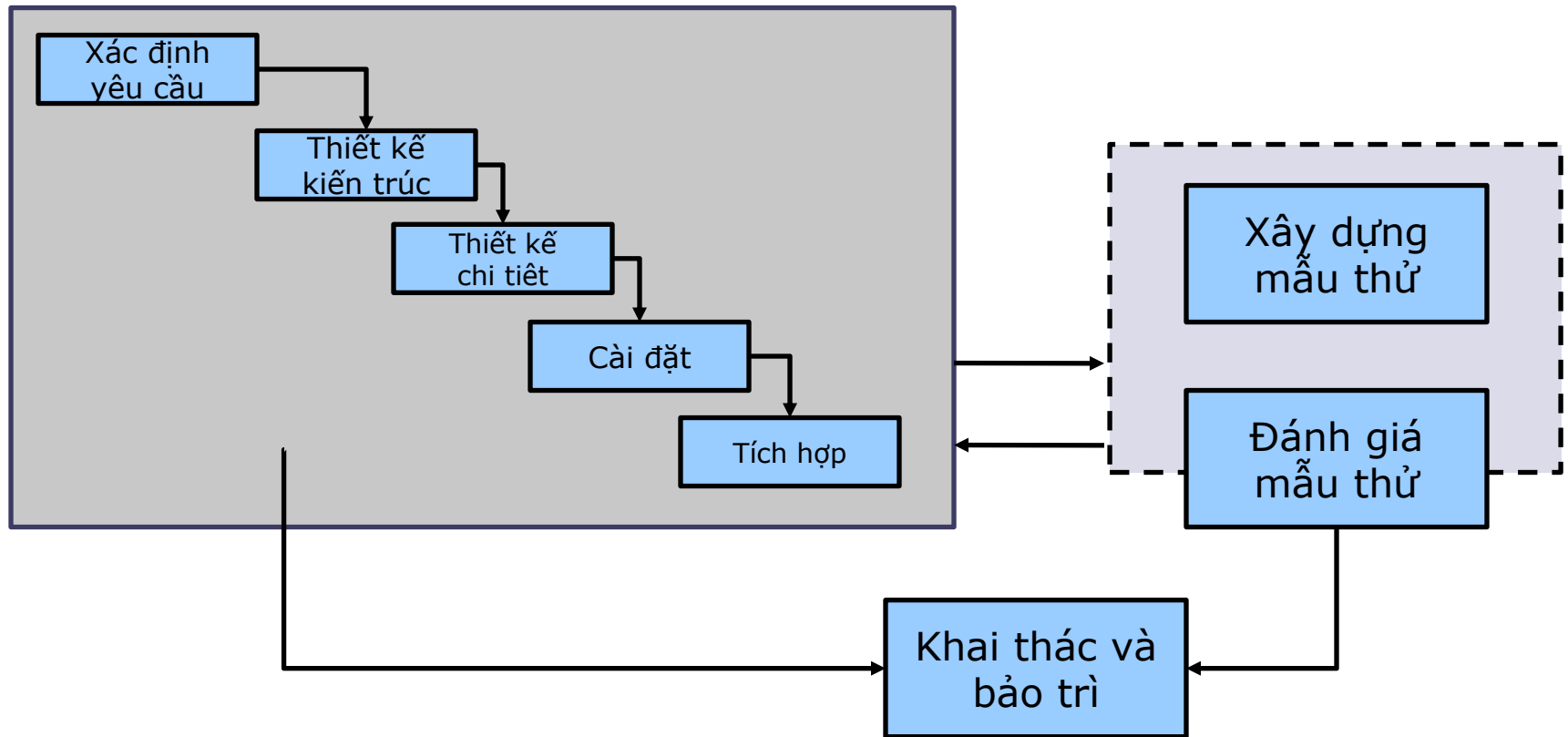
Mẫu thử được xây dựng và kiểm thử
Tri thức được thu thập từ cuộc tập
duyệt này sẽ có ích cho việc xây
dựng hệ thống mẫu cuối cùng
Mẫu thử hiện thời sẽ bị hủy bỏ

2. Mẫu thử gia tăng

- Sản phẩm cuối cùng là một chuỗi các thành phần riêng biệt, mỗi thành phần được thiết kế hoàn thiện ở một thời điểm



3. Mẫu thử kiểu tiến hóa



Mẫu thử không bị hủy bỏ mà được dùng như cơ sở cho lần lặp tiếp theo. Hệ thống hiện thời được xem như là sự tiến hóa phiên bản rất thô ban đầu để đến sản phẩm cuối cùng.



4. Ưu điểm của các mẫu thử

- Làm mịn đặc tả
- Làm mịn thiết kế
- Cho phép so sánh đánh giá các thiết kế
- Chứng minh cho một số ý tưởng
- Có thể áp dụng cho các đối tượng: người dùng, nhà thiết kế, ...



5. Hạn chế của các mẫu thử

- **Tốn thời gian:** Xây dựng mẫu thử cũng cần có thời gian. Mẫu thử kiểu tung ra sẽ lãng phí về thời gian và tiền bạc.
- **Kế hoạch:** Người quản lý dự án không có kinh nghiệm để lập một kế hoạch hợp lý và chi phí cho quá trình thiết kế.
- **Đặc trưng phi chức năng:** các đặc trưng phi chức năng như tính an toàn, độ tin cậy cần phải được đảm bảo trong quá trình thiết kế.




Tổng kết bài học

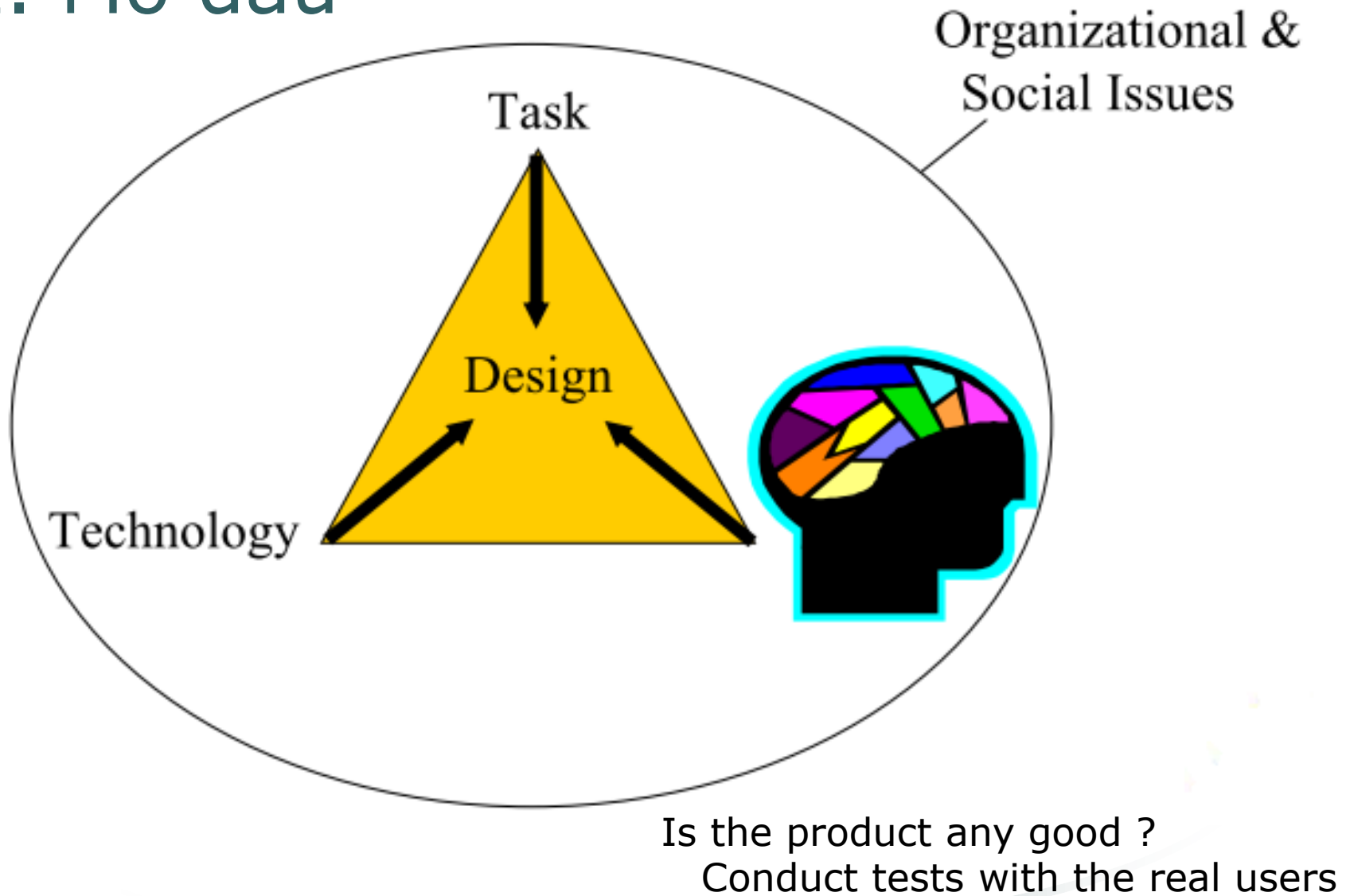
- SV nắm được cụ thể 3 pha đầu của quy trình
 - Đặc tả yêu cầu
 - Phân tích nhiệm vụ
 - Thiết kế tương tác
- Sinh viên lưu ý trong phần đặc tả yêu cầu: có sự có mặt của đặc tả về tính dùng được



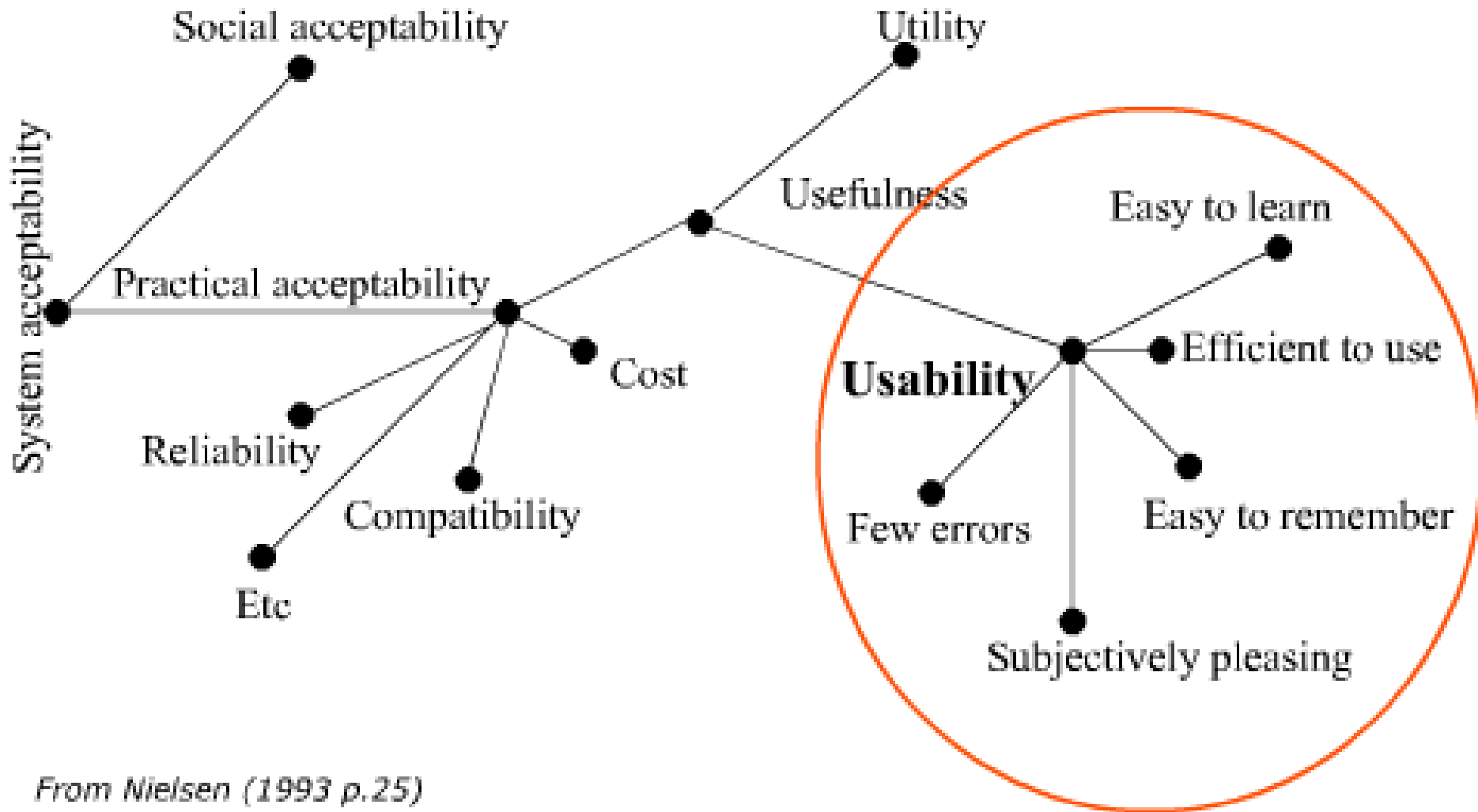
CHƯƠNG V: KIỂM THỬ, ĐÁNH GIÁ VÀ QUẢN LÝ HỆ TƯƠNG TÁC

- 
- I. Mở đầu
 - I. Khái niệm
 - II. Các mô thức đánh giá
 - II. Kế hoạch kiểm thử
 - III. Đánh giá hệ thống
 - IV. Quản lý tương tác

I. Mở đầu



Làm sao biết hệ thống được chấp nhận?



From Nielsen (1993 p.25)

Kiểm thử các chức năng hay đánh giá tính dùng được của hệ thống

- Kiểm thử (Testing):
 - Kiểm thử chức năng của hệ thống
 - Xác định và sửa lỗi mã nguồn, lỗi logic, v.v.
 - Cực kỳ quan trọng
- Đánh giá (Evaluation): kiểm thử tính dùng được của hệ thống để biết người dùng có đạt được mục đích theo các tiêu chí sau hay không:
 - Hiệu quả
 - Năng suất
 - Hiệu dụng
 - An toàn
 - Thỏa mãn



1. Khái niệm

- Đánh giá là thu thập dữ liệu kiểm tra về tính dùng được của thiết kế
- Đánh giá không phải là một giai đoạn trong thiết kế
- Đánh giá là nhiệm vụ trung tâm của vòng đời thiết kế và diễn ra trong suốt vòng đời của HCI
- Do không thể thực hiện các kiểm thử thực nghiệm trong suốt quá trình thiết kế vì thế ta phải dùng các kỹ thuật phân tích / phi hình thức

2. Các mô thức đánh giá

Quick and dirty: tranh luận không chính thức với người dùng vào bất cứ lúc nào, dùng các mẫu thử. Lấy phản hồi từ người dùng hoặc người tư vấn để xác nhận rằng ý tưởng của nhóm phát triển vẫn trùng với nhu cầu của người dùng và được người dùng ưa thích.

Field studies: đến chỗ người dùng và phỏng vấn, hay quan sát người dùng sử dụng giao diện

Người dùng thường làm gì
Công nghệ ảnh hưởng đến họ như thế nào

Usability testing: quan sát người dùng và ghi lại hiệu suất của các đối tượng người dùng điển hình khi thực hiện các nhiệm vụ điển hình theo các cấu hình cài đặt sẵn
Giải thích tại sao người dùng lại làm những hành động đó (tính hiệu suất thời gian, xác định lỗi)
Khuyến khích người dùng cho ý kiến (phỏng vấn, bảng câu hỏi)

Predictive: không cần sự có mặt của người dùng (tiến hành bên phía phát triển)

Sử dụng hiểu biết của các chuyên gia về các đối tượng người dùng điển hình để dự đoán các vấn đề về tính dùng được (heuristic evaluation). Có thể sử dụng các cách tiếp cận thuần túy lý thuyết.

Các mô thức đánh giá

Quick and
dirty

Field
studies

Experiences

**User
testings**

Quan sát người dùng

Hỏi ý kiến người dùng

Hỏi ý kiến chuyên gia

Kiểm thử hiệu năng người dùng

Mô hình hóa hiệu năng thực hiện
các nhiệm vụ của người dùng


Usability testing

Predictive





II. Kế hoạch kiểm thử

- Mục đích kiểm thử
 - Đặt vấn đề / mục tiêu kiểm thử
 - Hồ sơ các đối tượng tham gia (các tiêu chí cho phép tham gia hay không vào kế hoạch kiểm thử)
 - Phương pháp / kỹ thuật kiểm thử
 - Danh sách các công việc cần làm
 - Môi trường và phương tiện kiểm thử
 - Vai trò của các đối tượng tham gia thực nghiệm (monitor, coach etc.)
 - Các biện pháp đánh giá được áp dụng (qualitative vs. quantitative, subjective vs. objective)
 - Nội dung và cách trình bày báo cáo cho các đối tượng khác nhau
- 



III. Đánh giá hệ thống

- Đánh giá đảm bảo 3 nhiệm vụ chính
 - Khẳng định tính mở rộng của các chức năng
 - Hệ thống phải có khả năng đáp ứng các nhiệm vụ đặt ra một cách dễ dàng
 - Đánh giá khả năng sử dụng của hệ thống so với nhu cầu của người dùng
 - Khẳng định tính hiệu quả trong giao tiếp đối với người dùng
 - Đo đếm sự ảnh hưởng của hệ thống đối với người dùng
 - Tính dễ học, dễ dùng, dễ nhớ, v.v.
 - Xác định một số vấn đề đặc biệt nảy sinh trong quá trình sử dụng



Phân loại

- Phân chia theo điều kiện môi trường nơi tiến hành đánh giá
 - Đánh giá trong phòng thí nghiệm
 - Đánh giá thực địa
- Phân chia theo thời gian, vòng đời của quá trình thiết kế
 - Đánh giá thiết kế (thử nghiệm giao diện)
 - Đánh giá cài đặt



1. Đánh giá trong phòng thí nghiệm

- Diễn ra trong phòng thí nghiệm
- Dùng trong quá trình thiết kế
- Người đánh giá muốn thực hiện một số khẳng định mà không cần đến người dùng
- Người dùng cũng có thể tham gia vào quá trình đánh giá nếu muốn



Đánh giá trong phòng thí nghiệm

- Điều kiện khách quan
- Thiếu ngữ cảnh, điều kiện không tự nhiên, không có thật
- Giao tiếp không tự nhiên
- Cần thiết khi môi trường thực địa không cho phép (trạm vũ trụ, nơi nguy hiểm)
- Muốn phát hiện một số vấn đề, một số thủ tục ít dùng hoặc so sánh các thiết kế khác nhau thì đây là cách thức tốt nhất

2. Đánh giá tại chỗ

- Được tiến hành với sự tham gia của người dùng
- Diễn ra trong giai đoạn thiết kế hay cài đặt
- Được diễn ra trong môi trường người dùng nhằm đánh giá hệ thống trong hoạt động và trạng thái của người dùng
- Có nhiều yếu tố bị ảnh hưởng: tiếng ồn, chuyển động, người qua lại, v.v. gây mất tập trung
- Bản chất tự nhiên, cho phép quan sát được sự tương tác của hệ thống và người dùng, cái mà ta không quan sát được ở trong PTN
- Do có sự hiện diện của người đánh giá mà người dùng có thể mất tập trung, không tự nhiên



3. Đánh giá thiết kế

- Việc đánh giá thực hiện ngay trong quá trình thiết kế
- Những đánh giá đầu tiên về hệ thống nên được thực hiện trước khi hệ thống được cài đặt
- Nếu trong giai đoạn này, nhờ đánh giá lỗi sẽ được phát hiện sớm tránh những ảnh hưởng đáng tiếc, giảm chi phí chỉnh sửa



4. Đánh giá cài đặt

- Đánh giá có sự hiện diện của người dùng và hệ thống đã được cài đặt
- Có thể sử dụng các kỹ thuật
 - Đánh giá thực nghiệm
 - Kỹ thuật quan sát
 - Kỹ thuật hỏi đáp

5. Phương pháp đánh giá heuristic

- Kiểm tra xem hệ tương tác có tuân thủ theo các nguyên lý, luật thiết kế hay không.
- Các khía cạnh cần kiểm tra theo Nielsen:
 - Khả năng nhìn thấy được các trạng thái của hệ thống
 - Sự tương đồng giữa hệ thống và thế giới thực
 - Sự kiểm soát người dùng và sự tự do
 - Tính nhất quán và các tiêu chuẩn
 - Phòng ngừa lỗi
 - Giúp người dùng nhận biết, chẩn đoán và khôi phục khi xảy ra lỗi
 - Nhận biết thay vì nhớ lại
 - Tính linh hoạt và hiệu quả sử dụng
 - Tính thẩm mỹ và tính tối giản
 - Trợ giúp và tài liệu



Quy trình đánh giá

- Lập kế hoạch: mô tả các công việc chuyên gia đánh giá cần làm
- Đánh giá:
 - các chuyên gia thực hiện độc lập, sử dụng các heuristic để xem xét giao diện, đặc tả hoặc phác thảo màn hình
 - Lần 1: xem xét luồng tương tác và phạm vi của hệ thống
 - Lần 2: xem xét các phần tử giao diện cụ thể trong ngữ cảnh tổng thể để nhận dạng các vấn đề tiềm tàng về tính tiện dụng.
 - Khi phát hiện vấn đề phải ghi lại càng chi tiết càng tốt.
- Tổng kết: thảo luận các vấn đề phát hiện ra, xác định thứ tự ưu tiên và đề xuất giải pháp

6. Phương pháp đánh giá đi qua từng nhiệm vụ (Cognitive Walkthrough)

- Do các thành viên của nhóm thiết kế thực hiện, với giả thiết là người dùng khám phá giao diện hệ thống để học cách sử dụng hệ thống.
- Bước 1: ước lượng hiểu biết, kỹ năng và kinh nghiệm của nhóm người dùng mục tiêu.
- Bước 2:
 - Nhận diện các nhiệm vụ đặc trưng thường được người dùng thực hiện trong lần đầu tiên sử dụng hệ thống
 - Xây dựng tập câu hỏi áp dụng cho các hành động thành phần mà người dùng cần thực hiện thành công để hoàn thành các nhiệm vụ.
- Bước 3:
 - Thực hiện lần lượt từng hành động

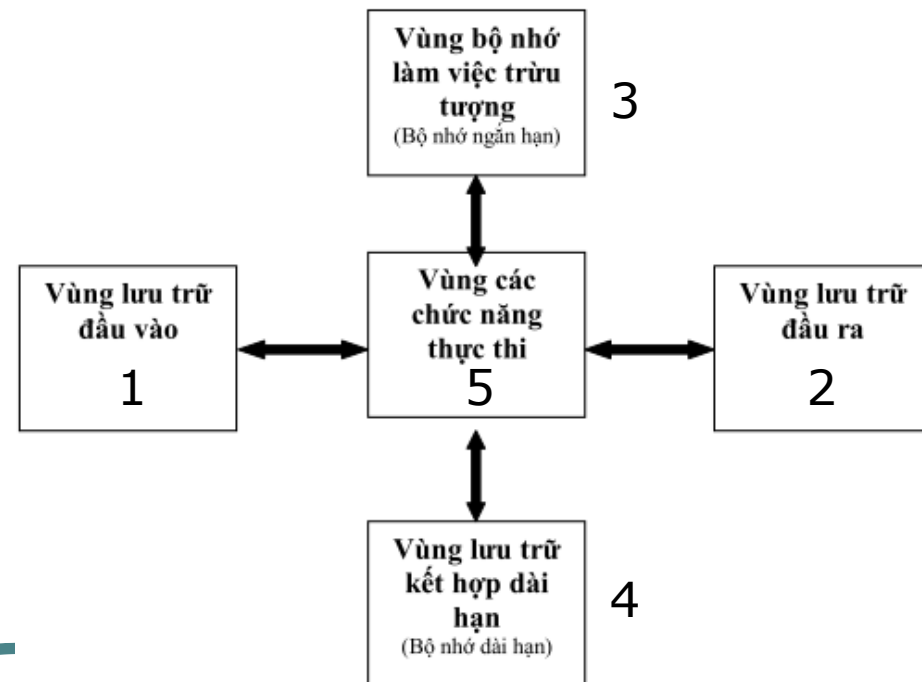
Phương pháp đánh giá đi qua từng nhiệm vụ (Cognitive Walkthrough)

- Bước 4:
 - Đánh giá từng hành động theo các tiêu chí
 - người dùng có phải cố gắng để đạt được kết quả đúng hay không?
 - người dùng có biết sự tồn tại của các hành động đúng hay không?
 - người dùng có biết được tính đúng đắn của hành động hay không?
 - Nếu hành động đúng được thực hiện, người dùng có thể hiểu được các phản hồi từ hệ thống và nhận biết là quá trình đó đang tiến đến kết quả mong muốn hay không?
 - Ghi lại các vấn đề để có cơ sở lựa chọn thiết kế khác.

7. Phương pháp đánh giá sử dụng mô hình Simplex One

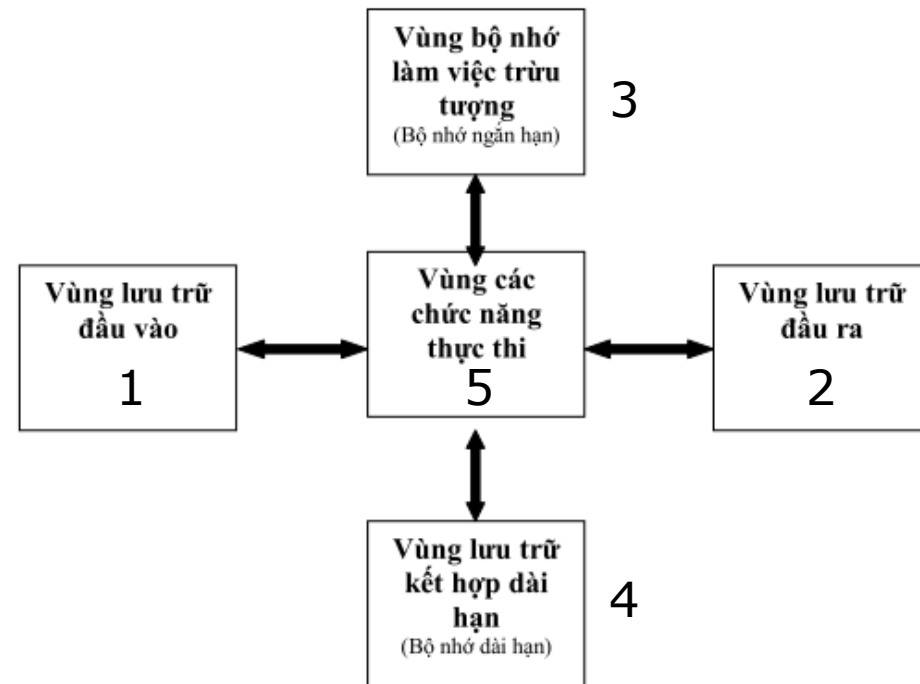
1. Cảm giác (đầu vào): Khả năng tiếp nhận các thông tin mới từ các giác quan để phân tích và lưu giữ các thông tin đó và liên hệ với các thông tin hiện có
2. Đáp ứng (đầu ra): Khả năng lựa chọn, tổ chức, định thời và thực hiện các đáp ứng thích hợp
3. Bộ nhớ làm việc ngắn hạn: Thu nhận, lưu giữ và xử lý các ký ức cần cho các hành động của nhiệm vụ.
 - Bị giới hạn về dung lượng và thời gian.
4. Bộ nhớ dài hạn: Lưu trữ đồng thời các sự kiện chính và các biểu tượng tương ứng.
 - Ít bị giới hạn hơn về dung lượng và thời gian
 - Bị giới hạn về chất lượng thông tin đầu vào và khả năng triệu gọi thông tin ra.

5. Các chức năng thực thi:
 - Chuyển thông tin giữa các vùng
 - Tổ chức các chuỗi hoạt động trao đổi thông tin
 - Điều độ chức năng của các vùng khác nhau
 - Tổ chức và giám sát các yêu cầu nhiệm vụ



Đánh giá sử dụng mô hình Simplex One

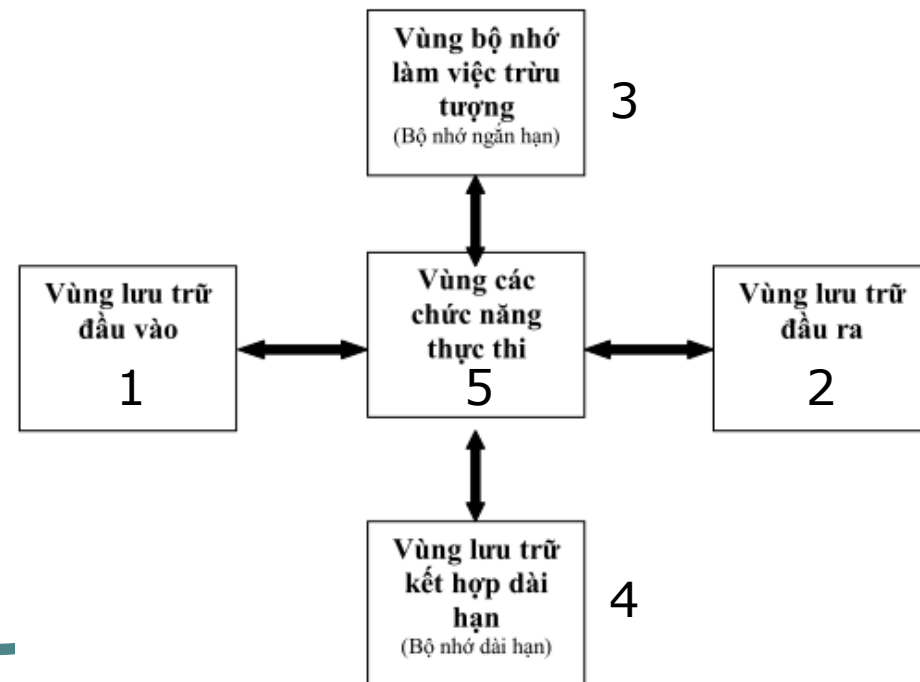
- Các khía cạnh cần đánh giá:
 1. Thiết kế đầu vào hợp lý cho người dùng
 2. Hỗ trợ các đáp ứng của người dùng và cho phép chúng được thực hiện dễ dàng
 3. Lượng thông tin lưu giữ không nhiều
 4. Cung cấp thông tin thích hợp cho việc lưu trữ dài hạn, hiệu quả: có mẫu phù hợp tại những thời điểm phù hợp để người dùng dễ học hỏi
 5. Hỗ trợ vùng các chức năng thực thi: đảm bảo rằng các nhiệm vụ do hệ thống yêu cầu không quá phức tạp để có thể làm chủ và duy trì



Đánh giá sử dụng mô hình Simplex One


- Các khía cạnh cần đánh giá:
 1. Thiết kế đầu vào hợp lý cho người dùng
 2. Hỗ trợ các đáp ứng của người dùng và cho phép chúng được thực hiện dễ dàng
 3. Lượng thông tin lưu giữ không nhiều
 4. Cung cấp thông tin thích hợp cho việc lưu trữ dài hạn, hiệu quả: có mẫu phù hợp tại những thời điểm phù hợp để người dùng dễ học hỏi
 5. Hỗ trợ vùng các chức năng thực thi: đảm bảo rằng các nhiệm vụ do hệ thống yêu cầu không quá phức tạp để có thể làm chủ và duy trì

- Các loại đánh giá
 - Đánh giá người dùng
 - Đánh giá thiết kế





9. Lựa chọn phương pháp đánh giá

- Đánh giá giai đoạn nào trong quá trình phát triển hệ thống: Đánh giá thiết kế hay đánh giá cài đặt hệ thống
 - Kiểu đánh giá: tại phòng thí nghiệm hay tại môi trường làm việc thực
 - Mục tiêu đánh giá là đánh giá mô hình người dùng (khách quan) hay đánh giá các lựa chọn thiết kế (chủ quan)
 - Biện pháp: định tính hay định lượng
 - Mức độ thông tin: cao hay thấp
 - Tài nguyên sử dụng: thời gian, số người tham gia, thiết bị, khả năng chuyên môn
- 



V. Quản lý hệ tương tác

- Khai thác: giúp người dùng làm việc với hệ thống được xây dựng
- Bảo trì:
 - Các nhiệm vụ nhằm giữ cho hệ thống hoạt động liên tục, bất kỳ các thao tác nào nhằm cải thiện hệ thống, thay đổi, thêm vào do người dùng yêu cầu
 - Các lỗi phát hiện được sẽ được lưu lại để cải thiện hệ thống trong tương lai
 - Các phản hồi từ phía người dùng về các chức năng và phi chức năng được ghi nhận