BÀI 6: PHÂN TÍCH NHIỆM VỤ

I. **Giới thiệu**
II. GOMS
III. HTA
IV. Mô hình thoại
V. Mô hình tương tác

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)

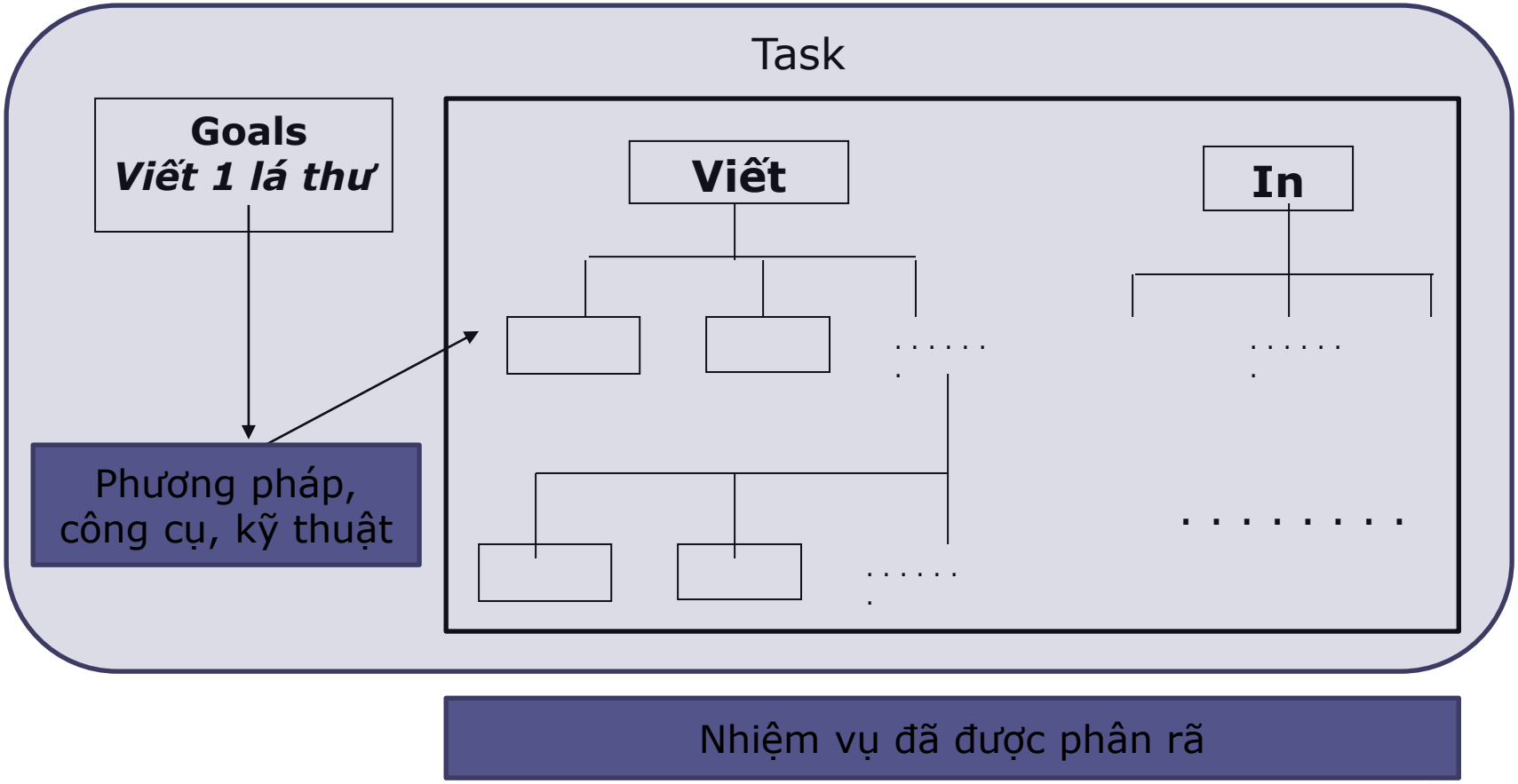
2. Ví dụ về phân tích nhiệm vụ "Hút bụi"

- 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

Các cách tiếp cận phân tích nhiệm vụ

Phân rã nhiệm vụ	Describe the actions people do Structure them within task / subtask hierarchy Describe order of subtasks	Action-based models: HTA
Các kỹ thuật dựa trên tri thức	Focus on: Objects used in task Actions performed Taxonomies represent levels of abstraction	Cognitive models: GOMS
Phân tích dựa vào mô hình quan hệ thực thể	Focus on objects, actions and their relationships Emphasize domain understanding not implementation	Object-oriented analysis + non-computer entities
Phân tích tổng quát	Observe Collect unstructured lists of words and actions Organize using notation or diagrams	Linguistic models: TAG, BNF

I. Khái niệm

II. Phân tích nhiệm vụ dựa trên nhận thức của người dùng - GOMS

Giới thiệu mô hình GOMS

- 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 hoach hành động để hoàn thành nhiệm vụ mà là kết quả của kinh nghiệm được tích luỹ.
- 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 ↑, chừng nào con trỏ chưa đúng vị trí nhấn ← (→)
- 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

I. Khái niệm
II. Phân tích nhiệm vụ dựa trên nhận thức của người dùng - GOMS
III. Mô hình ngôn ngữ

Giới thiệu

- 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→

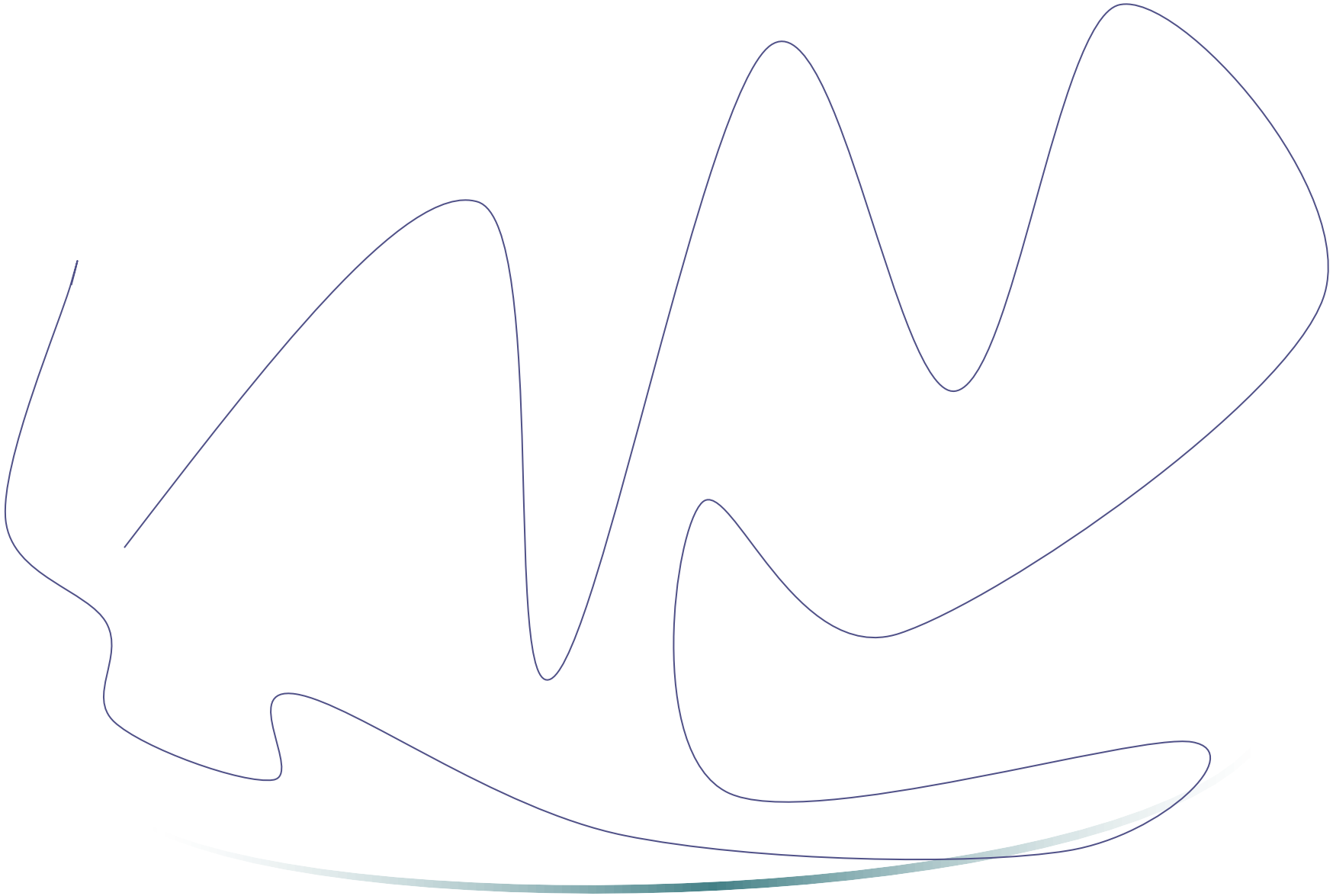
→ BNF được sử dụng khá rộng rãi để đặc tả cú pháp của các ngôn ngữ lập trình

→ Chỉ biểu diễn hành động ND mà không đề cập đến cảm nhận của ND vè 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.

III. PHÂN TÍCH CÔNG VIỆC (HTA)

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. Xoá 1 tệp
  2. Xoá 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 xoá 1 tệp
  - Step 1 Chọn lệnh ERASE từ tệp lệnh
  - Step 2 Nghĩ tên thư mục và tên tệp
  - Step 3 Nhập lệnh và thực hiện lệnh
  - Step 4 Kết thúc
- 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 xoá 1 thư mục

- Phương pháp thực hiện mục đích xoá 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

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 xoá TM
  - Step 3 Kết thúc
- Phương pháp thực hiện mục đích sao 1 thư mục
  - Step 1 TạoTM 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 Gắp tệp bỏ vào thùng rác
  - Step 2 Kết thúc
- Phương pháp thực hiện mục đích chuyển 1 tệp
  - Step 1 Gắp tệp bỏ vào nơi đến
  - Step 2 Kết thúc
- Phương pháp thực hiện mục đích xoá 1 TM
  - Step 1 Gắp TM bỏ vào thùng rác
  - Step 2 Kết thúc
- Phương pháp thực hiện mục đích chuyển 1 TM
  - Step 1 Gắp TM bỏ vào nơi đến
  - Step 2 Kết thúc

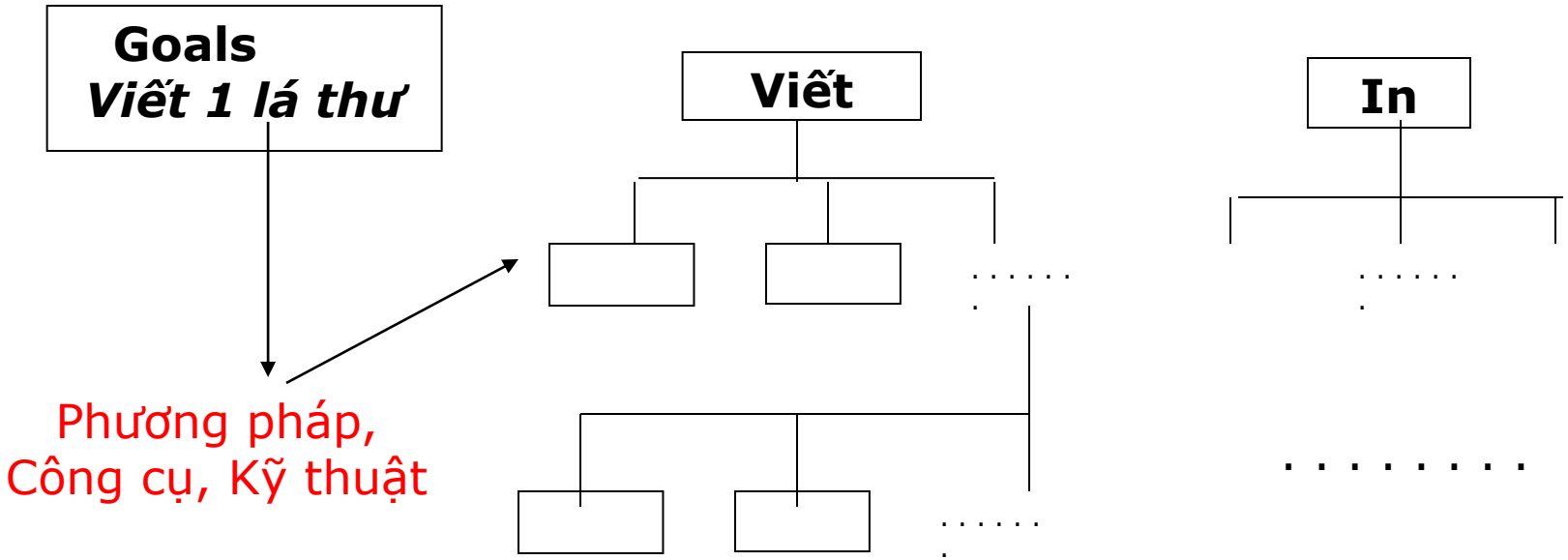
Macintosh
Generalize file manipulation methods

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

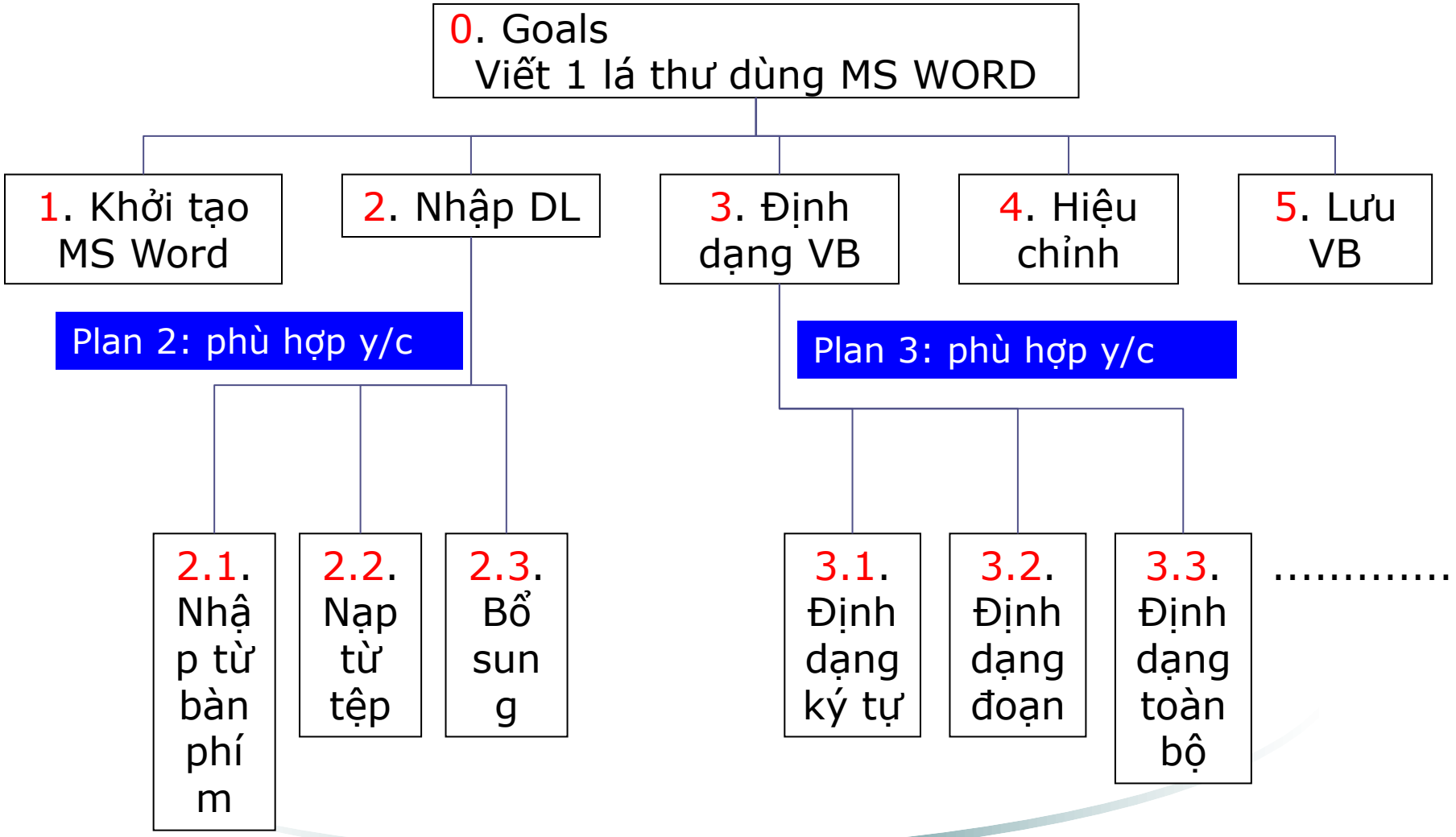
2. Sơ đồ quan hệ phân cấp các nhiệm vụ (HTA)

- 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

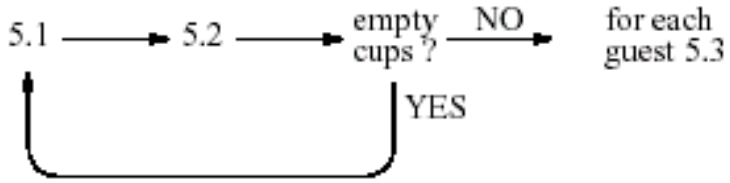
- Luật kết thúc: khi nào kết thúc công việc?

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

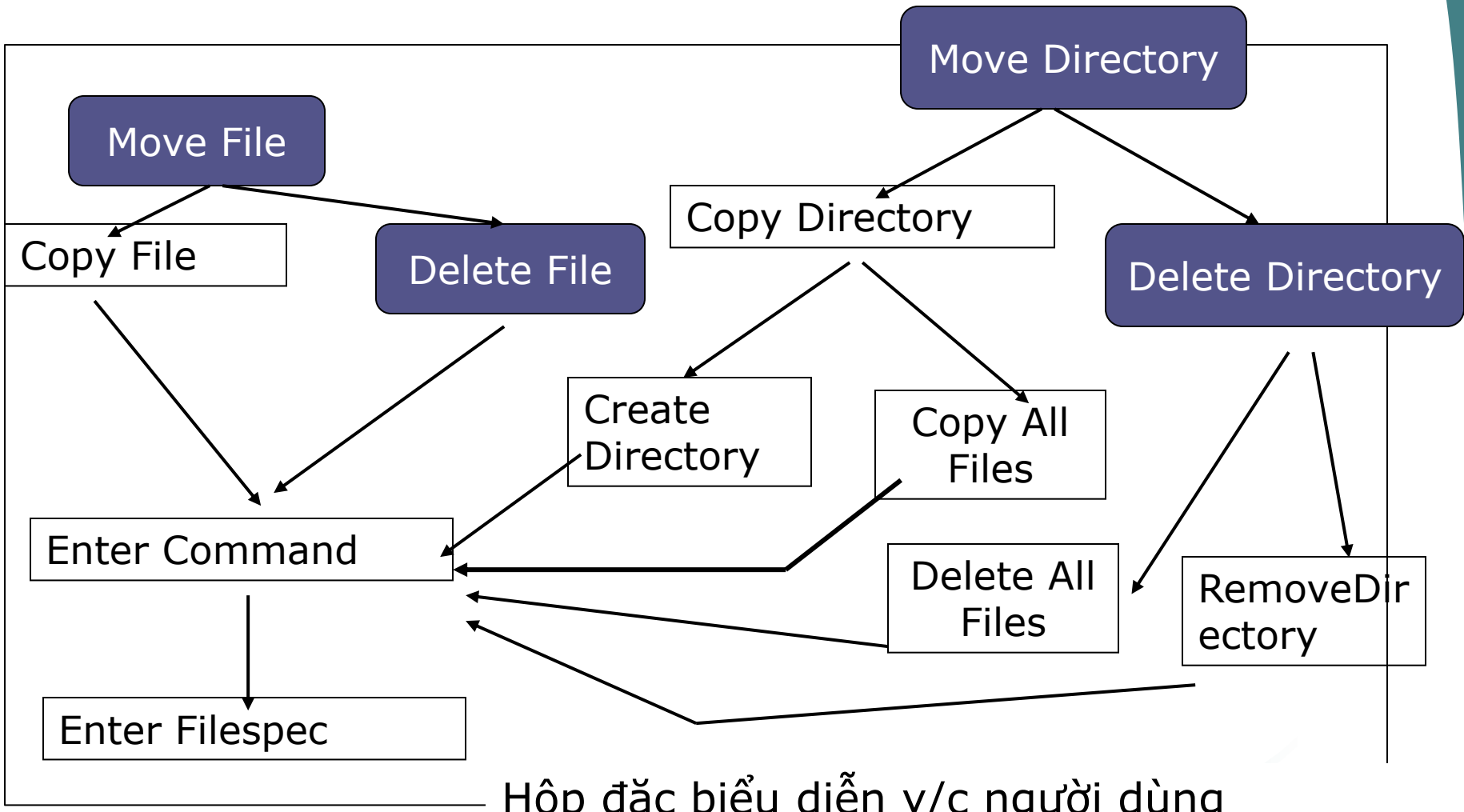
Các kiểu kế hoạch (plan)

- 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

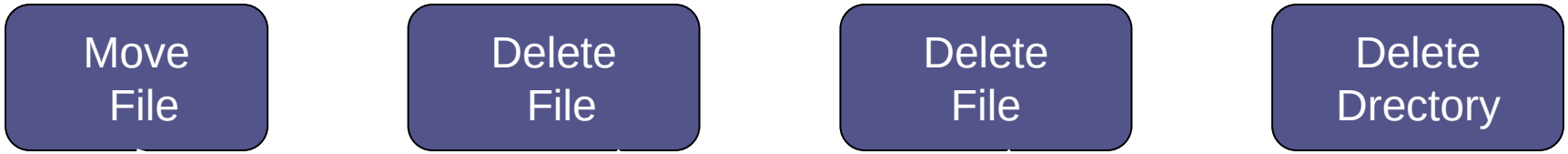
Cấu trúc mục đích của PC-DOS



Hộp đặc biểu diễn y/c người dùng

Hộp rỗng là nhiệm vụ con

Cấu trúc mục đích của Macintosh

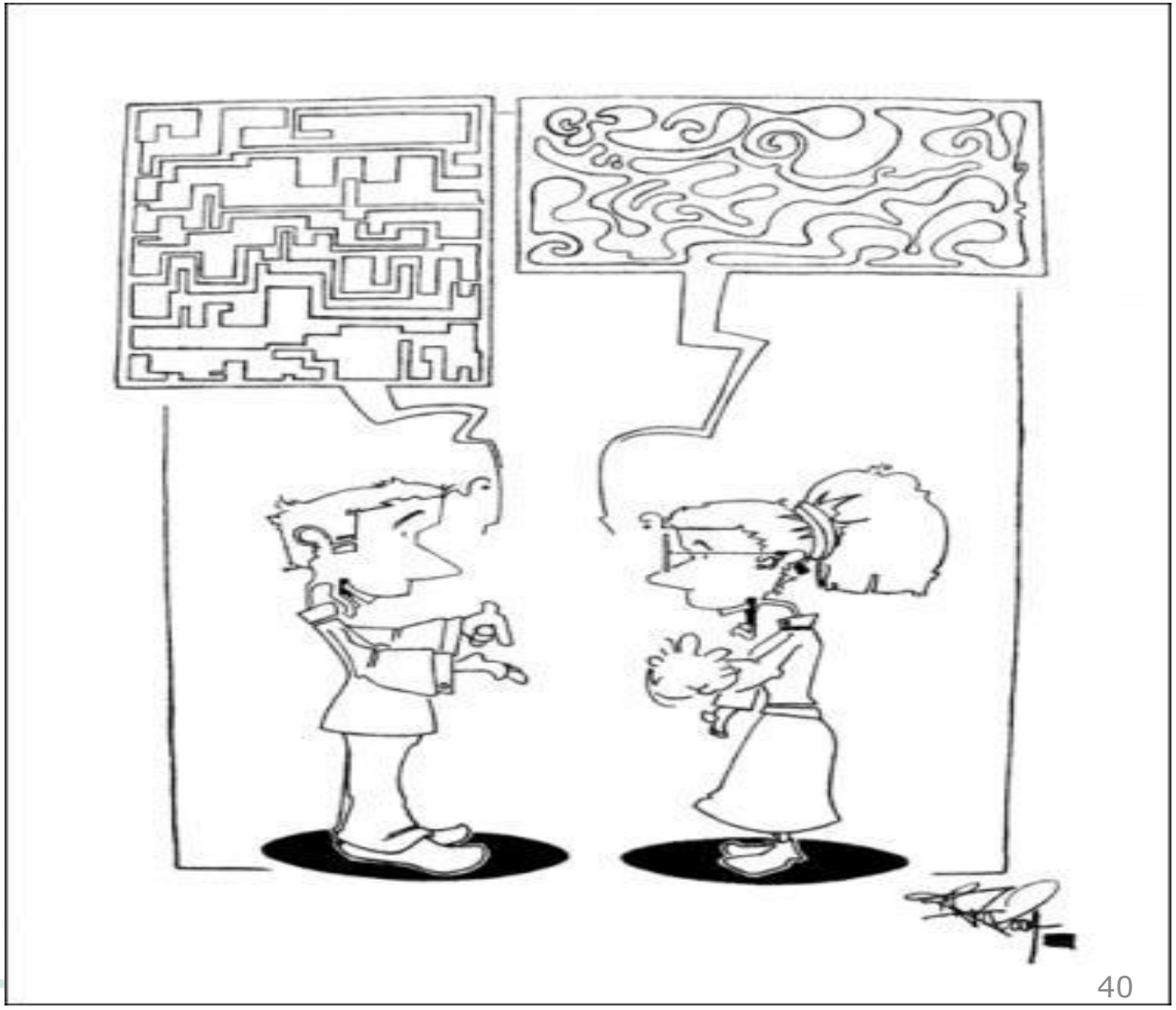


Drag Item to Destination

IV. 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

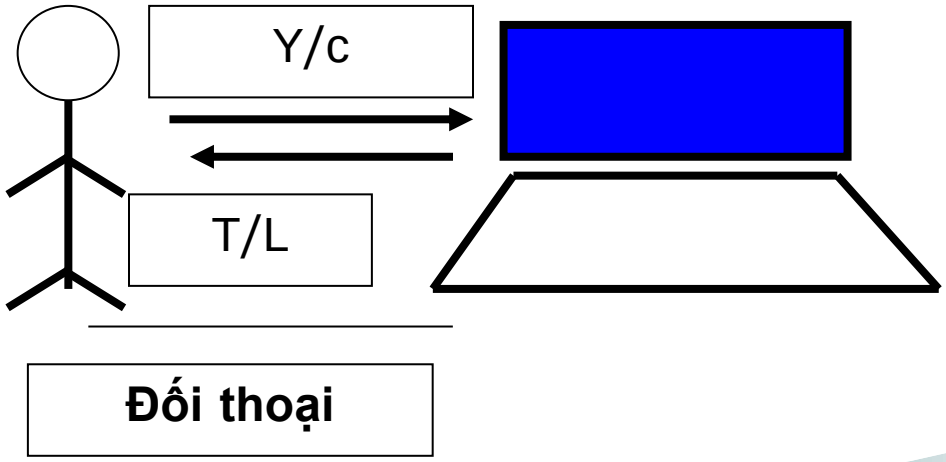
1. Khái niệm



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 HTT
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: dáng vẻ 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 chuyên trạng thái phân cấp (HSTN)

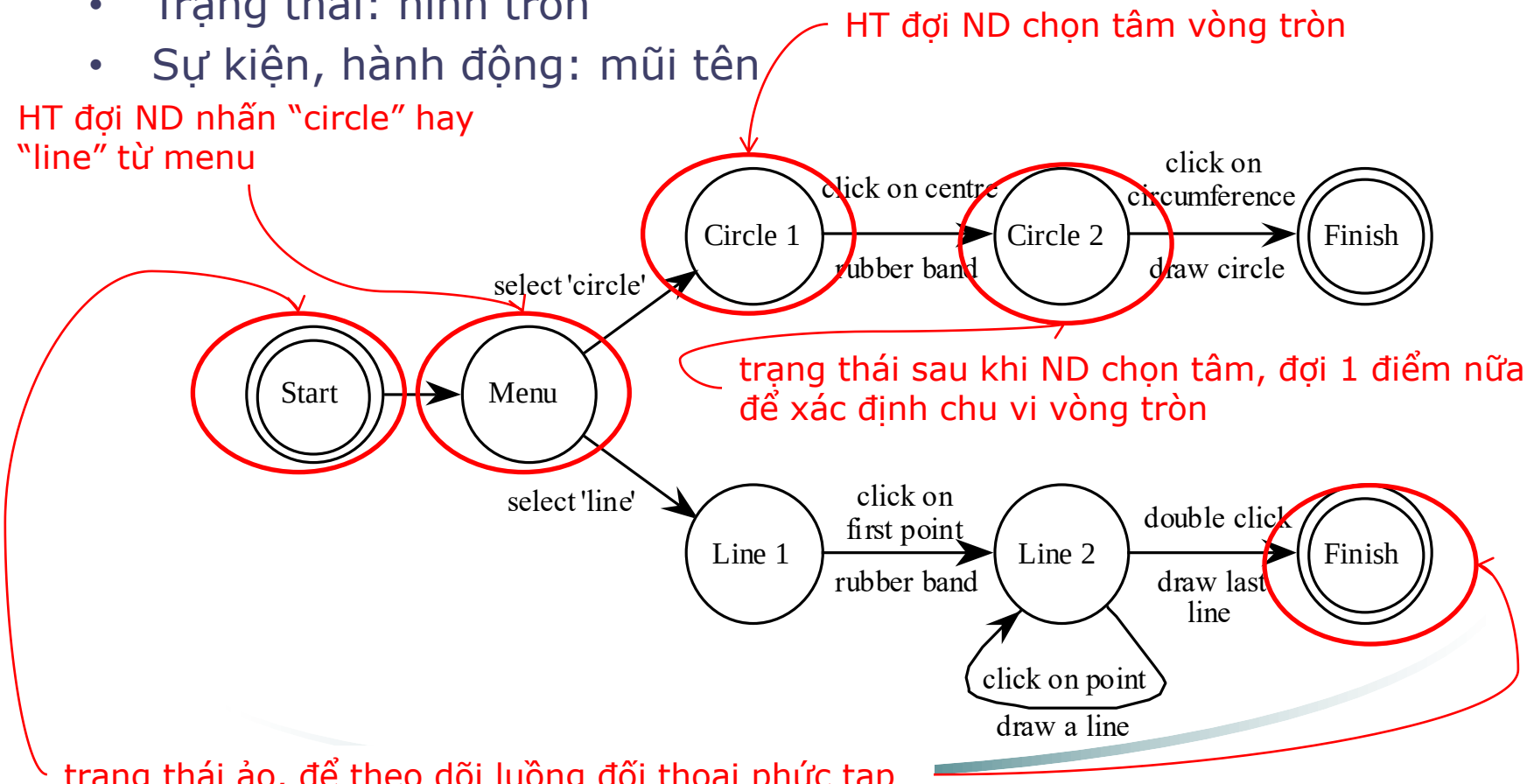
  → *Đối thoại tương tranh và bùng nổ tổ hợp*
  - Lưu đồ luồng (Flow Chart)
  - Lưu đồ JSD (Jackson Structured Design)

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 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.

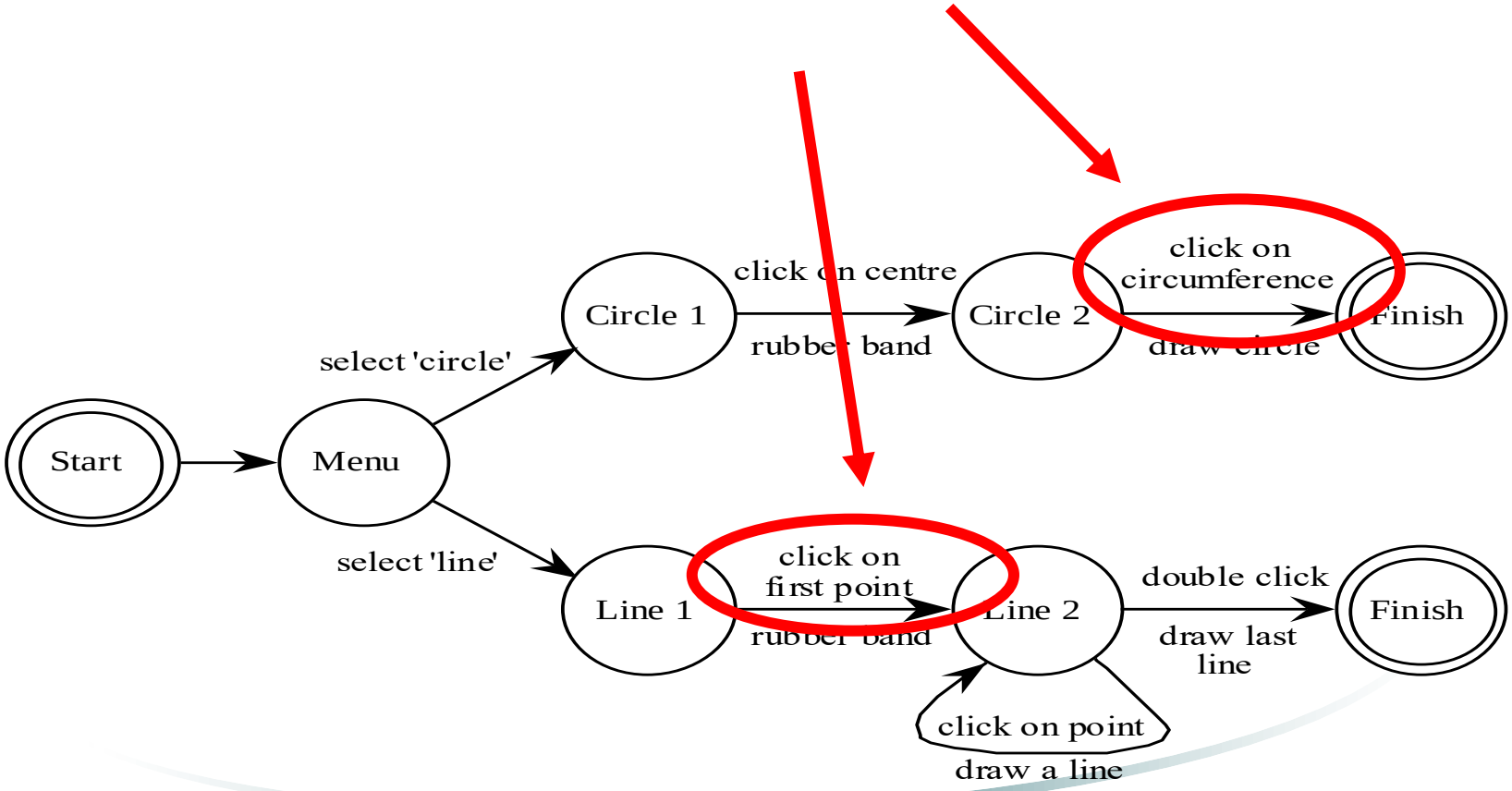
Ví dụ: Mạng dịch chuyển trạng thái biểu diễn công cụ vẽ

- 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



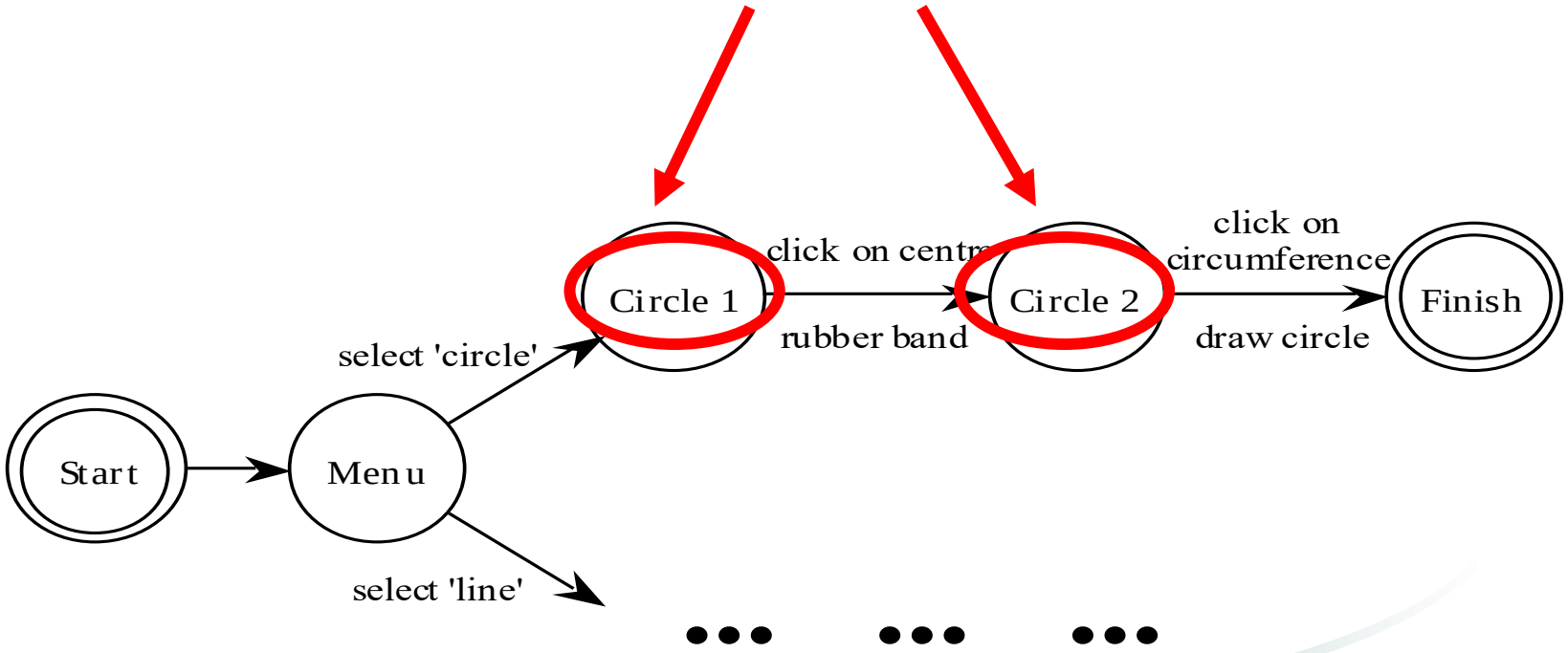
STN: sự kiện

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

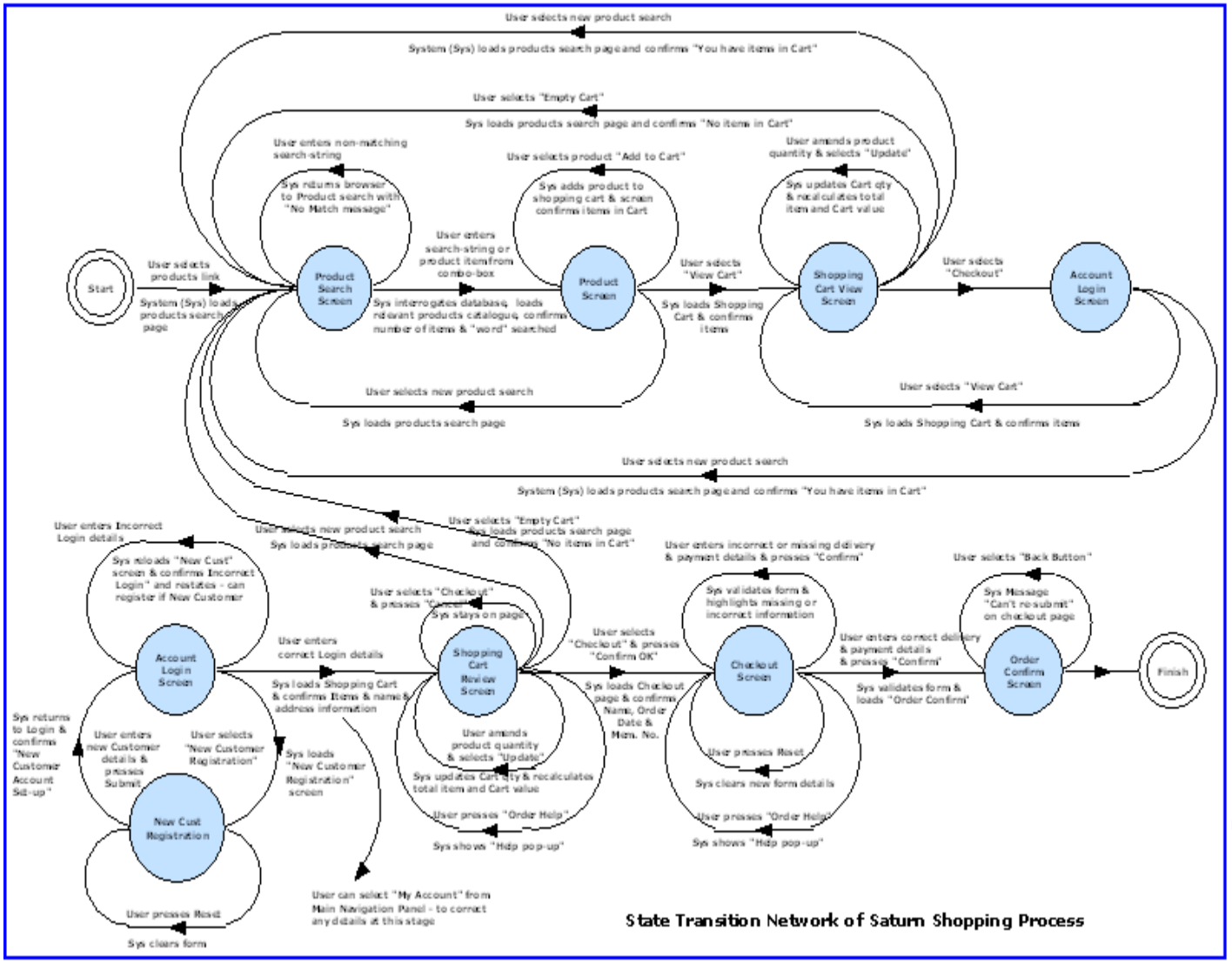


STN: trạng thái

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

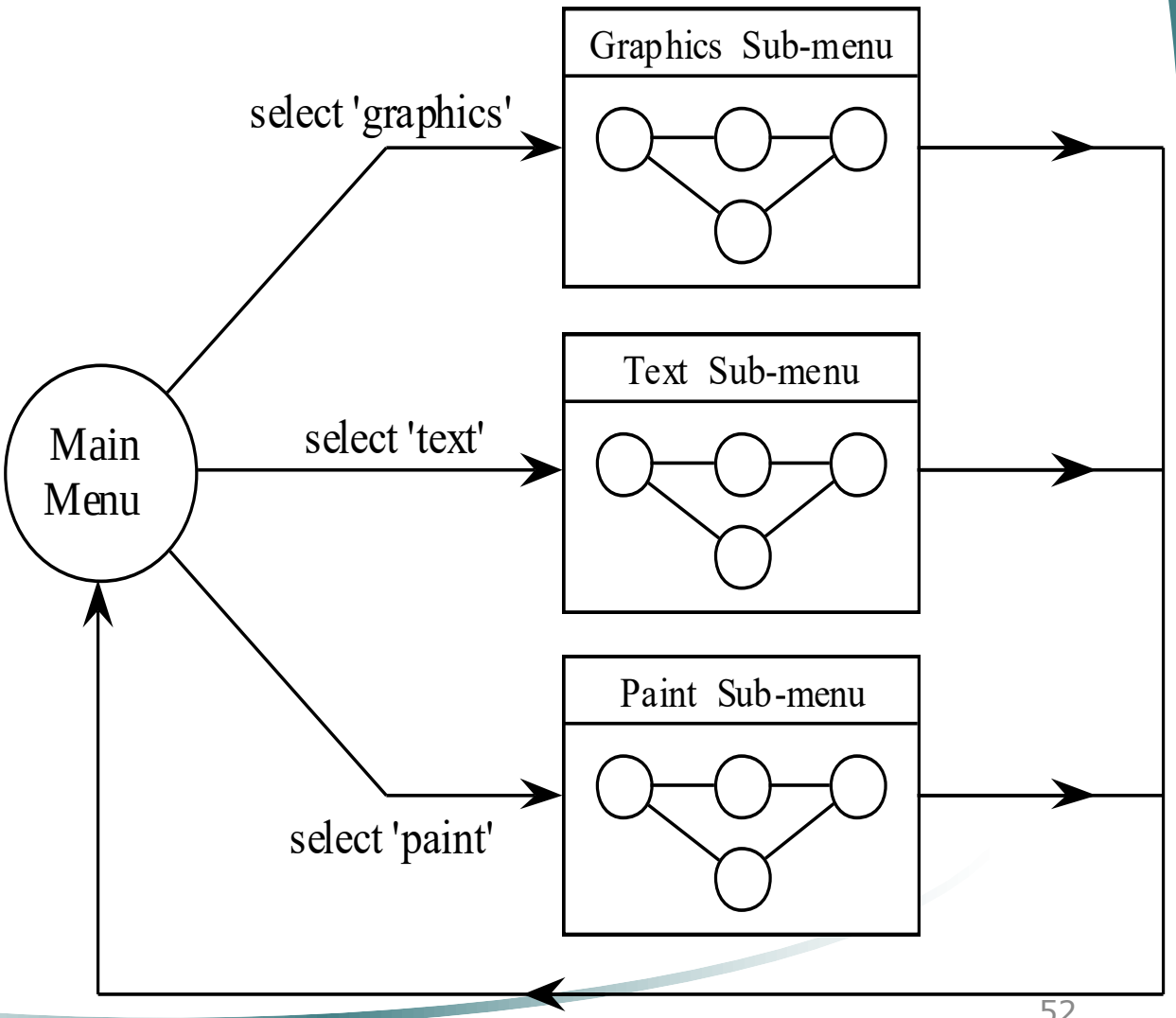


STN thường khá phức tạp



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

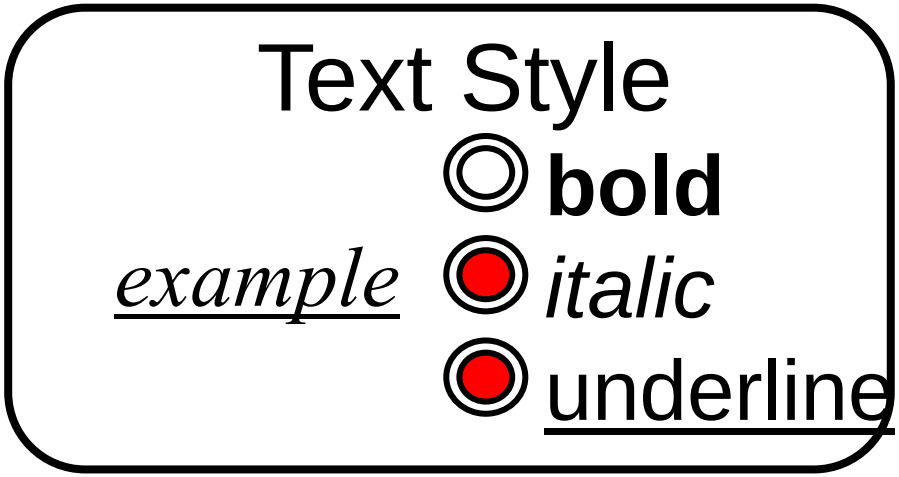


Sử dụng STN phân cấp

- 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

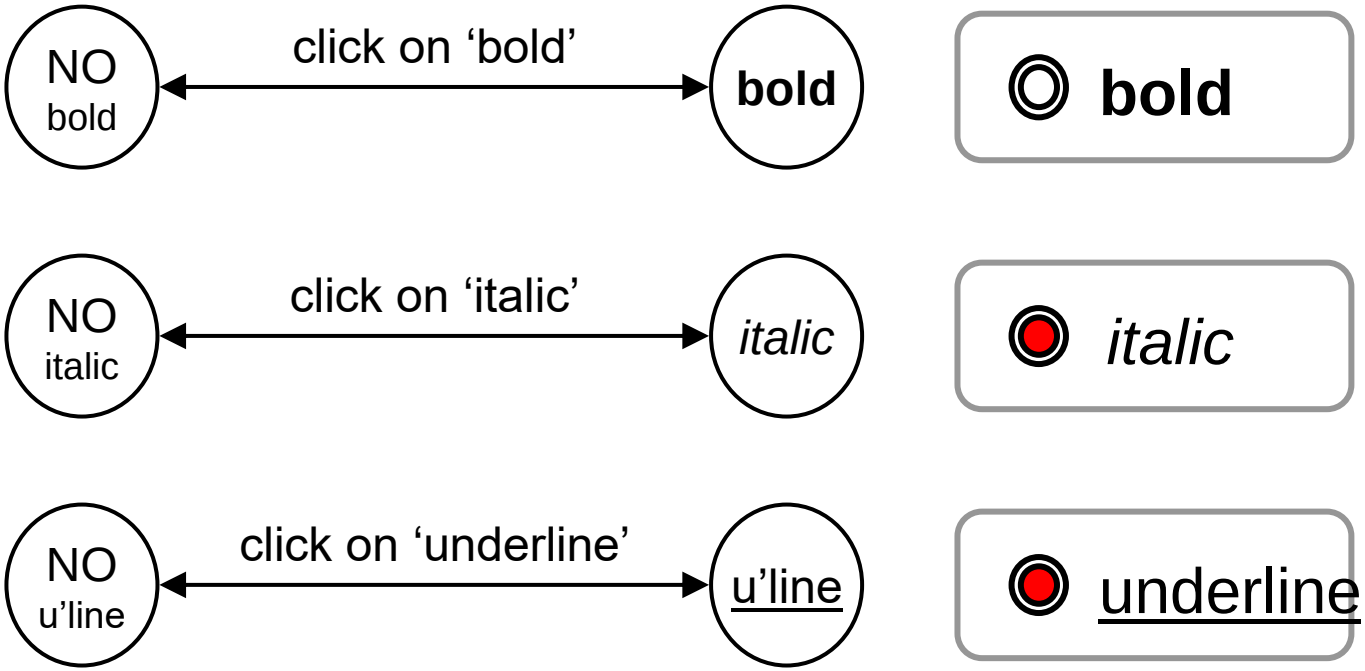
- Ví dụ: Giao diện thực đơn gồm 3 lựa chọn (dạng phím Toggle). Mỗi phím tương ứng với một kiểu định dạng: đậm, nghiêng hay gạch chân.
- Một đoạn văn bản có thể là nghiêng, đậm hay gạch chân và cũng có thể là bất kỳ tổ hợp nào của 3 thuộc tính trên.



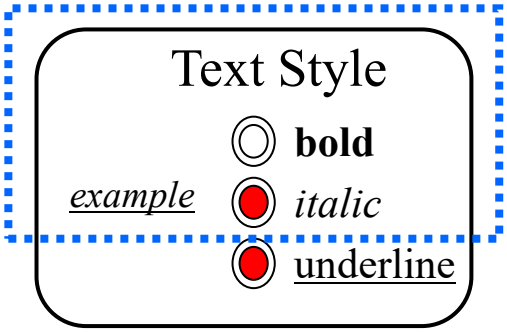
Nếu quan sát từng phím, có 1 STN 2 trạng thái.

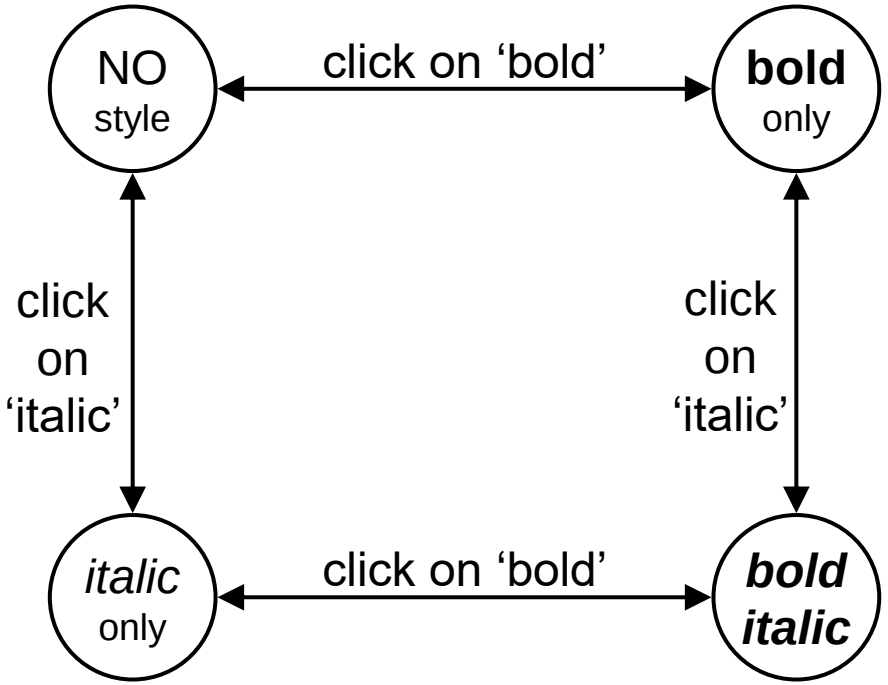
Nếu muốn biểu diễn tổ hợp trạng thái, cần tổ hợp lưu đồ.

Đối thoại tương tranh
1 STN per toggles

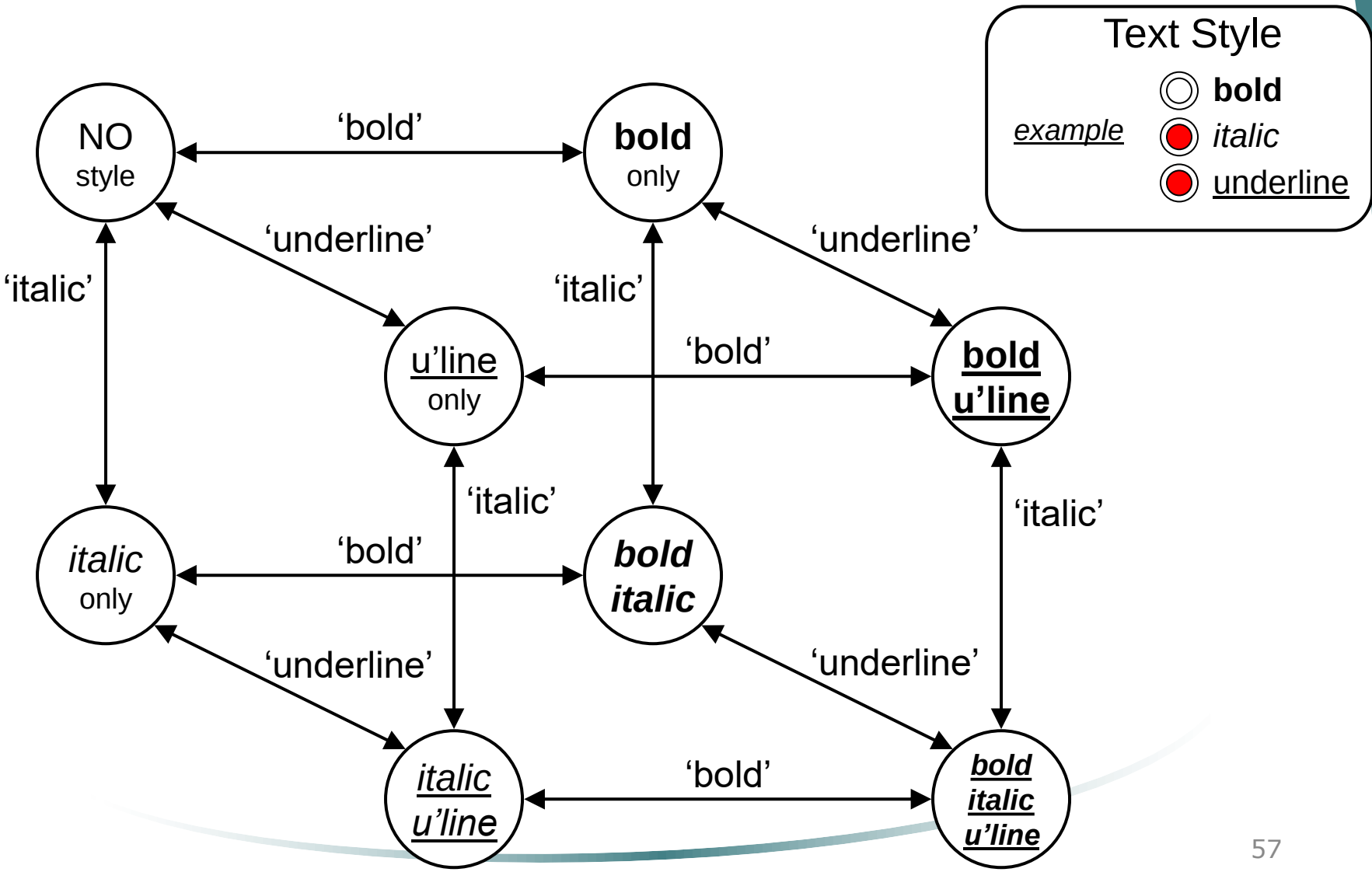


Đối thoại tương tranh
Tổ hợp nghiêng – đậm





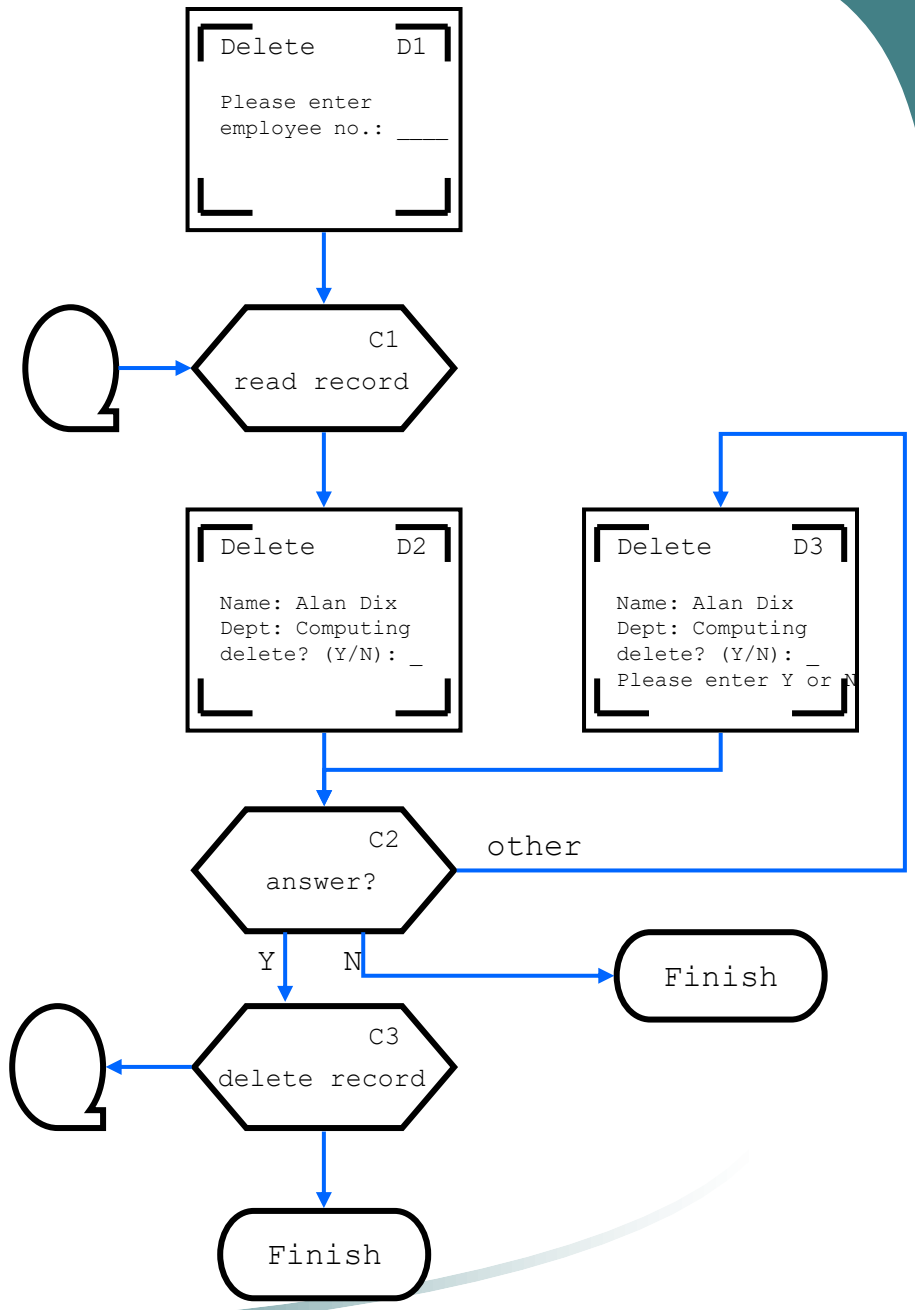
Đối thoại tương tranh
Bùng nổ tổ hợp đậm – nghiêng – gạch chân



d. Lưu đồ luồng (Flow charts)

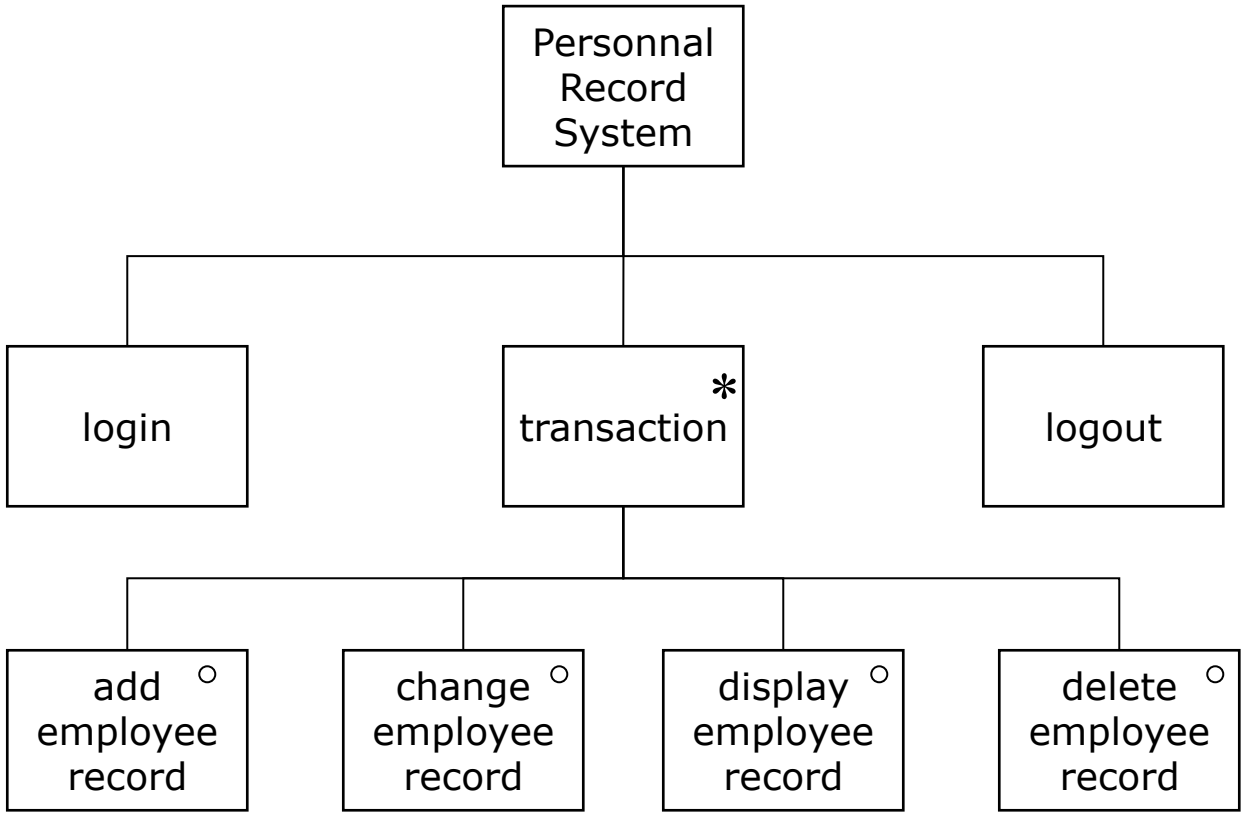
- Sử dụng nhiều loại hình khối khác nhau để biểu diễn các hoạt động khác nhau
  - Hình khối biểu diễn quy trình hay sự kiện
  - Hình khối không biểu diễn trạng thái

→ 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

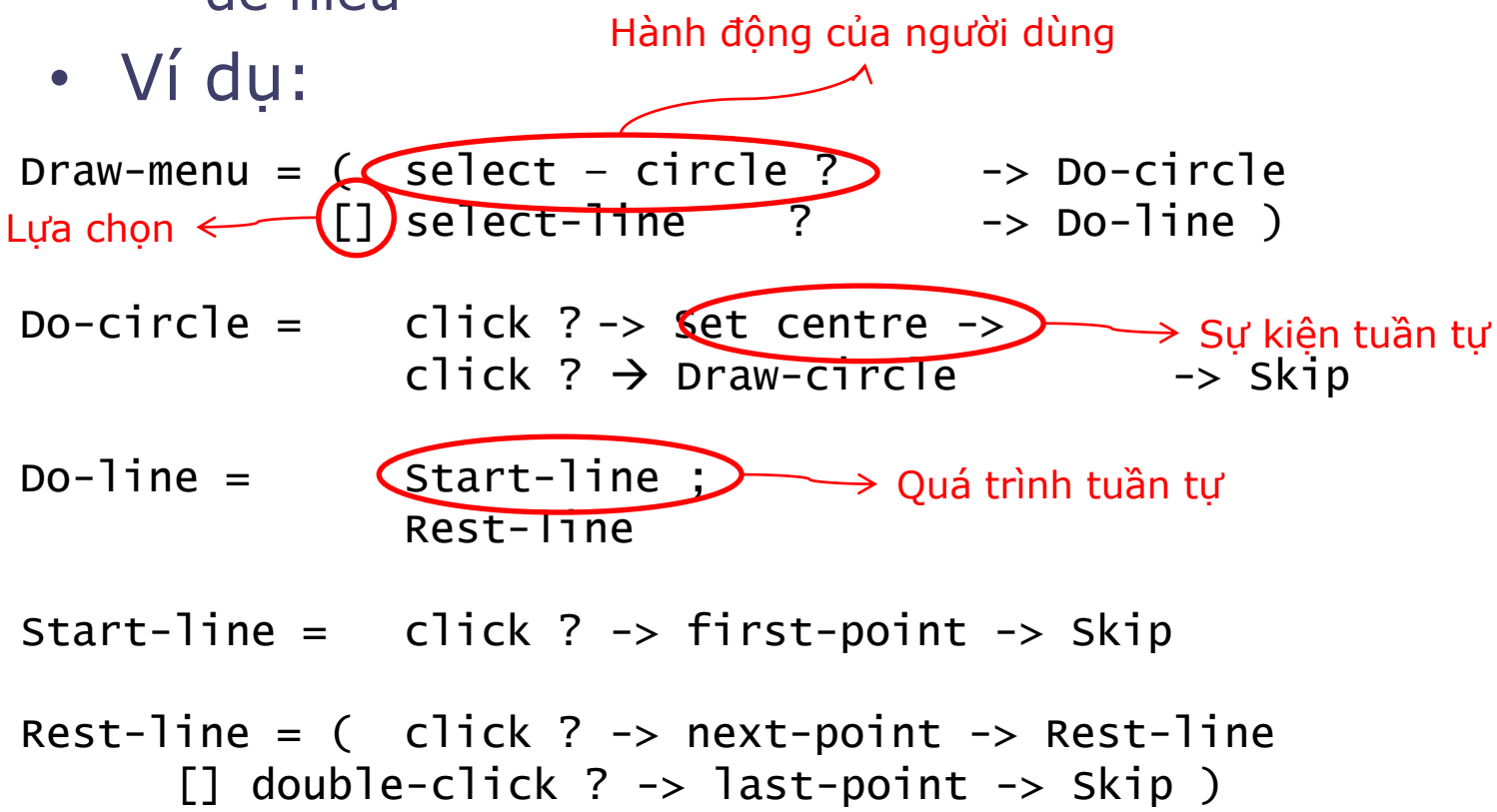
- STN không phù hợp với mô tả tương tranh
- Luật sản xuất thì không phù hợp với mô tả tuần tự, trạng thái
- Việc xử lý tương tranh + tuần tự đặt ra trong nhiều bài toán: truyền thông, điều khiển tương tranh
- Đại số quy trình (Process Algebra) là một ký pháp hình thức được phát triển cho các quá trình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




```
Draw-menu = ( select – circle ?     -> Do-circle
            [] select-line    ?     -> Do-line )

Do-circle =    click ? -> set centre ->
               click ? → Draw-circle      -> Skip

Do-line =      Start-line ;
               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 qua trình. Dấu "->" chỉ dùng sau 1 sự kiện.
- []: chỉ ra sự lựa chọn như dòng 1: ND có thể chọn cirle hay chọn line.

Hành động của người dùng

```
Draw-menu = ( select – circle ?     -> Do-circle
          [] select-line   ?        -> Do-line )

Do-circle =     click ? -> set centre ->
                click ? Draw-circle   -> Skip

Do-line =       Start-line ;
                Rest-line

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

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

Lựa chọn

Sự kiện tuần tự

Quá trình tuần tự

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 thoả.

```
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' |}
  . . . . }
…  }
```

- 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

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

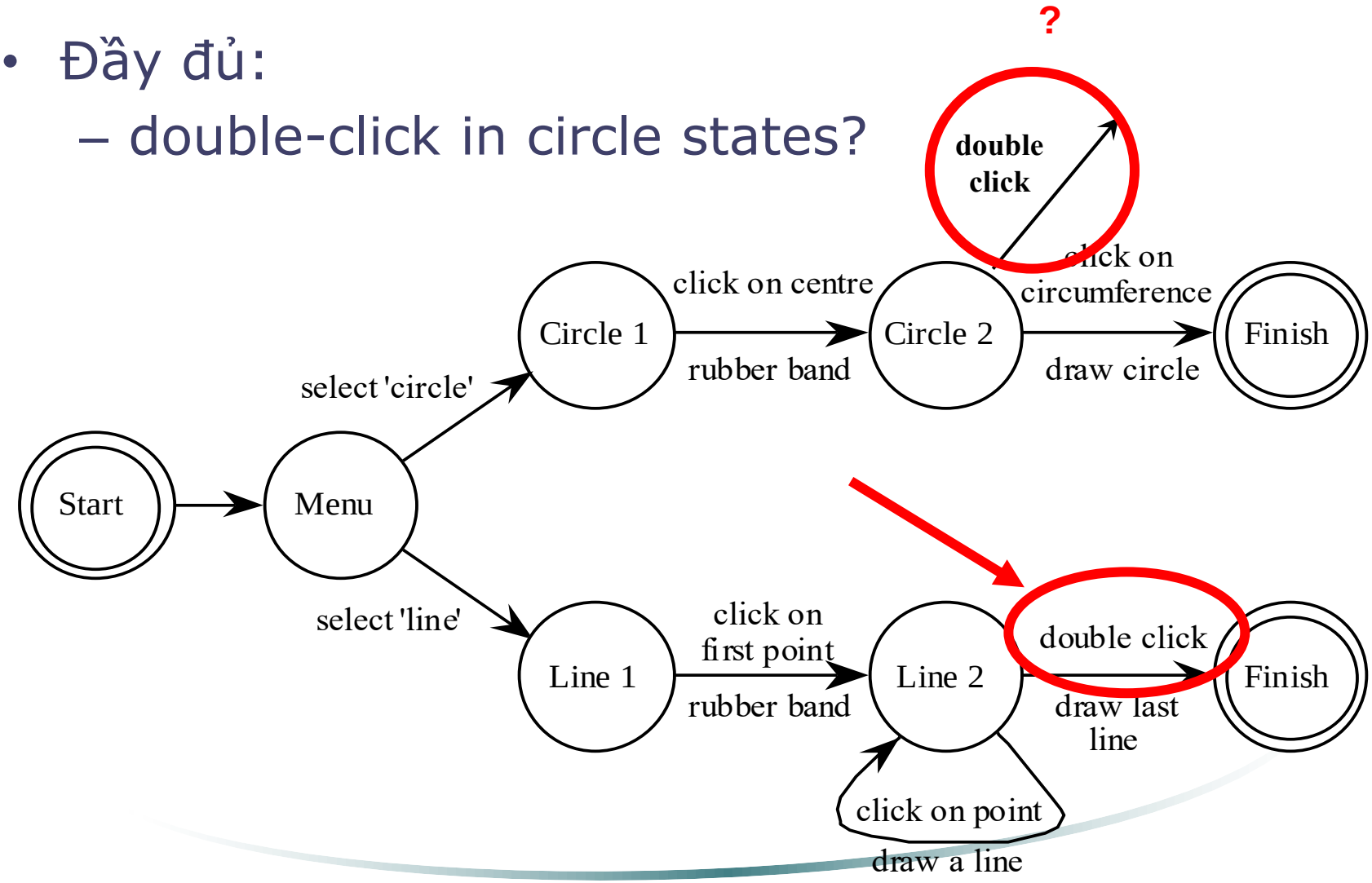
- Các cách thức mà đối thoại có thể được phân tích nhằm phát hiện tính tiện dụng tiềm năng bằng cách xem xét các nguyên lý thiết kế giao diện
- Trước tiên tập trung **vào hành động** của ND, tiếp theo **là trạng thái** của đối thoại. Cuối cùng là xem xét cách **biểu diễn và từ vựng.**

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: consistency
  - 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?

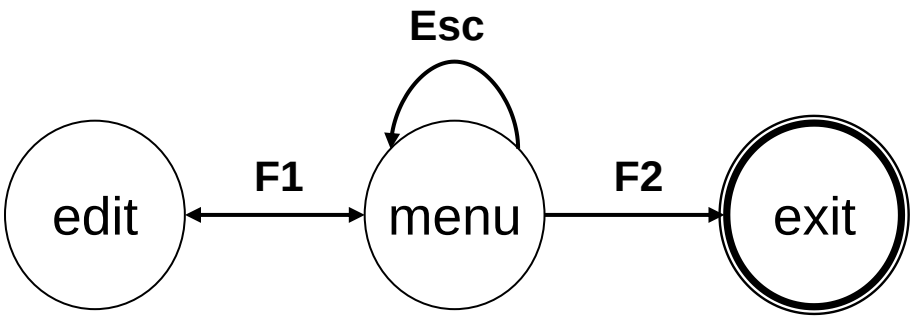


b. Tính chất của trạng thái

- 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

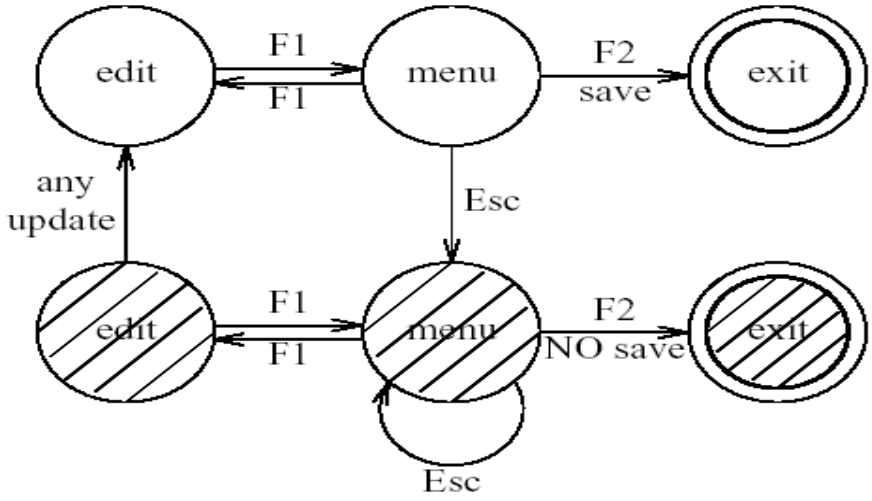
- Có 2 chế độ và thoát
  - F1 - thay đổi chế độ
  - F2 - thoát (và tự động ghi nội dung)
  - Esc - không thay đổi chế độ



  - 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

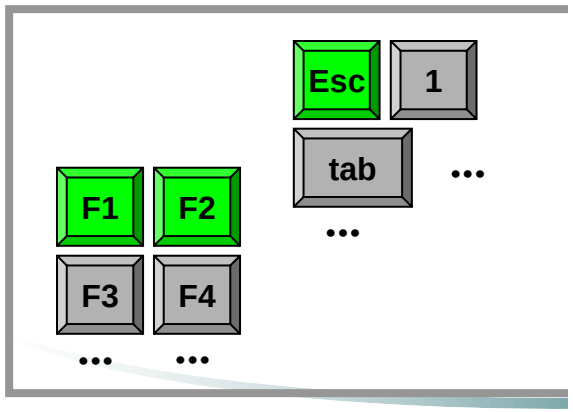
- Hình thức và chức năng của hội thoại ?
- Tính trừu tượng: Biểu diễn trừu tượng của hội thoại.
  - Ví dụ, nhập tọa độ một điểm từ bàn phím hay click chuột lên bề mặt đối tượng
- Các chế độ nhãn: Tập trung vào tính dễ nhìn, dễ quan sát và dễ dự đoán của các nhãn biểu diễn
- Tính phù hợp của kiểu hội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

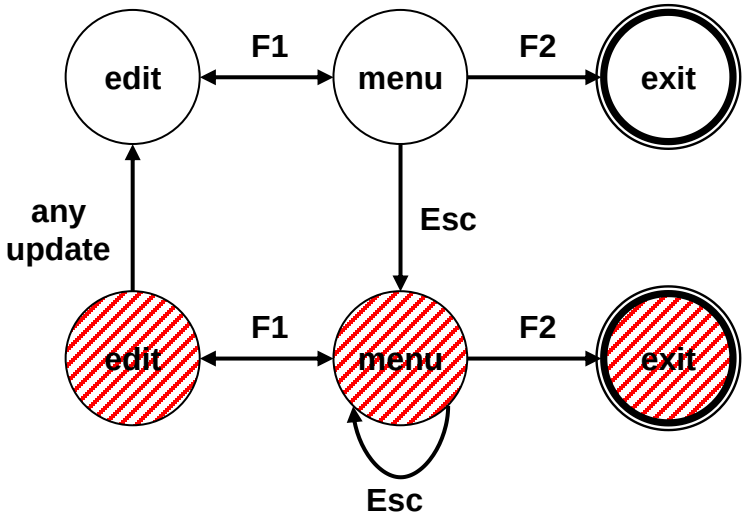
Ví dụ: Mô thức từ vựng của bộ xử lý văn bản

- 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

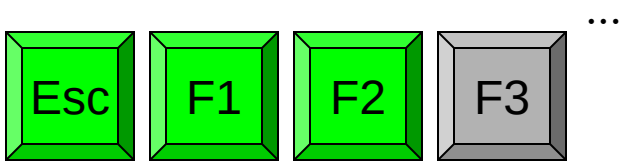
- old keyboard - OK





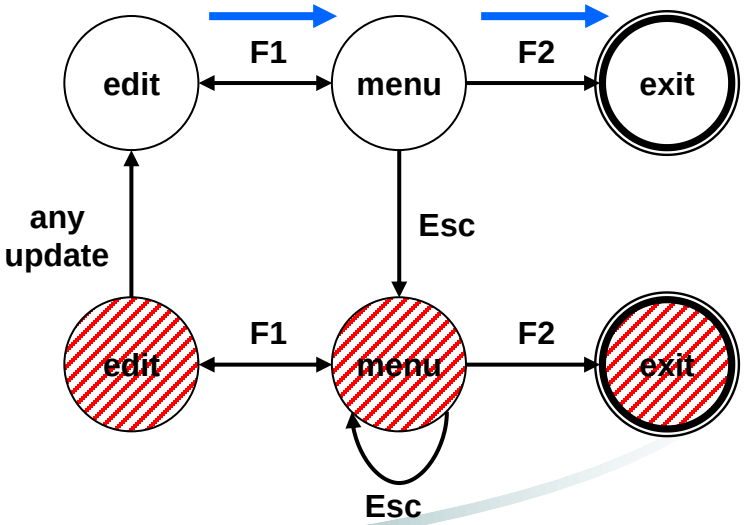
Ví dụ: hiện thị trạng thái nguy hiểm của bộ xử lý văn bản

- new keyboard layout



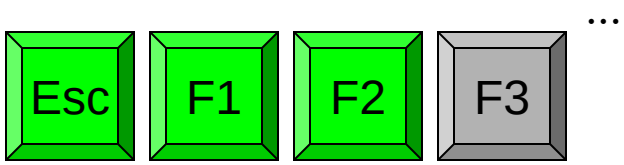
intend F1-F2 (save)

finger catches Esc



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

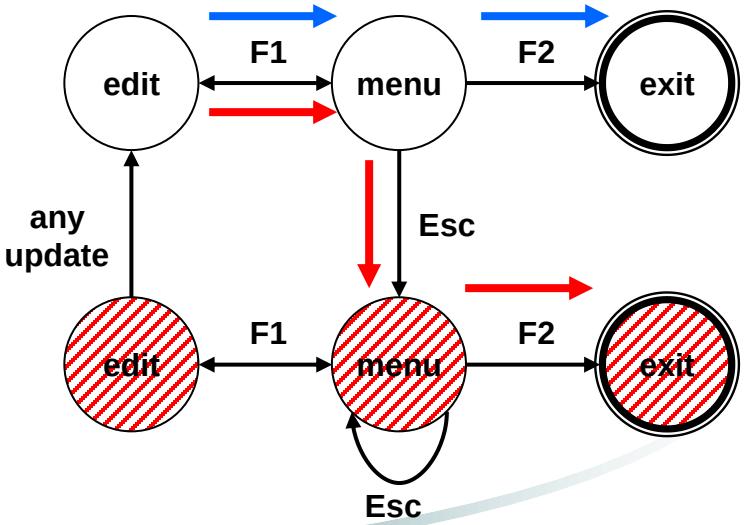
- new keyboard layout



intend F1-F2 (save)

finger catches Esc

F1-Esc-F2 - disaster!



V. MÔ HÌNH TƯƠNG TÁC

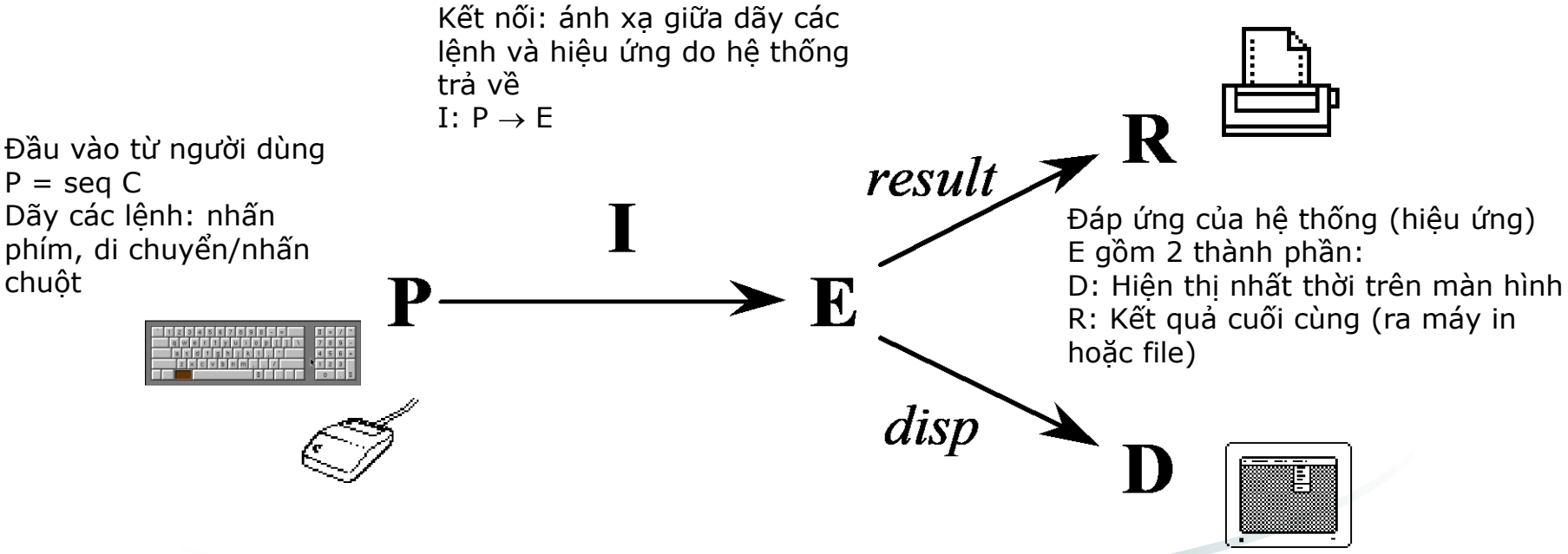
1. Mô hình PIE
2. Phân tích trạng thái-sự kiện

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