

# EXPERT SYSTEMS

---

PGS.TS. Hoàng Văn Dũng  
dunghv@hcmute.edu.vn

# Nội dung

---

- Tổng quan về hệ chuyên gia
- Biểu diễn tri thức
- Các kỹ thuật suy diễn và lập luận
- Phát triển và thiết kế hệ chuyên gia
- Máy học và công cụ xây dựng XPS
- Case study

# *Tài liệu tham khảo*

---

## **Textbook**

- Joseph C. Giarratano and Gary D. Riley (2005), "Expert Systems: Principles and Programming", Fourth Edition, Thomson Course Technology.
- Peter Jackson (1999), *Introduction to Expert Systems*, Third Edition, Addison-Wesley Longman, Harlow, England.

# Chapter 1

# Introduction

---

# Hệ chuyên gia

- In artificial intelligence, an expert system is a computer system that emulates the decision-making ability of a human expert.

*[Jackson, Peter (1998), Introduction To Expert Systems, Addison Wesley, p.2]*

- XPS (Expert System) là một chương trình máy tính thông minh sử dụng tri thức (knowledge) và các thủ tục suy luận (inference procedures) để giải những bài toán tương đối khó khăn đòi hỏi những chuyên gia mới giải được.

# Hệ chuyên gia

---

- Các thuật ngữ tương đương XPS:
  - Hệ thống dựa trên tri thức (knowledge-based system)
  - XPS dựa trên tri thức (knowledge-based expert system)
- Một XPS gồm ba thành phần chính:
  - Cơ sở tri thức (knowledge base)
  - Máy suy diễn hay máy suy diễn (inference engine)
  - Hệ thống giao tiếp với người sử dụng (user interface)

# Hệ chuyên gia

- XPS là một hệ thống sử dụng các khả năng lập luận để đạt tới các kết luận.
- Dạng phổ biến nhất của XPS là một chương trình gồm một tập luật phân tích thông tin (thường được cung cấp bởi người sử dụng hệ thống) về một lớp vấn đề cụ thể nào đó, cùng tập phân tích về các vấn đề tương ứng.
- Tùy theo thiết kế chương trình mà XPS đưa ra trình tự các hành động cần thực hiện để giải quyết vấn đề.

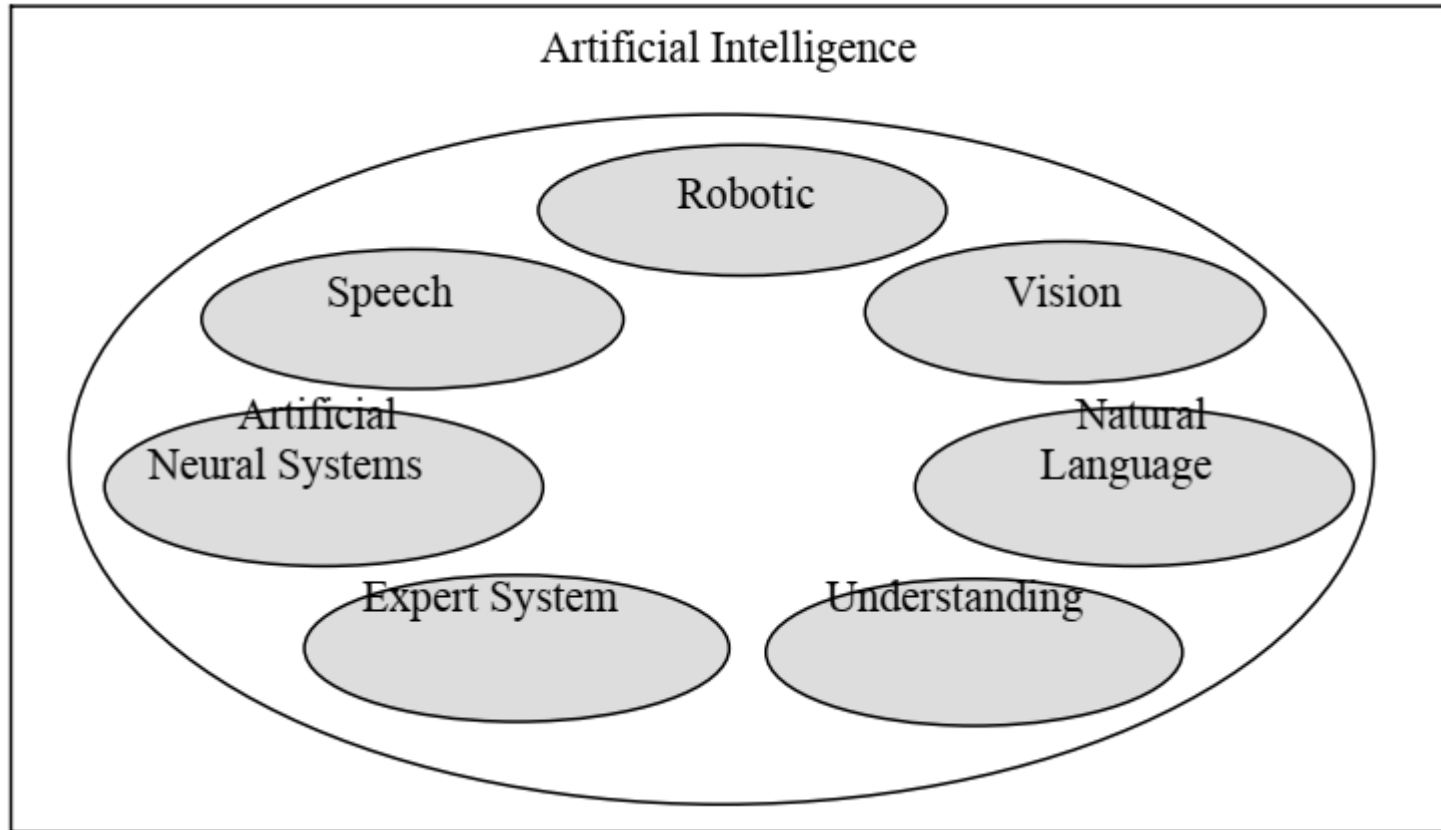
# Hệ chuyên gia

- Các chương trình XPS đã được phát triển từ những thập niên đầu 1970 và bắt đầu trở thành ứng dụng thương mại từ thập niên 1980
- Nhiều XPS được thiết kế và xây dựng để phục vụ các lĩnh vực:
  - Kế toán, kiểm toán (accounting, auditing)
  - Dịch vụ tư vấn tài chính (financial service),
  - Điều khiển tiến trình (process control),
  - Tài nguyên con người (human resources),
  - Y học (medical diagnostic decision, healthcare)



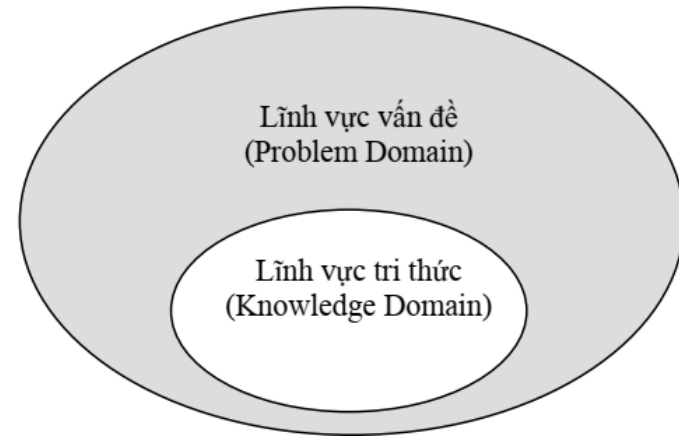
# Hệ chuyên gia

- *Artificial intelligent applications*



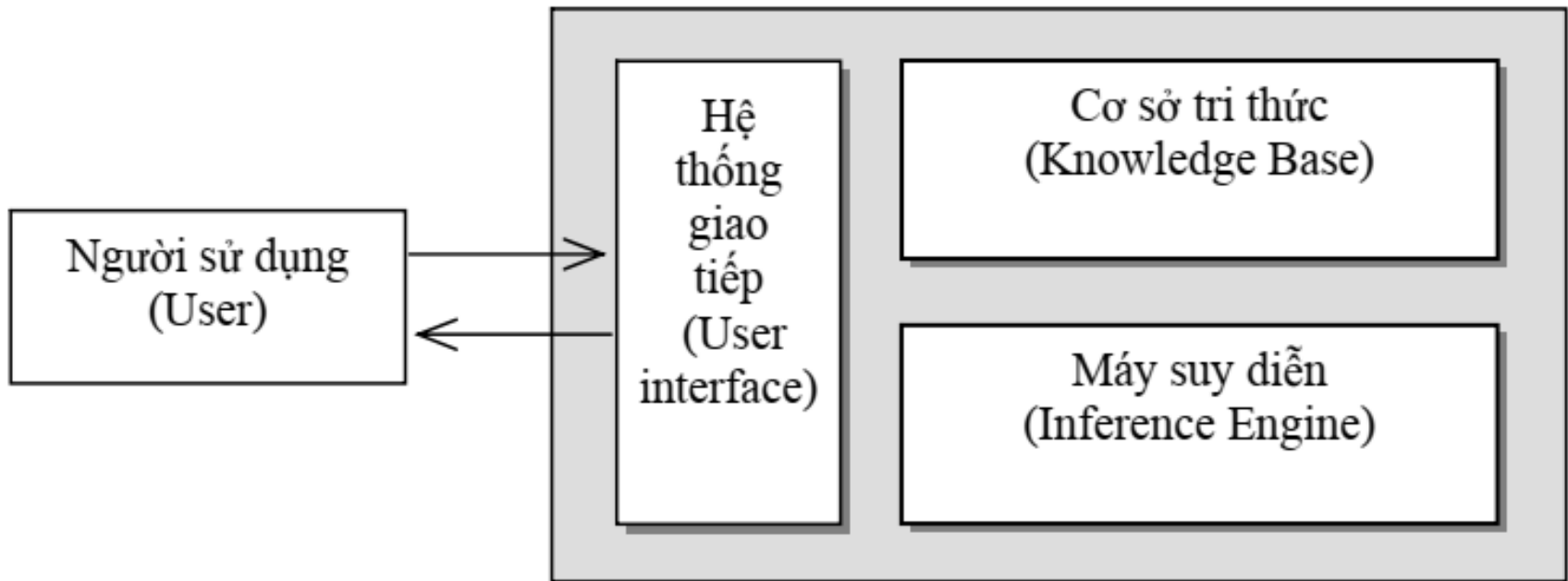
# Hệ chuyên gia

- Các XPS sử dụng các tri thức (knowledge) của các chuyên gia để giải quyết các vấn đề khác nhau thuộc từng lĩnh vực cụ thể.
- Quan hệ giữa lĩnh vực *bài toán* và lĩnh vực *tri thức*
- Mỗi XPS chỉ đặc trưng giải quyết cho một lĩnh vực bài toán (problem domain) cụ thể nào đó.
  - Ví dụ như XPS về y học, tài chính,, v.v...,
  - Một XPS mà không phải có thể giải quyết cho bất cứ lĩnh vực bài toán nào.



# Hệ chuyên gia

- Hoạt động của hệ chuyên gia



# Đặc trưng của XPS

Có bốn đặc trưng cơ bản của XPS:

- *Hiệu quả cao* (high performance): Khả năng trả lời với mức độ tinh thông bằng hoặc cao hơn so với chuyên gia (người) trong cùng lĩnh vực.
- *Thời gian trả lời thoả đáng* (adequate response time): Thời gian trả lời hợp lý, bằng hoặc nhanh hơn so với chuyên gia (người) để đi đến cùng một quyết định. XPS là một hệ thống thời gian thực (real time system).
- *Độ tin cậy cao* (good reliability): Không thể xảy ra sự cố hoặc giảm sút độ tin cậy khi sử dụng.
- *Dễ hiểu* (understandable): XPS giải thích các bước suy luận một cách dễ hiểu và nhất quán, không giống như cách trả lời bí ẩn của các hộp đen (black box).

# Ưu điểm của XPS

- *Phổ cập* (increased availability): Là sản phẩm chuyên gia, được phát triển không ngừng với hiệu quả sử dụng không thể phủ nhận.
- *Giảm giá thành* (reduced cost).
- *Giảm rủi ro* (reduced dangers): Giúp con người tránh được trong các môi trường rủi ro, nguy hiểm.
- *Tính thường trực* (Permanance): Bất kể lúc nào cũng có thể khai thác sử dụng, trong khi con người có thể mệt mỏi, nghỉ ngơi hay vắng mặt.
- *Đa lĩnh vực* (multiple expertise): Chuyên gia về nhiều lĩnh vực khác nhau và được khai thác đồng thời bất kể thời gian sử dụng.

# Ưu điểm của XPS

- *Độ tin cậy* (increased reliability): Luôn đảm bảo độ tin cậy khi khai thác.
- *Khả năng giảng giải* (explanation): Câu trả lời với mức độ tinh thông được giảng giải rõ ràng chi tiết, dễ hiểu.
- *Khả năng trả lời* (fast response): Trả lời theo thời gian thực, khách quan.
- *Tính ổn định, suy luận có lý và đầy đủ mọi lúc mọi nơi* (steady, une motional, and complete response at all times).
- *Trợ giúp thông minh như một người hướng dẫn* (intelligent -tutor).
- *Có thể truy cập như là một cơ sở dữ liệu thông minh* (intelligent database).

# Sự phát triển của XPS

Năm	Các sự kiện
1943	Dịch vụ bưu điện ; mô hình Neuron của (Mc Culloch and Pitts Model)
1954	Thuật toán Markov (Markov Algorithm) điều khiển thực thi các luật
1956	Hội thảo Dartmouth ; lý luận logic ; tìm kiếm nghiệm suy (heuristic search) ; thống nhất thuật ngữ trí tuệ nhân tạo (AI: Artificial Intelligence)
1957	Rosenblatt phát minh khả năng nhận thức ; Newell, Shaw và Simon đề xuất giải bài toán tổng quát (GPS: General Problem Solver)
1958	Mc Carthy đề xuất ngôn ngữ trí tuệ nhân tạo LISA (LISA AI language)
1962	Nguyên lý Rosenblatt's về chức năng thần kinh trong nhận thức (Rosenblatt's <i>Principles of Neurodynamicdynamics on Perceptions</i> )
1965	Phương pháp hợp giải Robinson. Ứng dụng logic mờ (fuzzy logic) trong suy luận về các đối tượng mờ (fuzzy object) của Zadeh. Xây dựng hệ chuyên gia đầu tiên về nha khoa DENDRAL (Feigenbaum , Buchanan , et.al)
1968	Mạng ngữ nghĩa (semantic nets), mô hình bộ nhớ kết hợp (associative memory model) của Quillian
1969	Hệ chuyên gia về Toán học MACSYMA (Martin and Moses)
1970	Ứng dụng ngôn ngữ PROLOG (Colmerauer, Roussell, et, al.)
1971	Hệ chuyên gia HEARSAY I về nhận dạng tiếng nói (speech recognition). Xây dựng các luật giải bài toán con người ( <i>Human Problem Solving</i> popularizes rules (Newell and Simon)
1973	Hệ chuyên gia MYCIN về chẩn trị y học (Shortliffe, et.al.)
1975	Lý thuyết khung (frames), biểu diễn tri thức (knowledge representation) (Minsky)

# Sự phát triển của XPS

Năm	Các sự kiện
1976	Toán nhân tạo (AM: Artificial Mathematician) (Lenat). Lý thuyết Dempster–Shafer về tính hiển nhiên của lập luận không chắc chắn (Dempster–Shafer theory of Evidence for reason under uncertainty). Ứng dụng hệ chuyên gia PROSPECTOR trong khai thác hầm mỏ (Duda, Har)
1977	Sử dụng ngôn ngữ chuyên gia OPS (OPS expert system shell) trong hệ chuyên gia XCON/R1 (Forgy)
1978	Hệ chuyên gia XCON/R1 (McDermott, DEC) để bảo trì hệ thống máy tính DEC (DEC computer systems)
1979	Thuật toán mạng về so khớp nhanh (rete algorithm for fast pattern matching) của Forgy ; thương mại hoá các ứng dụng về trí tuệ nhân tạo
1980	Ký hiệu học (symbolics), xây dựng các máy LISP (LISP machines) từ LMI.
1982	Hệ chuyên gia về Toán học (SMP math expert system) ; mạng nơ-ron Hopfield (Hopfield Neural Net) ; Dự án xây dựng máy tính thông minh thế hệ 5 ở Nhật bản (Japanese Fifth Generation Project to develop intelligent computers)
1983	Bộ công cụ phục vụ hệ chuyên gia KEE (KEE expert system tool) (intelli Corp)
1985	Bộ công cụ phục vụ hệ chuyên gia CLIPS (CLIPS expert system tool (NASA)

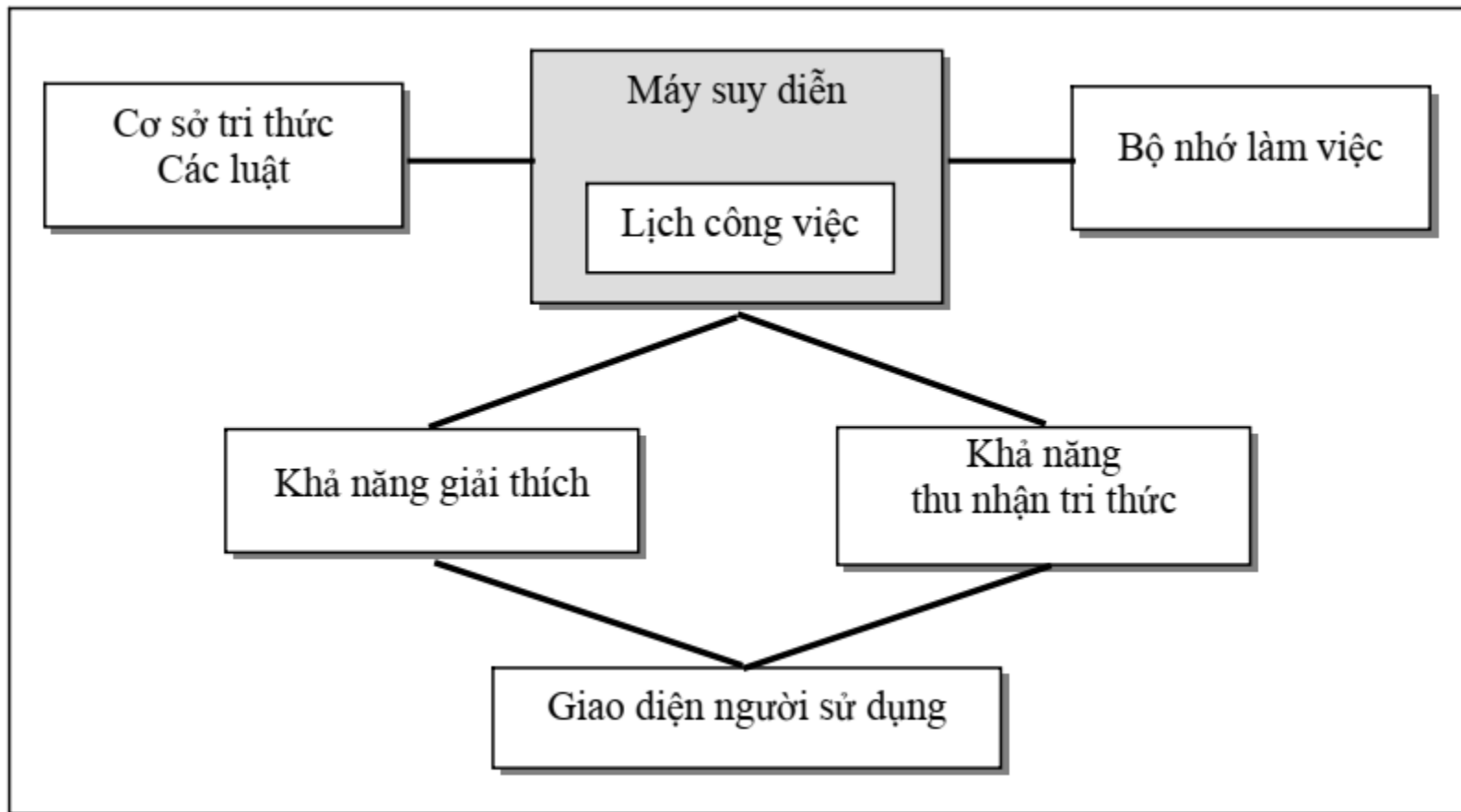


# Các lĩnh vực ứng dụng của XPS

Category	Problem addressed	Examples
Interpretation	Inferring situation descriptions from sensor data	Hearsay (speech recognition), PROSPECTOR
Prediction	Inferring likely consequences of given situations	Preterm Birth Risk Assessment <sup>[56]</sup>
Diagnosis	Inferring system malfunctions from observables	CADUCEUS, <a href="#">MYCIN</a> , PUFF, Mistral, <sup>[57]</sup> Eydenet, <sup>[58]</sup> Kaleidos <sup>[59]</sup>
Design	Configuring objects under constraints	<a href="#">Dendral</a> , <a href="#">Mortgage Loan Advisor</a> , <a href="#">R1</a> (DEC VAX Configuration), SID (DEC <a href="#">VAX 9000 CPU</a> )
<b>Planning</b>	Designing actions	Mission Planning for Autonomous Underwater Vehicle <sup>[60]</sup>
Monitoring	Comparing observations to plan vulnerabilities	REACTOR <sup>[61]</sup>
Debugging	Providing incremental solutions for complex problems	SAINT, MATHLAB, MACSYMA
Repair	Executing a plan to administer a prescribed remedy	Toxic Spill Crisis Management
Instruction	Diagnosing, assessing, and repairing student behavior	SMH.PAL, <sup>[62]</sup> Intelligent Clinical Training, <sup>[63]</sup> STEAMER <sup>[64]</sup>
Control	Interpreting, predicting, repairing, and monitoring system behaviors	Real Time Process Control, <sup>[65]</sup> Space Shuttle Mission Control <sup>[66]</sup>

# Kiến trúc tổng quát của XPS

- Một XPS kiểu mẫu gồm 7 thành phần cơ bản sau:



# Kiến trúc tổng quát của XPS

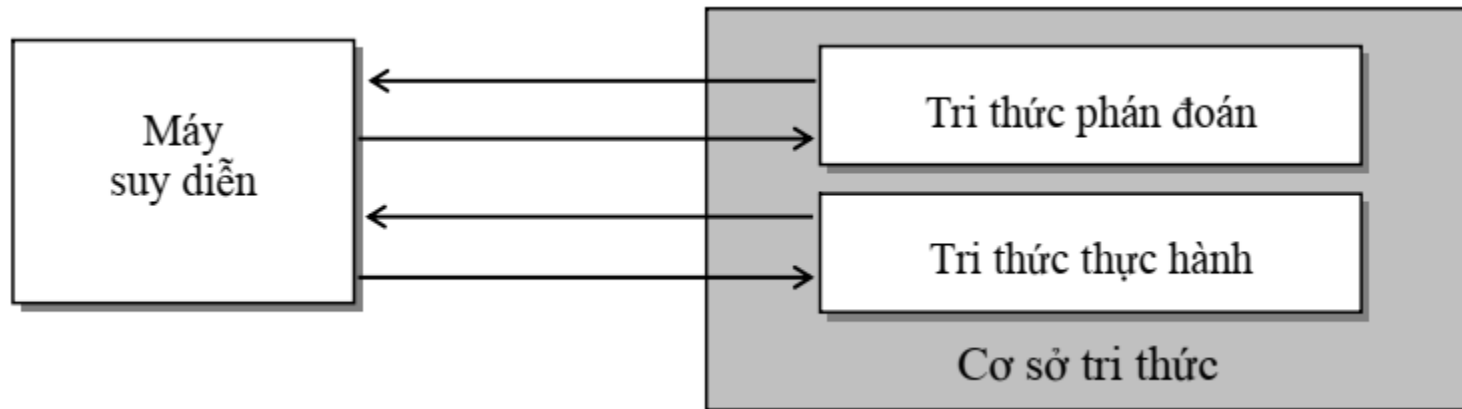
- *Cơ sở tri thức* (knowledge base). Gồm các phần tử (hay đơn vị) tri thức, thông thường được gọi là *luật* (rule), được tổ chức như một cơ sở dữ liệu.
- *Máy suy diễn* (inference engine). Công cụ (chương trình, hay bộ xử lý) tạo ra sự suy luận bằng cách quyết định xem những luật nào sẽ làm thỏa mãn các sự kiện, các đối tượng, chọn ưu tiên các luật thỏa mãn, thực hiện các luật có tính ưu tiên cao nhất.
- *Lịch công việc* (agenda). Danh sách các luật ưu tiên do máy suy diễn tạo ra thỏa mãn các sự kiện, các đối tượng có mặt trong bộ nhớ làm việc

# Kiến trúc tổng quát của XPS

- *Bộ nhớ làm việc* (working memory). Cơ sở dữ liệu toàn cục chứa các sự kiện phục vụ cho các luật.
- *Khả năng giải thích* (explanation facility). Giải nghĩa cách lập luận của hệ thống cho người sử dụng.
- *Khả năng thu nhận tri thức* (explanation facility). Cho phép người sử dụng bổ sung các tri thức vào hệ thống một cách tự động thay vì tiếp nhận tri thức bằng cách mã hoá tri thức một cách tường minh. Khả năng thu nhận tri thức là yếu tố mặc nhiên của nhiều hệ chuyên gia.
- *Giao diện người sử dụng* (user interface). Là nơi người sử dụng và XPSTrao đối với nhau.

# Kiến trúc tổng quát của XPS

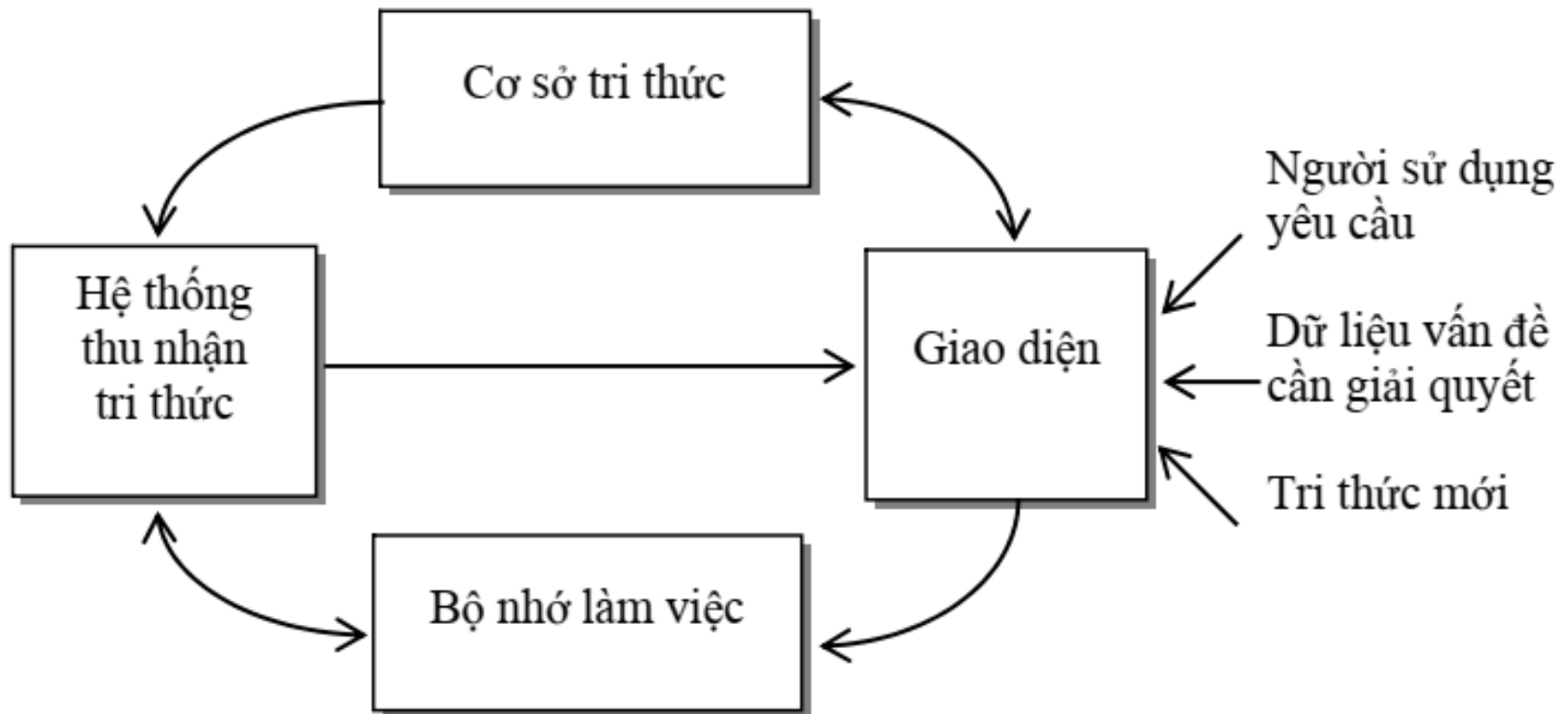
- Kiến trúc tổng quát mối quan hệ máy suy diễn và cơ sở tri thức



- Tri thức được phân chia thành 2 loại, máy suy diễn là công cụ triển khai các cơ chế (hay kỹ thuật) tổng quát để tổ hợp các tri thức phán đoán và các tri thức thực hành.

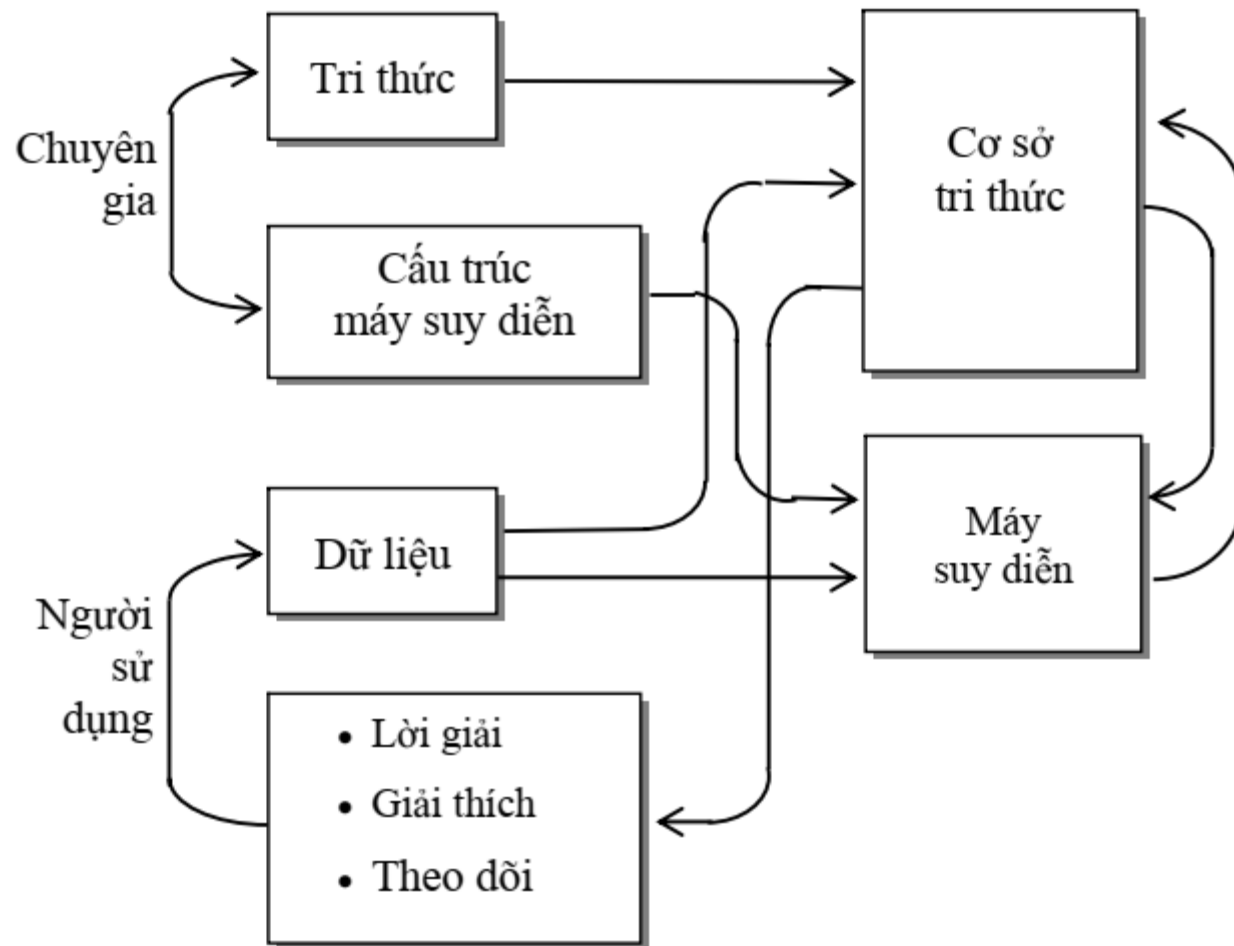
# Một số mô hình kiến trúc XPS

- Mô hình J. L. Ermine*



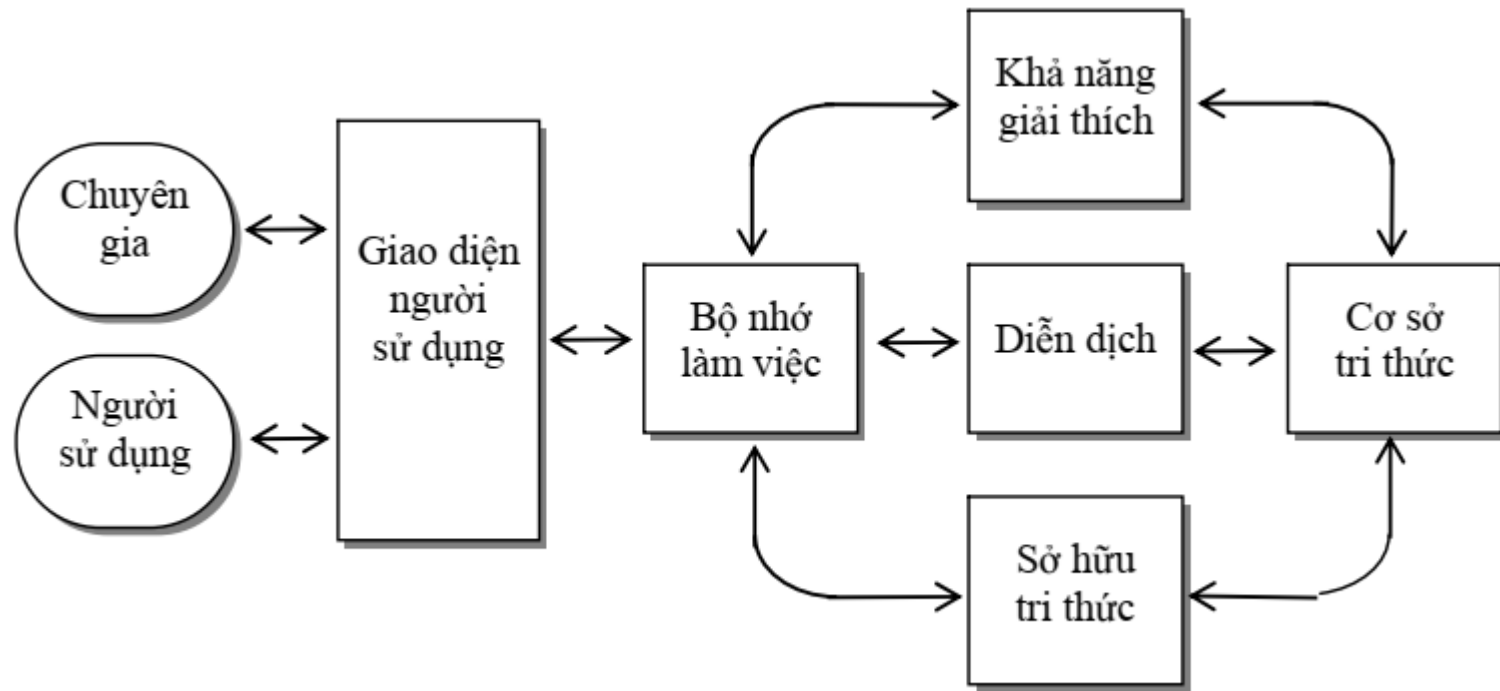
# Một số mô hình kiến trúc XPS

- Mô hình C. Ernest*



# Một số mô hình kiến trúc XPS

- *Mô hình E. V. Popov*





# Biểu diễn tri thức trong các XPS

- Tri thức của XPS có thể được biểu diễn theo nhiều cách khác nhau, ví dụ:
  - Biểu diễn tri thức bởi các luật sản xuất
  - Biểu diễn tri thức nhờ mệnh đề logic
  - Biểu diễn tri thức nhờ mạng ngữ nghĩa
  - Biểu diễn tri thức nhờ ngôn ngữ nhân tạo
- Ngoài ra, cũng có thể sử dụng cách biểu diễn tri thức nhờ các sự kiện không chắc chắn, nhờ bộ ba: đối tượng, thuộc tính và giá trị (O-A-V: Object-Attribute-Value), nhờ khung (frame), v.v...
- Tùy theo từng hệ chuyên gia, người ta có thể sử dụng một cách hoặc đồng thời cả nhiều cách.

# Các bài toán phù hợp với giải pháp XPS

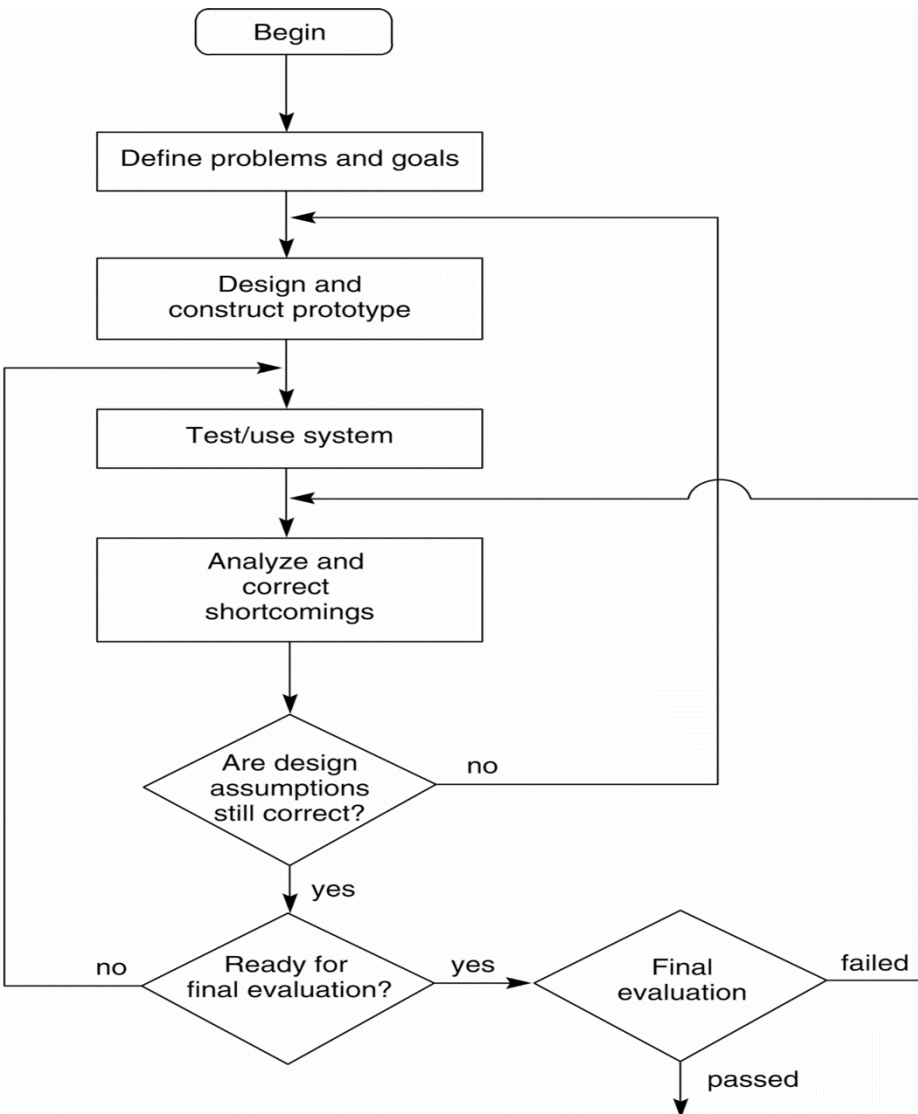
1. Sự cần thiết của một giải pháp biện minh cho chi phí và sức lực của việc xây dựng XPS.
2. Tri thức chuyên môn không sẵn sàng ở những nơi cần đến nó.
3. Vấn đề có thể được giải quyết bằng cách sử dụng các kỹ thuật suy luận ký hiệu
4. Vấn đề được cấu trúc tốt và không đòi hỏi sự suy luận theo lẽ thường.
5. Vấn đề có thể không giải quyết được bằng cách sử dụng các phương pháp tính toán truyền thống.
6. Có cơ sở hợp tác và hiểu ý nhau giữa các chuyên gia.
7. Vấn đề có kích cỡ và quy mô vừa phải.

# Quy trình công nghệ tri thức (knowledge Engineering)

- Ba đối tượng liên quan:
  - Kỹ sư tri thức (knowledge engineer): là các chuyên gia về ngôn ngữ và biểu diễn trong trí tuệ nhân tạo.
  - Chuyên gia (domain expert): là những người làm việc trong lĩnh vực chuyên môn và hiểu các phương pháp giải quyết vấn đề trong lĩnh vực đó.
  - Người sử dụng (end user): là những người xác định các ràng buộc thiết kế chủ yếu.
- Quá trình xây dựng XPS đòi hỏi một chu trình phát triển theo kiểu không truyền thống dựa trên các bản mẫu ban đầu và sửa lại chương trình với mức độ tăng dần => phương pháp lập trình thăm dò

# Quy trình công nghệ tri thức

Chu trình phát triển theo kiểu  
thăm dò



# Mô hình khái niệm & việc tích lũy tri thức

---

- Các khó khăn trong việc tích lũy tri thức:
  - Các kỹ năng của con người thường dựa trên thực nghiệm.
  - Tri thức của con người là “biết làm thế nào”
  - Tri thức của con người không căn cứ theo sự thật.
  - Tri thức luôn luôn thay đổi.

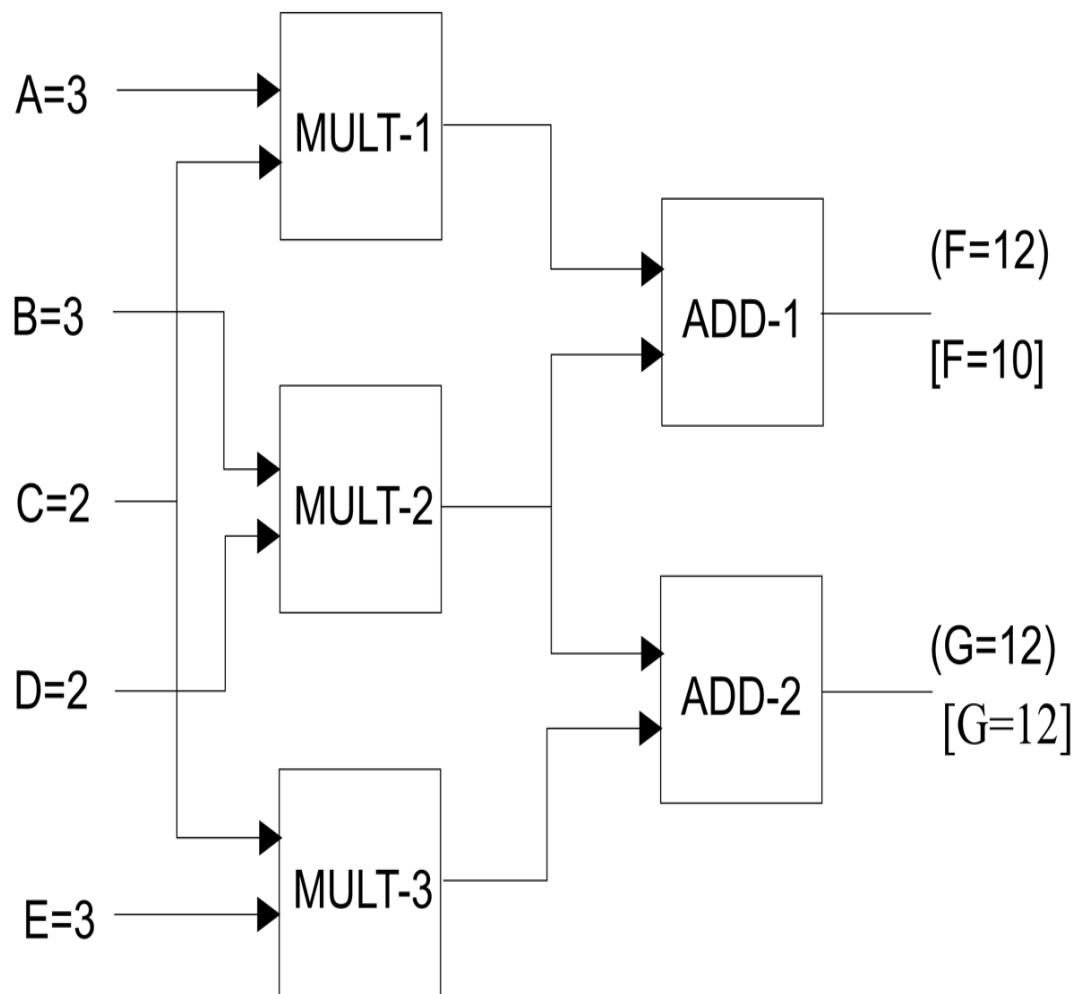
# XPS dựa trên mô hình

- là một hệ thống mà sự phân tích căn cứ trực tiếp trên sự mô tả chi tiết và chức năng của một hệ thống vật lý.
  - Ứng dụng: trong mục đích giảng dạy (mô hình của các thiết bị vật lý như mạch điện), các hệ thống tìm lỗi,...
  - Một hệ thống chẩn đoán dựa trên mô hình đòi hỏi:
    - Mô tả từng thành phần của một thiết bị => khả năng mô phỏng chức năng của từng thành phần
    - Mô tả cấu trúc bên trong của một thiết bị, thường là các thành phần và sự liên kết bên trong của chúng => khả năng mô phỏng sự tương tác giữa các thành phần.
    - Sự quan sát của việc thực hiện thật sự của thiết bị, ví dụ do các thông số vào/ra
- => Sự xác định lỗi thông qua việc giải thích sự khác biệt giữa các hành vi thật sự và hành vi mong đợi của thiết bị

# Ví dụ: định vị nơi gây lỗi

- Thực hiện 3 bước:
  - Tạo ra giả thuyết
  - Kiểm tra giả thuyết
  - Loại dần giả thuyết

Giới hạn: Chương trình hoạt động trên giả thuyết là hệ thống vật lý này chỉ có một lỗi



# Ưu điểm của XPS dựa trên mô hình

- Tạo khả năng sử dụng hiểu biết về cấu trúc và chức năng của vấn đề để giải quyết vấn đề.
- Vượt qua hạn chế của XPS dựa trên luật, XPS này có khuynh hướng mạnh, “khó vỡ”.
- Một số tri thức có thể chuyển tải cho các công việc khác.
- Có khả năng cung cấp các lời giải thích chỉ rõ nguyên nhân gây lỗi.



# Khuyết điểm của XPS dựa trên mô hình

- Mô hình chỉ là một mô hình nghĩa là một sự trừu tượng của hệ thống, vì vậy ở một mức độ chi tiết nào đó có thể không đúng (vd: tình trạng của đầu vào dữ liệu).
- Có một sự giả thiết ngầm hiểu về thế giới đóng => những gì không nằm trong mô hình có nghĩa là không tồn tại.
- Đòi hỏi một mô hình lý thuyết rõ ràng => việc tích lũy tri thức có thể gặp nhiều khó khăn, khó đạt được mô hình tốt, có khi là không tồn tại
- Hệ thống tạo ra có thể lớn và chậm

Tuy vậy, XPS dựa trên mô hình là một bổ sung quan trọng vào các gói phần mềm công nghệ tri thức, đặc biệt trong lĩnh vực chẩn đoán.

# XPS dựa trên trường hợp

- XPS dựa trên trường hợp (Case-based Reasoning – CBR) sử dụng một CSDL riêng biệt chứa giải pháp của các trường hợp đã giải quyết, để dựa vào đó tìm kiếm giải pháp cho một trường hợp mới.
- Phương pháp này minh họa cách giải quyết vấn đề của các chuyên gia trong nhiều lĩnh vực: luật sư, lập trình viên, kiến trúc sư, sử gia ...
- Để giải quyết một vấn đề, một CBR phải:
  1. Truy vấn các trường hợp thích hợp từ bộ nhớ của nó, dựa vào *sự tương tự* của một số *đặc điểm nổi bật*.
  2. Sửa đổi trường hợp đó để có thể áp dụng trong tình huống hiện tại.
  3. Áp dụng trường hợp đã chuyển đổi vào bài toán mới.
  4. Lưu lại lời giải và kết quả của nó (thành công hay thất bại).

# Ưu điểm của XPS dựa trên trường hợp

- Khả năng lưu trữ một cách trực tiếp các tri thức có được => có thể loại bỏ việc tích lũy tri thức từ các chuyên gia.
- Cho phép rút ngắn thời gian suy luận.
- Tạo khả năng tự học của hệ thống: giúp hệ thống tránh lỗi cũ và tận dụng những thành công trong quá khứ
- Việc phân tích tri thức của lĩnh vực chỉ diễn ra một lần, đó là khi tìm kiếm một sự biểu diễn hợp lý cho các trường hợp.
- Việc tích lũy tri thức và lập trình là tương đối đơn giản.
- Các chiến lược sắp xếp (index) thích hợp làm tăng sức mạnh của phương pháp này.

# Khuyết điểm của XPS dựa trên trường hợp

- Các trường hợp không thể hiện tri thức sâu về lĩnh vực bài toán  $\Rightarrow$  khó giải thích tại sao đưa ra lời giải như vậy, hoặc có thể đưa ra lời giải sai hoặc không tốt.
- Một cơ sở chứa các trường hợp lớn phải xem xét sự tương xứng giữa tính toán và lưu trữ.
- Khó đưa ra tiêu chuẩn đánh giá sự tương tự của các trường hợp, và sắp xếp chúng.

Q&A

---