



Chương 6: Hệ thống dựa trên luật (Rule-Based Systems)

Nội dung

- Suy diễn tiến
- Suy diễn lùi
- Hệ luật sinh

Hệ thống dựa trên luật

- Thay vì biểu diễn tri thức theo cách khai báo thông thường, hệ thống dựa trên luật biểu diễn tri thức dưới dạng các luật, tư vấn những điều nên làm hoặc đưa ra các kết luận cho những tình huống cụ thể
- Một hệ thống dựa trên luật bao gồm một tập các luật IF-THEN, các sự kiện, và trình biên dịch điều khiển việc áp dụng các luật, và các sự kiện được cho.

Hệ thống dựa trên luật

- Các hệ thống dựa trên luật:
 - Suy diễn tiến (forward chaining)
 - Suy diễn lùi (backward chaining)

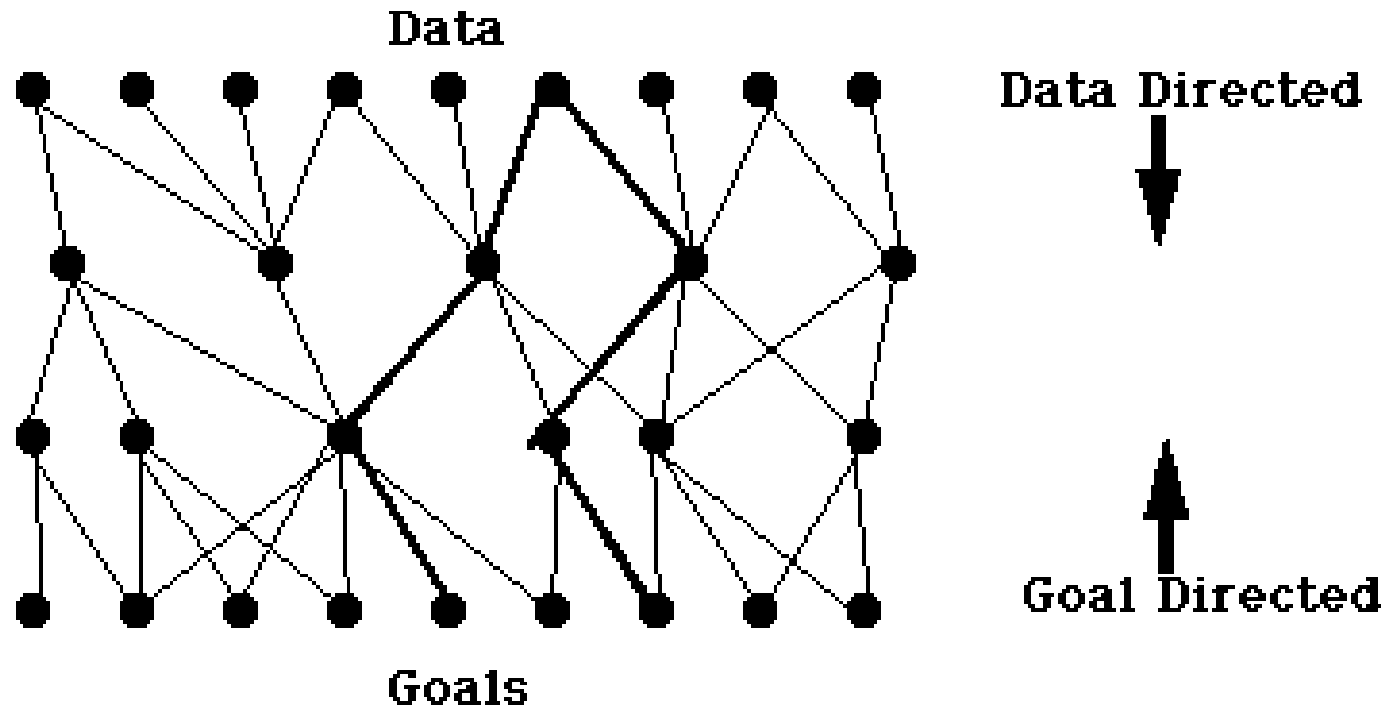
Suy diễn tiến (forward chaining)

- Việc tìm kiếm đi từ dữ liệu đến mục tiêu
 - tiến trình giải bài toán bắt đầu bằng các sự kiện cho trước + một tập các luật hợp thức dùng thay đổi trạng thái hiện tại
 - Quá trình tìm kiếm được thực hiện bằng cách áp dụng các luật vào các sự kiện để tạo ra các sự kiện mới, sau đó các sự kiện mới này lại được áp dụng các luật để sinh ra các sự kiện mới hơn cho đến khi tìm ra giải pháp cho bài toán
- Còn gọi là *tìm kiếm hướng từ dữ liệu* (data-driven search)

Suy diễn lùi (backward chaining)

- Việc tìm kiếm đi từ mục tiêu trở về dữ liệu
 - bắt đầu từ mục tiêu của bài toán, khảo sát xem có thể dùng những luật hợp thức nào để đạt đến mục tiêu này => xác định các điều kiện nào phải được thỏa mãn
 - Các điều kiện này sẽ trở nên những mục tiêu mới, hay còn gọi là các mục tiêu phụ (subgoals) trong tiến trình tìm kiếm.
 - Quá trình tìm kiếm cứ tiếp tục như thế, hoạt động theo chiều ngược, qua hết các đích phụ kế tiếp nhau cho đến khi gặp các sự kiện thực của bài toán.
- còn được gọi là *tìm kiếm hướng từ mục tiêu* (goal-driven-reasoning)

Suy diễn tiến vs. Suy diễn lùi



Ví dụ

- Xác định phương pháp tìm kiếm hướng dữ liệu hay tìm kiếm hướng mục tiêu là thích hợp hơn trong việc giải quyết các vấn đề dưới đây. Giải thích sự lựa chọn của bạn :
 - Chẩn đoán các trục trặc về thiết bị trong một xe ô tô.
suy diễn lùi
 - Bạn gặp một người cho biết là anh em họ xa với bạn, có cùng một ông tổ tên là John. Bạn cần kiểm tra lại thông tin này.
kết hợp suy diễn tiến & lùi
 - Một chứng minh định lý trong hình học phẳng.
suy diễn lùi

Ví dụ

- Một chương trình dùng để kiểm tra và diễn giải kết quả đọc của máy dò đường biển bằng sóng phản âm, chẳng hạn dùng thông báo cho một tàu ngầm lớn biết sắp có một tàu ngầm nhỏ, một con cá voi hay một đàn cá ở cự ly nào đó.

suy diễn tiến

- Một hệ chuyên gia giúp cho con người phân loại cây trồng theo đặc tính, chủng loại...

suy diễn tiến

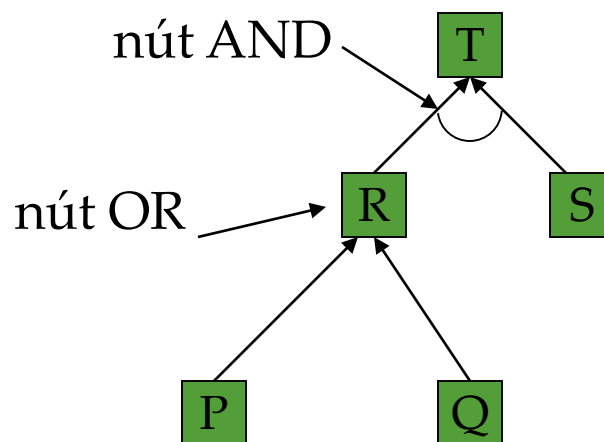
Đồ thị AND/ OR

- Một tập hợp các mệnh đề/ câu vị từ tạo thành một đồ thị AND/OR
- Đồ thị AND/ OR có hai loại nút trong:
 - Các nút AND
 - Các nút OR

Ví dụ:

$$P \vee Q \Rightarrow R$$

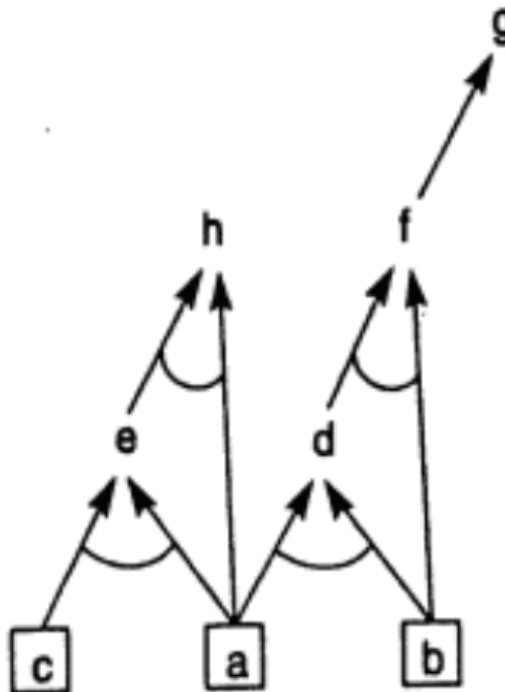
$$R \wedge S \Rightarrow T$$



Ví dụ

Giả sử các mệnh đề sau đây là đúng

- a
- b
- c
- $a \wedge b \Rightarrow d$
- $a \wedge c \Rightarrow e$
- $b \wedge d \Rightarrow f$
- $f \Rightarrow g$
- $a \wedge e \Rightarrow h$.



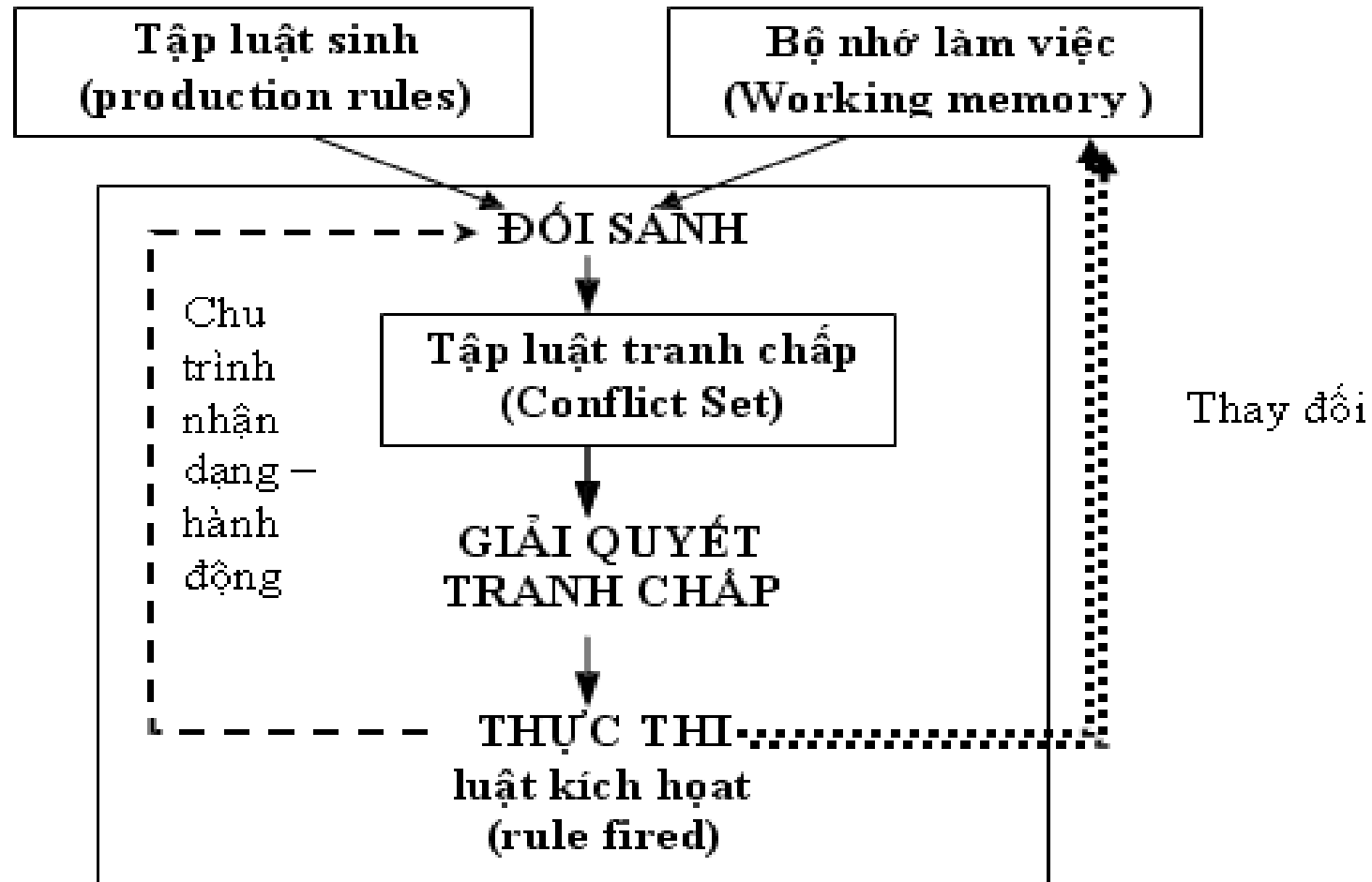
1. h đúng?

2. h còn đúng nếu b sai?

Hệ thống luật sinh (1)

- Hệ thống luật sinh (hệ sinh) bao gồm:
 - **Tập luật sinh (Production rules):** Mỗi luật sinh có dạng:
condition → action
 - **Bộ nhớ làm việc (Working memory):** Chứa một mô tả về trạng thái hiện thời của bài toán trong quá trình suy luận.
 - **Chu trình nhận dạng – hành động (Recognize – act cycle):** là cấu trúc điều khiển của hệ sinh.

Hệ thống luật sinh (2)



Cấu trúc của hệ sinh

Hệ thống luật sinh (3)

- Một số khái niệm:
 - *Luật khả thi (enable rule)*: là luật có điều kiện phù hợp với mẫu trong bộ nhớ làm việc
 - *Tập luật tranh chấp (conflict set)*: là tập hợp tất cả các luật khả thi.
 - *Giải quyết tranh chấp (conflict resolution)*: chọn một luật từ tập luật tranh chấp để kích hoạt

Một số ví dụ về hệ sinh

VD 1: Hệ sinh đơn giản dùng để sắp xếp một dãy các chữ cái a, b, c theo thứ tự từ điển

Với tập các quy luật như sau, anh/chị hãy sắp xếp dãy **cbaca** theo thứ tự từ điển:

1. ba \rightarrow ab
2. ca \rightarrow ac
3. cb \rightarrow bc

Một số ví dụ về hệ sinh

VD 1: Hệ sinh đơn giản dùng để sắp xếp một dãy các chữ cái a, b, c theo thứ tự từ điển

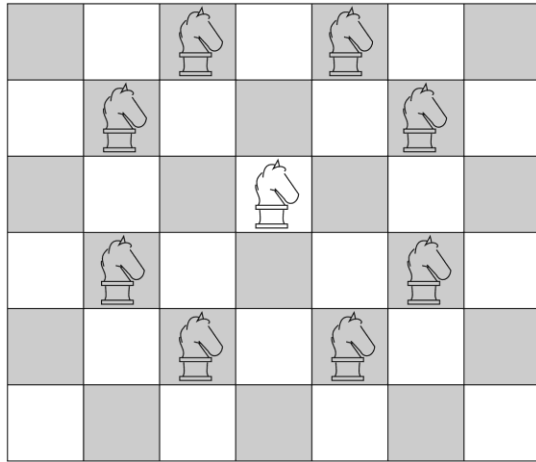
Production set:

1. $ba \rightarrow ab$
2. $ca \rightarrow ac$
3. $cb \rightarrow bc$

Iteration #	Working memory	Conflict set	Rule fired
0	cbaca	1, 2, 3	1
1	cabca	2	2
2	acbca	2, 3	2
3	acbac	1, 3	1
4	acabc	2	2
5	aacbc	3	3
6	aabcc	\emptyset	Halt

lặp đến khi mẫu trong bộ nhớ làm việc không còn khớp với điều kiện của bất kỳ luật sinh nào.

VD 2: Bài toán đường đi quân mã (1)



Các bước đi hợp lệ
của quân mã

1	2	3
4	5	6
7	8	9

Đánh số các ô trong
bàn cờ 3x3

RULE#	CONDITION	ACTION
1	Quân mã ở ô 1	Di chuyển đến ô 8
2	Quân mã ở ô 1	Di chuyển đến ô 6
3	Quân mã ở ô 2	Di chuyển đến ô 9
4	Quân mã ở ô 2	Di chuyển đến ô 7
5	Quân mã ở ô 3	Di chuyển đến ô 4
6	Quân mã ở ô 3	Di chuyển đến ô 8
7	Quân mã ở ô 4	Di chuyển đến ô 9
8	Quân mã ở ô 4	Di chuyển đến ô 3
9	Quân mã ở ô 6	Di chuyển đến ô 1
10	Quân mã ở ô 6	Di chuyển đến ô 7
11	Quân mã ở ô 7	Di chuyển đến ô 2
12	Quân mã ở ô 7	Di chuyển đến ô 6
13	Quân mã ở ô 8	Di chuyển đến ô 3
14	Quân mã ở ô 8	Di chuyển đến ô 1
15	Quân mã ở ô 9	Di chuyển đến ô 2
16	Quân mã ở ô 9	Di chuyển đến ô 4

VD 2: Bài toán đường đi quân mã (2)

- **Cơ chế điều khiển:** Áp dụng luật đầu tiên tìm được mà không tạo vòng lặp (không đi lại ô đã đi qua), cho phép quay lui
- VD: Tìm 1 con đường để quân mã đi từ vị trí 1 đến vị trí 2

Iteration #	Working memory		Conflict set (rule #'s)	Fire rule
	Current square	Goal square		
0	1	2	1, 2	1
1	8	2	13, 14	13
2	3	2	5, 6	5
3	4	2	7, 8	7
4	9	2	15, 16	15
5	2	2		Halt

1	2	3
4	5	6
7	8	9

Điều khiển tìm kiếm trong các hệ sinh

- Điều khiển chọn chiến lược tìm kiếm hướng từ dữ liệu hay hướng từ mục tiêu
- Điều khiển tìm kiếm thông qua cấu trúc luật
- Điều khiển chọn chiến lược giải quyết tranh chấp

Tìm kiếm hướng từ dữ liệu

- Tìm kiếm hướng từ dữ liệu trong một hệ sinh

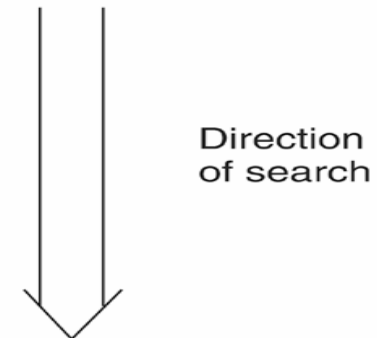
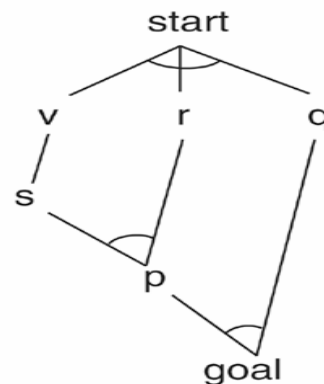
Production set:

1. $p \wedge q \rightarrow \text{goal}$
2. $r \wedge s \rightarrow p$
3. $w \wedge r \rightarrow q$
4. $t \wedge u \rightarrow q$
5. $v \rightarrow s$
6. $\text{start} \rightarrow v \wedge r \wedge q$

Trace of execution:

Iteration #	Working memory	Conflict set	Rule fired
0	start	6	6
1	start, v, r, q	6, 5	5
2	start, v, r, q, s	6, 5, 2	2
3	start, v, r, q, s, p	6, 5, 2, 1	1
4	start, v, r, q, s, p, goal	6, 5, 2, 1	halt

Space searched by execution:



Tìm kiếm hướng từ mục tiêu

- Tìm kiếm hướng từ mục tiêu trong một hệ sinh

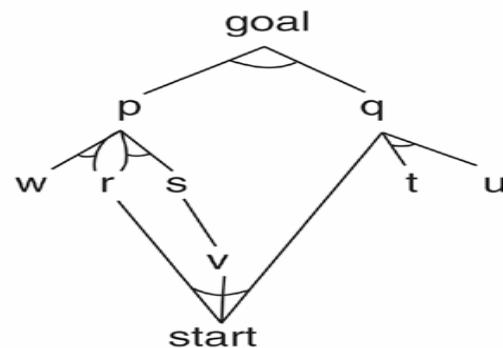
Production set:

1. $p \wedge q \rightarrow \text{goal}$
2. $r \wedge s \rightarrow p$
3. $w \wedge r \rightarrow p$
4. $t \wedge u \rightarrow q$
5. $v \rightarrow s$
6. $\text{start} \rightarrow v \wedge r \wedge q$

Trace of execution:

Iteration #	Working memory	Conflict set	Rule fired
0	goal	1	1
1	goal, p, q	1, 2, 3, 4	2
2	goal, p, q, r, s	1, 2, 3, 4, 5	3
3	goal, p, q, r, s, w	1, 2, 3, 4, 5	4
4	goal, p, q, r, s, w, t, u	1, 2, 3, 4, 5	5
5	goal, p, q, r, s, w, t, u, v	1, 2, 3, 4, 5, 6	6
6	goal, p, q, r, s, w, t, u, v, start	1, 2, 3, 4, 5, 6	halt

Space searched by execution:



Direction
of search

Tìm kiếm thông qua cấu trúc luật

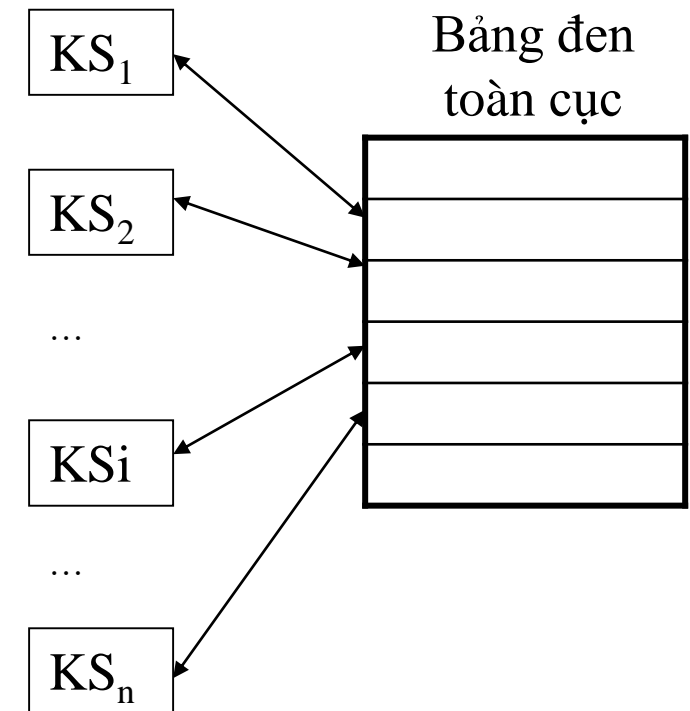
- Cấu trúc luật trong hệ sinh: điều kiện, hành động và thứ tự mà các điều kiện được thử
- Hệ sinh thử các luật theo một thứ tự xác định => quá trình tìm kiếm phụ thuộc cấu trúc và thứ tự của các luật trong tập luật sinh.
- Vận dụng heuristic

Điều khiển chọn chiến lược giải quyết tranh chấp

- *Khúc xạ (refraction)*: một luật không thể được kích hoạt lại khi các phần tử trong bộ nhớ làm việc chưa thay đổi => ngăn vòng lặp
- *Vừa mới xảy ra (recency)*: chọn những luật có phần điều kiện phù hợp với các mẫu vừa được bổ sung vào bộ nhớ làm việc => tập trung việc tìm kiếm vào một dòng suy luận duy nhất
- *Tính chi tiết (specificity)*: chọn luật giải toán chi tiết hơn (luật chi tiết là luật có nhiều điều kiện hơn)
- *Thứ tự lưu trữ các luật*: chọn luật khả thi đầu tiên.

Kiến trúc bảng đen (1)

- Bộ nhớ làm việc được sắp xếp thành các module riêng, mỗi module tương ứng với một tập luật sinh khác nhau => **kiến trúc bảng đen**
- Một kiến trúc bảng đen bao gồm:
 - **Bảng đen (blackboard):** là một cơ sở dữ liệu toàn cục tập trung
 - **Các nguồn tri thức (Knowledge Source):** độc lập, không đồng bộ, giao tiếp với nhau thông qua bảng đen.
 - **Bộ lập lịch trình (Scheduler):** tổ chức việc cấp tài nguyên và truy cập bảng đen của KS_i
- Kiến trúc bảng đen dùng trong việc tổ chức một chương trình lớn. VD: *HEARSAY-II*, là 1 chương trình hiểu lời nói



Kiến trúc bảng đen (2)

Ưu điểm của kiến trúc bảng đen:

- Mở rộng của các hệ thống luật sinh: cho phép tổ chức bộ nhớ làm việc vào các module riêng, mỗi module tương ứng với các tập con luật sinh khác nhau.
- Cho phép tổ chức và phối hợp nhiều chương trình giải quyết vấn đề trong một cấu trúc toàn cục duy nhất.
- Thích hợp cho việc thực thi chương trình trong một môi trường tính toán phân tán, đa bộ xử lý.
- Hỗ trợ mạnh mẽ cho những ứng dụng tri thức có cấu trúc cao

Bài tập nhóm

- Nhóm từ 3 – 4 SV
- Chọn 1 trong các chủ đề:
 - Ứng dụng tìm kiếm có thông tin và không có thông tin (bài toán tìm đường đi trong mê cung; bài toán người nông dân và sói, dê, bắp cải)
 - Sử dụng mạng ngữ nghĩa để giải quyết vấn đề (VD: bài toán tam giác)
 - Ứng dụng giải thuật thỏa mãn ràng buộc để giải quyết bài toán 8 hậu, Sudoku
 - Xây dựng hệ chuyên gia hỗ trợ chẩn đoán bệnh, dự báo thời tiết