

## Les systèmes experts

**Un constat :** Les experts d'un domaine ont pour caractéristiques :

- d'être rares, donc peu disponibles
- d'être compétents (si possible les meilleurs)
- d'être souvent incapables d'expliquer leur démarche
- d'être mortels

**Les objectifs des SE** • rendre une expertise accessible à tous

- approcher au mieux la perfection
- décortiquer le raisonnement expert pour expliquer
- rendre une connaissance experte insensible au temps

Simuler le comportement d'un expert humain sans avoir les défauts de la nature humaine.

**Les principales difficultés : comment**

- programmer une machine pour en faire un expert
- gérer les contradictions éventuelles entre différents experts
- faire évoluer un tel système sans que ses compétences se dégradent (trop-plein de connaissance)

**les Systèmes Experts (SE) :** L'un des succès industriels de l'Intelligence Artificielle. C'est un domaine interdisciplinaire : Psychologie cognitive, Neurobiologie, Sociologie, etc...

**Les 1ers travaux des SE :** Le système DENDRAL, en 1965 (identification en chimie)

**Le premier système expert digne de ce nom :** Le système MYCIN, en 1974 (maladies du sang) :

**Objectif** Aide au diagnostic dans le domaine des maladies infectieuses, en particulier méningites et maladies du sang.

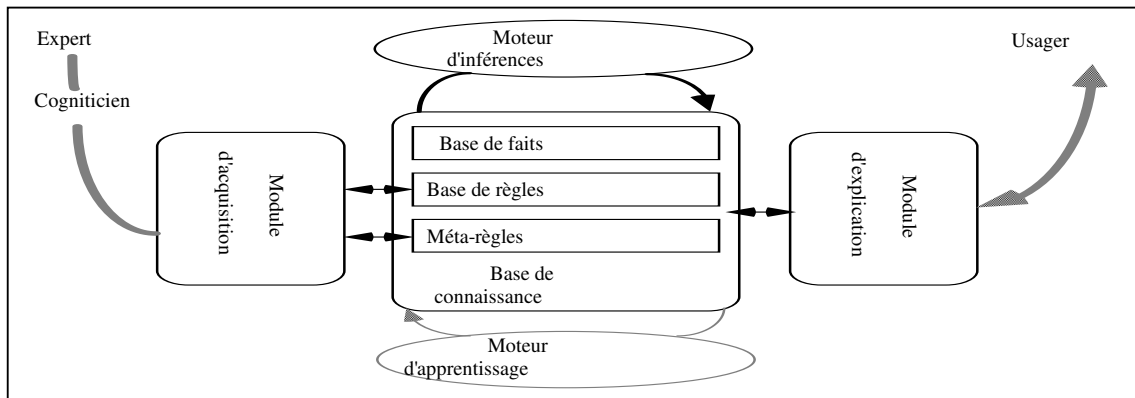
**Exemple d'opération ajoutant des informations**

SI le site de la culture est le sang  
ET l'identité de l'organisme n'est pas connue avec certitude  
ET la coloration de l'organisme est GRAM négatif  
ET la morphologie de l'organisme est bâtonnet  
ET le patient a été sérieusement brûlé  
ALORS l'identité de l'organisme est Pseudomonas (0.4)

**Remarque :** Les informations peuvent être incertaines

**Caractéristiques d'un SE :**

- Un système dans lequel la connaissance est facilement codée  $\Rightarrow$  Représentation de la connaissance
- Un système capable de faire des inférences  $\Rightarrow$  Mécanismes de raisonnement (déduction)
- Un système facilement modifiable  $\Rightarrow$  Evolution de la connaissance
- Un système capable d'expliquer son raisonnement  $\Rightarrow$  Etre convaincu de la justesse des résultats



SE de base = base de connaissance + moteur d'inférences

Base de connaissance = faits + règles

Moteur d'inférences : mécanisme de raisonnement, indépendant du domaine et du problème

Faits : mémoire à court terme, dépendant du problème

Règles : mémoire à long terme, indépendant du problème, dépendant du domaine

⇒ une règle manipule des faits

(Méta-règles : mémoire à long terme, indépendant du problème, voire du domaine

⇒ une méta-règle manipule des règles)

**Les Faits :** Toute description d'une situation implique des **entités** liées par des relations. Un fait représente au moins une relation liant diverses entités.

### Représentation centrée sur les entités

Un homme en général :

HOMME

est-un MAMMIFERE

mortel? (par défaut = oui)

Un philosophe en général :

PHILOSOPHE

est-un HOMME

nationalité (par défaut = GREC)

### Représentation centrée sur les relations

La relation est-un : est-un ( PHILOSOPHE, HOMME)

La relation nationalité : nationalité ( ALI, ALGERIEN )

La relation mortel? : est-un ( x, HOMME) mortel? ( x, oui )

### Représentation de la connaissance :

**Question :** comment représenter une connaissance afin :

- qu'elle puisse être exploitée le plus efficacement possible
- qu'elle occupe un espace mémoire le plus faible possible

**Réponse :** il n'y a pas de système de représentation miracle.

Il y a des systèmes de représentation, chacun adapté à une forme d'exploitation

### Les systèmes de représentation existants

Les réseaux sémantiques, les objets ⇒ proche de la pensée humaine (compréhension aisée)

La logique, les règles de production ⇒ proche des mécanismes de raisonnement

Les procédures ⇒ lié à notre conception actuelle des ordinateurs

## **- Les Règles de Production :**

**Définition :** Une règle de production est composée

- d'une partie prémisses (SI) = condition logique que doivent vérifier les faits
- d'une partie action (ALORS) = actions à opérer sur la base de faits

Les actions usuelles : ajouter, modifier, effacer des faits

Autres caractéristiques possibles d'une règle : coefficients de certitude, contexte d'application (groupe de règles), ...

**Stratégies de raisonnement :** Il existe 3 stratégies de raisonnement :

### **1) en Chainage avant :**

**Principe (démarche déductive)** on part de l'état initial et on cherche à atteindre le but.

entrée : une base de faits F, une base de règles R

sortie : la base de faits F transformée

algorithme :

**répéter**

S = ensemble des règles applicables de R (\*)

**si**  $S \neq \emptyset$

**alors** - « a » := choix d'une règle de S

- application de la règle choisie « a »

- marquer la règle « a »

**finsi**

**jusqu'à**  $S = \emptyset$

(\*) Une règle applicable est une règle :

- non marquée

- dont la prémisse est satisfaite par la base des faits

**Exemple :** Appliquer l'algorithme précédent à l'exemple suivant :

Base de faits : A C D F

Base de règles : R1 R2 R3 R4

R1 : si E C alors B

R2 : si A D alors E

R3 : si A B F alors G

R4 : si C D alors F

### **2) en chainage arrière :**

**Principe (démarche hypothético-déductive)**

entrée : une base de faits F, une base de règles R, un ensemble P de faits à prouver

sortie : la base de faits transformée F

algorithme :

**tantque**  $P \neq \emptyset$  **faire**

extraire (en l'effaçant) de P un fait f

**si** le fait f est absent de F

**alors** S = ensemble des règles concluant sur f

**si**  $S \neq \emptyset$

**alors** a = choix d'une règle de S

prouver tous les faits en prémisse de a

**si** a est applicable

**alors** appliquer la règle a

**finsi**

**finsi**

**finsi**

**fintant** (attention au problème du bouclage)

### **Exemple**

Base de faits : A C D F

Base de règles : R1 R2 R3 R4

R1 : si E C alors B

R2 : si A D alors E

R3 : si A B F alors G

R4 : si C D alors F

Faits à prouver : G F

### **3) Chainage mixte :**

Fonctionner en marche arrière

Dès qu'un fait nouveau est connu, fonctionner en marche avant

Poursuivre la marche arrière

### **Exemple**

Base de faits : A C D F

Base de règles : R1 R2 R3 R4

R1 : si E C alors B

R2 : si A D alors E

R3 : si A B F alors G

R4 : si C D alors F

Fait à prouver : G

**Stratégies :** Résoudre l'ensemble de conflits = choisir une règle parmi celles applicables

### **Quelques stratégies courantes**

prendre la première règle de l'ensemble

prendre la règle qui conclut sur le plus de faits

prendre la règle qui a le plus de conditions

essayer toutes les règles

prendre la règle qui travaille sur les faits les plus récents

prendre la règle ayant la prémisse de plus fort coefficient de certitude

**En général,** la stratégie est prédéfinie une fois pour toute.

Considérer que sélectionner une règle parmi d'autres est un problème en soi, qui nécessite une expertise. Autrement dit, utiliser un autre système expert pour opérer ce choix !

⇒ notion de méta-connaissance

### **Moteur d'inférences d'ordre 0**

Le fonctionnement du moteur d'inférences peut s'expliquer à l'aide de la logique des propositions :

- aucune notion de variable n'est autorisée
- les seules actions autorisées sont l'ajout et l'effacement de faits
- toute règle appliquée est éliminée

### **exemple**

si l'animal est un mammifère

et l'animal possède un pelage rayé de noir et de blanc

alors l'animal est un zèbre

### **Moteur d'inférences d'ordre 0+**

Moteur d'ordre 0 autorisé à modifier des faits (par exemple un compteur que l'on incrémente).

Une règle appliquée n'est pas nécessairement effacée.

**exemple**

si la température est  $< 30^{\circ}\text{C}$  et  
la consigne est  $\leq 50$  alors ajouter 2 à la consigne

**Moteur d'inférences d'ordre 1**

Le fonctionnement du moteur s'explique avec la logique des prédicats :

- notion de variable et d'appariement (unification)

Une règle appliquée reste toujours possiblement applicable

**Exemple**

si X est un homme alors X est mortel

**Remarque :** Conception de A à Z (programmation du moteur, etc...) de plus en plus rare...

**Utilisation de générateurs de SE (moteurs d'inférences nus)**

M1, OPS5, MP-LRO

**Usage de langages de programmation pour l'IA**

Langages fonctionnels : LISP, ML, CAML

Langages logiques : PROLOG

**Les systèmes multi experts :**

Pour disposer d'un SE ayant plusieurs domaines d'expertise, il suffit :

- d'un simple moteur d'inférences et d'indiquer dans les règles le domaine dont elles dépendent
- de plusieurs SE, chacun expert dans un domaine, contrôlés par un SE qui supervise le tout et activera le bon SE

**Des propositions étudiées dans les laboratoires**

- Les tableaux noirs

La base de faits est une mémoire partagée par divers experts (sources de connaissance)

Chaque expert réagit dès qu'il y a une modification de la base qui le concerne

Chaque expert n'agit que sur une partie de la base

⇒ systèmes multi-agents

⇒ vie artificielle

**Collaboration cogniticien-expert**

Interview de l'expert par le cogniticien

Codage par le cogniticien de la connaissance extraite

Confrontation du système avec l'expert

**Apprentissage automatique**

Donnée par l'expert d'une série de problèmes résolus

Induction par la machine d'une base de règles capable de retrouver les solutions proposées aux problèmes de l'expert

Confrontation du système avec l'expert

**Les principales difficultés**

Trouver une stratégie d'extraction de connaissance qui soit efficace et universelle

Vérifier la cohérence de la base de connaissance

### **Conclusion provisoire**

L'expert est :

- au début, enchanté : explicitation de sa connaissance (il apprend lui-même de l'analyse de son propre savoir ainsi mis à plat)
- ensuite, inquiet : la compétence du SE semble réelle
- enfin, rassuré : il existe toujours des problèmes que le SE ne sait pas résoudre, malgré les multiples révisions de la base de connaissance

### **Quelques applications :**

Chimie (DENDRAL)      Géologie (PROSPECTOR)      Médecine (MYCIN) Diagnostic de pannes