

REPRÉSENTATION DES CONNAISSANCES¹

Par Gilbert Paquette

SOMMAIRE

Introduction	1
1. Connaissances et fonctionnement cérébral.....	1
1.1 Modèle de traitement de l'information chez l'humain.....	2
1.2 Mémoire à court terme.....	4
1.3 Mémoire à long terme	4
1.4 Exemple de représentation et de traitement.....	5
2. Divers types de connaissances.....	7
2.1 Quelques exemples de domaines de connaissances	7
2.2 Connaissances déclaratives.....	9
2.3 Connaissances procédurales	11
2.4 Connaissances conceptuelles.....	13
3. Modèles pour la représentation des connaissances.....	14
3.1 Le langage graphique de modélisation par objets typés (MOT)	14
3.2 Les triplets objet-attribut-valeur.....	20
3.3 Modèles à base de règles (non structurés).....	21
3.4 Modèles à base de schémas (structurés).....	25
3.5 Avantages et limites des modèles présentés	29
Conclusion.....	30
À retenir.....	31

¹ Certains exemples sont extraits de G. Paquette et L. Roy, « Systèmes à base de connaissances », Beauchemin et Télé-université, pp 61-120

Introduction

Dans un premier texte, nous avons mis en évidence un fait important :

Pour qu'une machine effectue « intelligemment » une tâche, il faut lui fournir une représentation des connaissances qu'elle devra exploiter.

Le modèle des connaissances constitue la partie la plus significative d'un programme d'intelligence artificielle (IA). On peut dès lors déduire que la performance d'un programme est surtout déterminée par la pertinence, l'exactitude et la généralité du modèle que l'on peut définir. Plus un modèle est approprié, plus le programme qui l'utilise est performant. On appelle modélisation le procédé d'analyse par lequel on peut définir les données et en concevoir une *représentation* que l'on pourra soumettre à un programme. Or, dans le domaine de l'IA, les données ont un caractère général : elles décrivent l'ensemble des connaissances que doit acquérir une machine pour exécuter convenablement une grande variété de tâches. C'est pour ces raisons que le processus d'analyse des données d'un programme d'IA est appelé la *représentation des connaissances*.

Pour représenter des connaissances, il faut savoir ce qu'est une connaissance, c'est-à-dire de quoi elle est constituée. On admet généralement que, pour connaître un sujet, il faut identifier les objets qui le décrivent et les relations entre ces objets. Nous venons de nommer les deux éléments essentiels de tous les modèles de savoir : *les objets et les relations*. Le terme objet désigne une entité réelle ou abstraite qui fait partie du domaine à l'étude. C'est le contexte de son utilisation qui lui donnera une signification.

Dans ce deuxième texte, nous présentons un modèle de traitement de l'information chez l'humain pour montrer comment on peut classer et représenter les connaissances en mémoire. Puis, nous suggérons une classification des connaissances selon leur niveau de généralité et la manière de les exprimer et de les utiliser. Ensuite, nous introduisons quelques outils de représentation des connaissances. Parmi ces outils, figurent des modèles comme les règles et les schémas. Ils permettent la codification standardisée des connaissances en vue de leur traitement informatique. Enfin, on fera un survol des outils de développement informatique du point de vue des représentations qu'ils supportent.

1. Connaissances et fonctionnement cérébral

En informatique, nous implantons dans un ordinateur des connaissances d'un sujet donné en vue de les utiliser pour résoudre des problèmes. Pour y parvenir, on s'interroge d'abord sur la façon d'organiser les connaissances pour que l'ordinateur les exploite correctement. Mais l'organisation des connaissances renvoie directement à la manière de les mémoriser pour pouvoir ensuite les utiliser. Cela nous amène à considérer le rôle que joue la mémoire dans un processus de traitement des connaissances.

1.1 Modèle de traitement de l'information chez l'humain

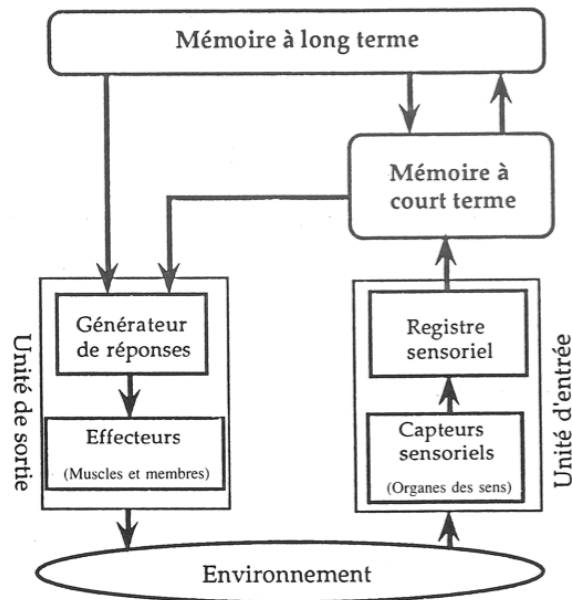


Figure 1 – Un modèle de la mémoire en psychologie cognitive

Pour mieux comprendre, examinons le fonctionnement des mémoires dans un cerveau humain. Il s'agit d'une règle méthodologique générale en IA : s'inspirer du fonctionnement du cerveau humain pour concevoir des éléments de l'architecture des ordinateurs et inversement. Cette analogie nous permettra d'identifier différents types de mémoires et nous conduira à une classification des connaissances.

Nous présentons d'abord un modèle emprunté à la psychologie cognitive :

Le modèle de la figure 1 décrit les éléments essentiels de notre système de traitement de données qui nous permet d'interagir avec notre environnement. Outre la diversité de ses composantes, ce schéma met en évidence des transferts de données qui forment un cycle complet de traitement des informations dont l'environnement est le point de départ et l'aboutissement. Dans cet ensemble, les sens tiennent lieu d'« unité d'entrée » qui achemine les données aux mémoires, et les muscles sont l'« unité de sortie » qui reçoit les données traitées en mémoire et qui les convertit en action sur le milieu. Ce modèle présente des ressemblances frappantes avec le schéma fonctionnel d'un ordinateur. Il ne faut pas s'en étonner, car en IA, l'être humain et l'ordinateur sont considérés comme deux machines qui traitent de l'information.

Il manque cependant un élément très important à ce modèle : l'unité centrale de traitement. Il s'agit d'un ensemble de cellules très spécialisées qui reçoit des informations provenant des organes des sens. Ces informations sont acheminées par des faisceaux nerveux jusqu'au thalamus. Pour être transformées en action, les données reçues des sens sont traitées par le système nerveux de cette façon :

- Les impulsions qui produisent les « sensations » sont décodées, transformées en « structures de données » et entreposées dans les différentes zones de mémoire.
- Elles sont ensuite interprétées et on leur attribue une signification.
- Puis, elles sont extraites des mémoires, transformées en intentions et transmises aux muscles, de nouveau sous forme d'impulsions.

Tel que le propose le schéma de la figure 1, on reconnaît généralement l'existence de deux types distincts de mémoires : la mémoire à long terme (MLT) et la mémoire à court terme (MCT). On devine aisément à leurs noms que ces mémoires ont la faculté de retenir des données plus ou moins longtemps. De plus, leur contenu, leur taille et leurs fonctions diffèrent substantiellement. Voici comment ces mémoires interviennent dans un cycle complet de traitement d'information.

- Enregistrement sensoriel : des signaux sont captés par les sens et sont transmis au cerveau sous forme d'impulsions qui sont mémorisées pendant un laps de temps très court dans les registres sensoriels.
- Stockage temporaire dans la MCT : les impulsions sont décodées pour en extraire des éléments d'information élémentaire.
- *Construction d'une représentation.* et intégration au contenu de la MLT : il s'agit d'un traitement symbolique et d'une organisation des connaissances en réseau. C'est l'étape d'apprentissage.
- Production d'un plan d'action : les connaissances sont extraites de la MLT, traitées et traduites en intentions puis emmagasinées dans la MCT.
- Envoi d'ordre d'action aux organes effecteurs : le générateur de réponses récupère les données dans la MCT, les traduit en impulsions qu'il achemine par la suite aux muscles et aux membres chargés de l'action.

Ainsi, le cerveau est une machine qui *transforme* des données pour traduire des sensations reçues de l'environnement en action sur celui-ci. On pourrait même ajouter qu'il produit de nouvelles informations en établissant des liens entre des données déjà connues, mettant ainsi en œuvre des capacités évidentes de structuration, d'induction, de déduction, donc de création. Les transformations opérées par le cerveau sur des données permettent littéralement d'en changer la forme pour les rendre de plus en plus significatives à mesure qu'elles progressent dans le cycle de traitement.

Nous avons trois raisons de prétendre que le modèle proposé par la psychologie cognitive a un caractère scientifique. Premièrement, il est tout à fait compatible avec le modèle neurologique généralement admis. Celui-ci fait ressortir les faits suivants :

- le cerveau est un réseau de neurones;
- la boîte crânienne est le siège d'activités cérébrales détectables;
- les fonctions cérébrales sont localisées et très spécialisées.

En second lieu, le modèle cognitif rend compte des résultats de la psychologie expérimentale qui confirment que :

- le cerveau traite des « objets mentaux »;
- le cerveau construit des concepts, sorte de représentations abstraites des objets, et les classe en catégories.

Troisièmement, le modèle suggéré par la psychologie cognitive a un pouvoir explicatif et un pouvoir de prédiction. En effet, il permet de comprendre et d'expliquer les manifestations du traitement de l'information par un cerveau humain. On peut également l'appliquer pour anticiper et prédire les résultats d'expériences sur le comportement. Enfin, on a testé ce modèle dans des milliers d'expériences reproductibles dont on a contrôlé les paramètres. Voilà autant d'attributs d'une théorie scientifique générale.

1.2 Mémoire à court terme

Comme son nom l'indique, la mémoire à court terme (MCT) contient des informations éphémères qui sont directement reliées à un usage immédiat, d'où son autre appellation de mémoire de travail. On estime que la MCT peut mémoriser une donnée pendant quelque dix secondes. Son contenu est donc volatile et, pour être préservé, il doit être rafraîchi régulièrement en appliquant des techniques basées sur la répétition. D'autre part, le volume de la MCT semble fort restreint puisque l'on admet généralement que l'on peut y entreposer environ sept *éléments* différents.

Par ailleurs, on suppose son contenu diversifié, mais à faible niveau d'organisation, c'est-à-dire très peu structuré. En effet, selon des expériences maintes fois répétées en psychologie, la MCT pourrait contenir des éléments de formes aussi diverses qu'un nombre, une lettre, un nom de personne, un numéro de téléphone, un nom de lieu, une phrase courte, une couleur, une image, etc. Tous ces éléments sont caractérisés par la nature indivisible de leur contenu logique : les Européens les appellent « granules », alors que les Américains les nomment *chunks*. Les expériences tendent également à démontrer que les chances qu'un élément soit gardé ou non dans la MCT dépendent de son ordre d'entrée par rapport à d'autres éléments. En effet, on a observé que, lors de la mémorisation d'une séquence d'éléments, les premiers et les derniers éléments mémorisés étaient retenus plus facilement que ceux du milieu de la séquence. Lorsqu'un nouvel élément est introduit, les premiers « effacés » sont ceux formant le cœur d'une séquence mémorisée antérieurement. On reconnaît ici un phénomène typique de gestion d'une file d'attente. En somme :

La MCT contient environ sept éléments de formes diverses. La rétention d'un élément dépend de son ordre d'entrée dans la MCT.

Mentionnons enfin un type de mémoire très particulier que l'on pourrait qualifier de « mémoire à très court terme ». Il s'agit des registres sensoriels qui tiennent lieu de mémoire tampon. Ils se caractérisent par la nature très éphémère, voire instantanée, de leur contenu et par le caractère brut des données qu'ils contiennent. En effet, on estime que le temps de mémorisation des impulsions y est de l'ordre du quart de seconde. Ces registres sont la terminaison des faisceaux nerveux en provenance des sens. De plus, leur contenu n'est qu'une suite d'impulsions successives qui sont récupérées, décodées et traitées immédiatement pour être livrées à la mémoire à court terme.

1.3 Mémoire à long terme

La mémoire à long terme (MLT), comme son nom l'indique clairement, peut mémoriser des données pendant une longue période allant jusqu'à la vie entière d'une personne. Mais elle se distingue de la MCT par ces trois aspects fondamentaux : son volume, le type de données qu'elle contient et leur organisation ainsi que l'élaboration progressive de son contenu.

Les connaissances actuelles en neurologie font état de la situation suivante : l'organisation du cerveau est celle d'un réseau. Les nœuds de ce réseau forment un ensemble de quelque dix milliards de cellules nerveuses appelées neurones. On estime que chaque neurone est relié en moyenne à 10 000 autres par les « fils neuronaux » que sont les axones et les dendrites. Il serait hasardeux de prédire le contenu d'un seul neurone. Par contre, une observation des activités quotidiennes nous apprend que le cerveau peut mémoriser des données de formes aussi diverses que celles énumérées pour la MCT. En outre, la MLT comporte une particularité fondamentale : non seulement elle contient des données élémentaires, mais elle contient également des méthodes, c'est-à-dire des indications sur la manière d'exécuter une tâche. Ce sont des « recettes » ou des *heuristiques* qui décrivent, à la

manière d'un algorithme, les étapes d'exécution d'une opération. C'est ce qui permet de répéter machinalement une séquence de gestes et de découvrir le parcours conduisant à un but.

La MLT retiendrait donc deux types de données : l'un qui décrit les choses (*quoi*), l'autre qui explique leur fonctionnement (*comment*). Ce sont là deux aspects généraux et universels des connaissances. Nous y reviendrons. Quant à l'*organisation* des connaissances dans la MLT, on admet généralement qu'elles sont structurées en *réseau*. Ainsi, l'ensemble de nos connaissances est constitué d'éléments indivisibles entre lesquels sont établies des relations. On dit parfois que comprendre c'est établir des liens. Ce point de vue a d'ailleurs conduit à la notion de *mémoire associative* qui repose sur une hypothèse semblable d'organisation des connaissances en mémoire. On y suppose que le contenu de la mémoire est progressivement élaboré par associations de connaissances. En outre, des liens seraient établis sur la base d'affinités de connaissances pour construire un réseau de concepts. Toutes ces hypothèses sont compatibles avec l'organisation biologique du cerveau en un réseau de cellules nerveuses reliées entre elles par des axones et des dendrites

Un dernier aspect distingue la MLT de la MCT : son contenu est élaboré progressivement par tous les événements que nous vivons. Ainsi, nos expériences peuvent modifier certaines données qui nous étaient connues. C'est ce qui nous permet d'en enregistrer de nouvelles et d'en éliminer certaines autres devenues caduques ou fausses avec le temps. De plus, il arrive très souvent que nous établissions des rapports entre des informations récemment apprises et d'autres plus anciennes. Nous pouvons même déduire de nouveaux faits à partir des connaissances dont nous disposons. Ces quelques observations montrent que la MLT est un outil privilégié d'apprentissage. Il s'agit d'une faculté typique d'un comportement intelligent. En somme, on peut résumer les caractéristiques essentielles de la MLT comme suit :

La mémoire à long terme contient une très grande quantité de données qui sont mémorisées et mises à jour tout au long de la vie. Sa structure est celle d'un réseau. Elle contient des éléments d'information et des règles d'opération.

1.4 Exemple de représentation et de traitement

L'aspect dynamique des connaissances maintenues en mémoire à long terme est révélateur d'une capacité d'apprentissage. Nos connaissances sont régulièrement mises à jour et réorganisées au fil de nos expériences. On présume que des nouvelles connaissances sont incorporées au réseau des concepts existants. Il se peut que la mémorisation de nouvelles connaissances oblige à une réorganisation de celles déjà apprises. Cette hypothèse constitue la base d'un modèle de l'apprentissage très révélateur.

Nous allons illustrer ces aspects par un exemple concret. Supposons que l'étude et certaines expériences ont permis à un enfant de construire la représentation, à la figure 2, du concept *poisson*.

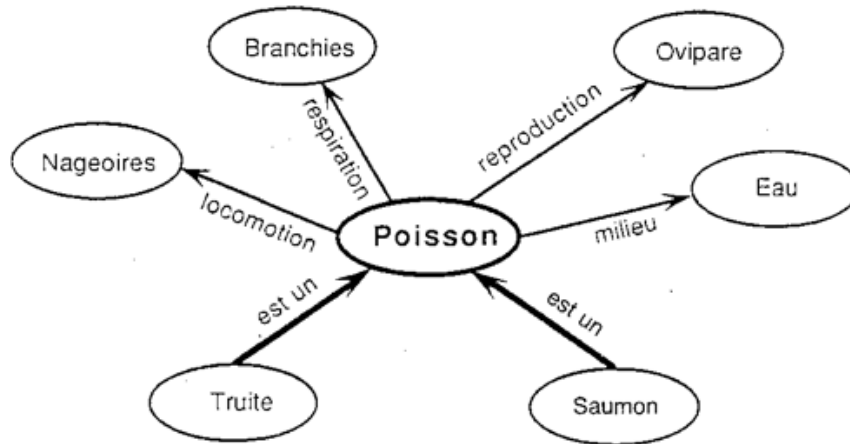


Figure 2 – Un réseau d’association dans la MTL pour le concept de poisson.

Supposons maintenant que notre jeune sujet vient d’apprendre l’existence de la baleine. Bien que son habitat naturel soit l’eau, la baleine est en fait un mammifère. La première classification des animaux ne tient plus. Pour intégrer la nouvelle connaissance *baleine*, l’enfant doit donc réorganiser la structure du concept *poisson* qu’il avait antérieurement construite. La figure 3 nous montre ce que pourrait être la nouvelle représentation du concept *animal marin*.

L’exemple du poisson et de la baleine illustre le fait que l’on doit traiter nos connaissances en mémoire pour apprendre. Dans ce cas, il a fallu changer des liens et créer de nouvelles classes pour modifier un concept. De plus, il fait ressortir les notions de classification *d’objets* et de leur organisation hiérarchique pour constituer un *concept*. Il s’agit d’une construction mentale sans laquelle il nous serait impossible de gérer la diversité des objets et des événements auxquels nous sommes confrontés dans une vie. C’est un outil de synthèse, de structuration et de généralisation des connaissances emmagasinées dans la mémoire à long terme.

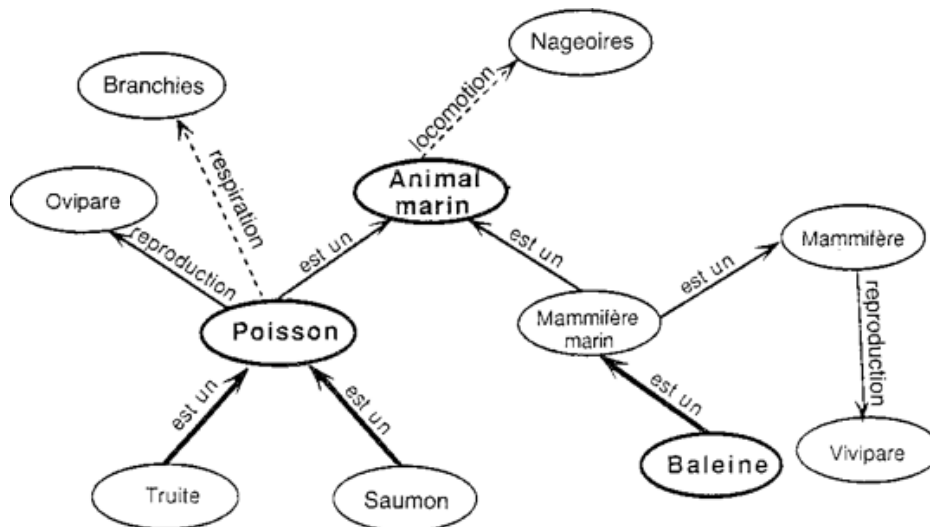


Figure 3 – Un réseau d’association du concept « animal marin » dérivé de celui du concept « poisson »

2. Divers types de connaissances

Nous venons de prendre conscience de la diversité de forme, d'organisation et de complexité des données traitées par un cerveau humain. Cela est probablement le résultat d'une longue adaptation à nos conditions d'existence. En effet, l'un des traits essentiels de notre environnement est la variété étonnante des sensations qu'il nous transmet. Pour s'y adapter, il faut le décoder en structures de données assez souples et établir des liens entre les formes diverses des signaux recueillis par les sens. C'est justement en structurant les données et en établissant des relations entre elles que l'on peut générer un large éventail de *significations* à leur attribuer selon le contexte. Or, l'ensemble des données mémorisées et structurées par une personne constitue justement son *bagage de connaissances*.

2.1 Quelques exemples de domaines de connaissances

Nous suggérons ici quelques contextes. Nous identifions la principale faculté mise en œuvre et nous déduisons le type de connaissances normalement invoquées.

Géographie

Dans ce domaine, les connaissances ont parfois l'aspect d'un ensemble de données brutes. Ces données sont relatives à des villes, des régions et des pays. Elles peuvent aussi nous renseigner sur la situation physique, la population, les particularités de l'économie, etc.

En voici un échantillon.

- Les coordonnées de Montréal sont : 45°30' de latitude nord et 73°36' de longitude ouest.
- La population du Québec était de 6 700 000 personnes en 1989; elle dépasse maintenant 8 millions.
- Les Chinois ont une espérance de vie moyenne de 68 ans.
- Les principales ressources naturelles du Mexique sont le pétrole, l'argent et le cuivre.
- Les Andes forment une chaîne de montagnes en Amérique du Sud.

On constate que toutes ces informations doivent être mémorisées pour être utilisées, à moins de pouvoir les déduire indirectement si l'on dispose de d'autres connaissances connexes. On dit que de telles données ont un caractère *déclaratif*. Elles sont généralement énoncées sous forme de propositions simples ayant un caractère particulier : les *faits*. Chaque fait est un élément de connaissances qui décrit un aspect particulier d'une réalité. Ce type de connaissances fait appel à notre faculté de *mémorisation*. De plus, un agencement « logique » peut grandement faciliter de les *remémorer* en vue de les utiliser.

Par contre, d'autres connaissances géographiques sont d'un niveau plus général. Par exemple, les phrases « Au Québec, l'agriculture est très développée dans la vallée du Saint-Laurent entre Montréal et Québec » ou « Les grandes villes sont généralement situées en bordure de la mer ou près d'une grande voie d'eau » sont des énoncés de portée générale. Elles sont applicables à un grand nombre d'objets : plusieurs régions du Québec ou plusieurs villes du monde. Ce type de connaissances appelées *règles* a un caractère général. Il fait appel à notre faculté d'induction de lois à partir d'exemples.

Cuisine

Dans un contexte comme celui-ci, les connaissances requises décrivent une série d'actions à exécuter. Ces connaissances ont principalement l'aspect de séquences ou de cycles d'opérations qui indiquent *comment* exécuter une tâche. À titre d'exemple, considérons la

recette d'un succulent gâteau chocolaté aux courgettes. Celle-ci commence par une liste d'ingrédients avec leur quantité. Ensuite, elle décrit comment faire le gâteau à partir des ingrédients.

« Mélanger la margarine, l'huile et le sucre. Ajouter les œufs, le lait sur et la vanille. Battre, tamiser les ingrédients secs et les ajouter au mélange crémeux. Bien battre, puis incorporer les courgettes coupées en dés. Verser dans un moule graissé et enfariné. Saupoudrer les brisures de chocolat sur le dessus. Cuire dans un four préchauffé à 160°C pendant 45 minutes. On peut napper le gâteau d'un glaçage à la vanille. »

Une telle expression des connaissances décrit un savoir-faire. Il s'agit d'une suite d'étapes, d'une marche à suivre pour exécuter une tâche. On peut également les exprimer sous une forme plus rigide : *si* une condition est vraie *alors* faire telle opération *sinon* faire une autre opération. Ce sont des opérations conditionnelles décrites sous forme de règles ou de connaissances procédurales. L'accumulation de ce type de connaissances constitue en quelque sorte *l'expérience* ou le savoir-faire d'une personne.

Écriture

Il s'agit d'une activité qui fait référence aux deux types de connaissances évoquées plus haut. Elle requiert un certain degré de « culture générale », tant sur le plan des connaissances brutes que sur celui de l'articulation de la pensée, pour donner des résultats intelligibles tout en respectant les règles de la langue. Les connaissances invoquées en écriture sont de trois niveaux : du vocabulaire, des règles de grammaire et des connaissances relatives au contexte sur lequel on veut écrire. Supposez que l'on vous demande de produire un texte sur l'organisation du cerveau. Vous devrez probablement rédiger à partir des connaissances suivantes :

Concepts : Cerveau, modèle, neurone, réseau, dendrite, axone, mémoire.

Règles de grammaire

- Pour écrire une phrase, il faut agencer correctement un groupe sujet et un groupe verbal.
- Le verbe s'accorde en personne et en nombre avec son sujet.
- Si le participe passé est conjugué avec le verbe avoir et placé après celui-ci, il s'accorde avec le complément direct lorsque celui-ci est placé avant le verbe.

Connaissances déclaratives

- Le cerveau est un réseau de dix milliards de neurones.
- Les dendrites sont la voie d'entrée au noyau du neurone.

Cet exemple fait intervenir un mélange de différents types de connaissances : des données mémorisées, l'habileté à les exprimer ainsi que les facultés de déduction et de synthèse qui confèrent un sens à l'ensemble de notre argumentation.

Notre but est de transmettre ces divers types de connaissances à un programme informatique. C'est pourquoi nous devons savoir les exprimer à l'aide de représentations précises et standardisées de la même manière que des types de données acceptés par un langage de programmation sont définis rigoureusement, voire rigidement. En somme, nous recherchons des formes pour encadrer les connaissances. Pour y arriver, nous commençons par une classification des divers types de connaissances.

2.2 Connaissances déclaratives

En général, on peut dire que les connaissances déclaratives sont exprimées à l'aide d'énoncés ou d'assertions. Chaque assertion constitue une entité qui traduit un aspect de la réalité. Contrairement aux méthodes qui indiquent comment faire des choses, les connaissances déclaratives ou assertions décrivent les choses, le *quoi*. Nous nous limiterons à exposer trois types d'assertions : les faits ou connaissances factuelles, les règles ou connaissances conditionnelles et les connaissances prédicatives.

Connaissances factuelles (faits)

C'est la forme la plus rudimentaire. Une connaissance factuelle ne rend compte que d'un fait sans fournir une explication. Il s'agit donc de données brutes qui doivent être traitées et intégrées à un réseau de données pour en déduire des informations significatives. En informatique, les connaissances factuelles sont les données qui constituent les fichiers ou les bases de données traités par des programmes. En voici quelques exemples :

- $1T = 3,1416$,
- Je demeure à Québec,
- Ce pantalon vaut 30 \$,
- Il fait 20°C ,
- Un poisson respire avec des branchies.

Par ces exemples, on voit qu'une connaissance factuelle se rapporte à un cas particulier : elle décrit un fait en rapport avec une seule chose une personne ou un événement. Enfin, une connaissance factuelle exprime une idée à propos d'une situation particulière.

Connaissances conditionnelles (règles)

Il s'agit d'une forme d'expression plus générale, car les connaissances conditionnelles peuvent être appliquées dans plusieurs situations. De plus, cette forme de connaissances permet d'exprimer deux réalités importantes :

- des relations entre des faits ou des idées,
- des actions, des opérations ou des méthodes.

De cette manière, on peut établir des relations entre des connaissances déclaratives et les rendre opérationnelles. Les connaissances conditionnelles ont un pouvoir déductif, car lorsqu'elles sont appliquées à un cas particulier, elles peuvent générer de nouveaux faits. Considérons les exemples suivants :

- Quand la température est froide, je porte un manteau.
- Si le revenu imposable est inférieur à 10 000 \$, le taux d'imposition est 20 %.
- La surface d'un rectangle est égale à sa base multipliée par sa hauteur.
- Si le patient est rouge, prendre sa pression artérielle.

Chacun de ces énoncés comporte deux faits : une condition et une conclusion. Pour traduire ces expressions dans un langage structuré, il faudra relier deux faits et déboucher sur une action. Or, une *règle* est un amalgame de ces deux éléments dans une même expression.

Ces exemples reflètent une particularité importante des règles : elles traduisent des *connaissances dynamiques*, c'est-à-dire qu'elles déclenchent un processus ou une transformation. Elles incorporent les connaissances factuelles. La partie action d'une règle

décrit comment s'effectue une opération. Mais le plus important, c'est que *les règles établissent des liens entre des connaissances*. Ceci est capital, si l'on veut tenir compte de la nature associative des connaissances. En conclusion, une règle exprime des faits reliés par les connecteurs *si* et *alors*. *Une règle c'est un ensemble de connaissances liées* et un outil d'organisation et de structuration des connaissances.

Connaissances prédicatives (ou relationnelles)

Il existe beaucoup d'autres formes d'énoncés qui ne sont pas des règles. Les connaissances prédicatives font appel à la notion de variable individuelle, de prédicat et de quantificateur. Prenons par exemple les énoncés suivants :

1. Platon est un philosophe.
2. Socrate est un philosophe.
3. Platon est un élève de Socrate.
4. Tout philosophe étudie la nature.
5. Tout élève de Socrate est un génie.
6. Certains élèves de Socrate sont des génies.

Aucune de ces phrases ne contient de connecteur logique comme en logique propositionnelle, en particulier de *si... alors...* De ce point de vue, rien ne les distingue et pourtant il y a des liens entre ces différentes phrases qui peuvent se prêter à des inférences. Par exemple, si on affirme les énoncés 1, 2 et 4, tout le monde sera d'accord pour en conclure que Socrate et Platon étudiaient la nature. Pour représenter ces liens, nous devons distinguer, à l'intérieur des énoncés, les propriétés et les objets auxquels ils s'appliquent. Les deux premières phrases disent que Platon et Socrate ont la même propriété, celle d'être des philosophes. On peut représenter cela par :

1. PHILOSOPHE (Platon)
2. PHILOSOPHE (Socrate)

Quant au troisième énoncé, il met en cause une propriété *ÉLÈVE_DE* non plus d'un seul objet, mais de deux objets, Socrate et Platon. On peut représenter cet énoncé ainsi :

3. *ÉLÈVE_DE* (Platon, Socrate)

Une propriété d'un objet ou un lien reliant deux objets est appelé prédicat. On peut utiliser des prédicats établissant des liens entre plus de deux objets. Par exemple, l'énoncé « Jean et Marie sont les parents de Yves » peut se représenter à l'aide du prédicat *PARENTS DE* (Yves, Jean, Marie).

Examinons maintenant la quatrième phrase : « Tout philosophe étudie la nature ». Elle contient un élément de plus. Il s'agit d'un énoncé universel qui s'applique à tous les hommes. À moins de vouloir remplacer cette phrase par un ensemble d'énoncés pour chaque philosophe :

- SI PHILOSOPHE (Platon) ALORS *ÉTUDIE_LA_NATURE* (Platon),
SI PHILOSOPHE (Socrate) ALORS *ÉTUDIE_LA_NATURE* (Socrate), etc.,

nous devons avoir recours à l'emploi « d'individus variables » ou de variables logiques. On peut alors représenter la phrase comme suit :

4. SI PHILOSOPHE (individu) ALORS *ÉTUDIE_LA_NATURE* (individu).

Une variable (comme « individu ») prend ses valeurs dans un certain domaine qui doit être spécifié. Ici, il est sous-entendu que ce domaine est l'ensemble des êtres humains. Un individu donné comme Socrate ou Platon est appelé constante. Les trois premières phrases ne contenaient que des constantes. Voilà pourquoi il s'agit de connaissances factuelles. Par contre, la phrase 4 est générale et se ramène à un si... alors... On reconnaît là une connaissance conditionnelle.

Par contre, pour distinguer les phrases 5 et 6, on doit déborder le cadre des deux types de connaissances déclaratives précédents. Des mots comme « Tout » ou « Certains » sont déterminants. On les appelle quantificateurs, car ils spécifient la quantité d'individus auxquels l'énoncé s'applique : tout le monde ou au moins un individu. Ainsi, on accordera plus de crédibilité à l'assertion 6

6. ÉLÈVE_DE (Socrate, personne) et GÉNIE (personne) qu'à l'assertion 5;

5. SI ÉLÈVE_DE (Socrate, personne) ALORS GÉNIE (personne).

Les notions de variable individuelle, de quantificateur et de prédicat constituent la base d'un schème de représentation des connaissances appelé logique des prédicats. C'est le modèle utilisé par les ontologies et le langage OWL (Ontology Web Language) dont nous avons parlé dans un texte précédent. OWL comporte plusieurs sous-langages qui permettent de traiter divers sous-ensembles de la logique des prédicats.

En simplifiant un peu, les connaissances prédictives peuvent être décrites comme suit :

- un ou plusieurs quantificateurs suivis d'une assertion contenant les variables quantifiées;
- cette assertion est formée de prédicats accompagnés de leurs arguments (variables ou constantes) et reliés par des connecteurs logiques (et, ou, non, si... alors, etc.).

Les connaissances prédictives généralisent les connaissances factuelles et conditionnelles. Cependant, elles ne recouvrent pas tous les types d'énoncés. Par exemple, dans certains cas, on ne veut pas associer une valeur vraie ou fausse à l'énoncé, mais une probabilité : « Il est probable à 75 % qu'il pleuve demain. » Ce type d'énoncé peut être représenté dans ce qu'on appelle des logiques probabilistes ou des logiques floues.

Nous avons des outils de base pour exprimer des connaissances déclaratives. Ces outils sont complémentaires et cohabitent dans la plupart des représentations puisqu'ils représentent des aspects différents du savoir : des connaissances particulières, des connaissances générales et des liens entre des connaissances. D'autre part, les connaissances déclaratives sont des *connaissances statiques* dans le sens où elles n'ont pas pour fonction principale de décrire un processus. Elles permettent de décrire le *quoi* sans préciser le comment. Pour être utiles, de telles connaissances doivent être manipulées et reliées à d'autres formes du savoir.

2.3 Connaissances procédurales

Les **connaissances procédurales** forment un ensemble structuré de connaissances. Elles décrivent la manière de faire les choses. On peut les exprimer en « style libre » comme on écrit une recette de cuisine ou un guide d'emploi d'un appareil d'utilité courante. Mais on peut également les représenter selon des règles standard de structuration. Dans ce cas, on formulera les connaissances procédurales sous forme *d'algorithme*. On comprendra qu'un algorithme peut s'appliquer à plusieurs situations selon le niveau de généralité de sa formulation et des connaissances qu'il fait intervenir.

La conception d'algorithmes est au cœur de la programmation structurée. Cette discipline informatique prescrit des règles de structuration dans l'écriture de la solution d'un problème. Brièvement, voici les règles de base de la programmation structurée.

Un algorithme est une combinaison d'un nombre fini de Structures de contrôle logique élémentaires suivantes :

- la séquence,
- la répétitive (*tant que* [condition] *faire* [séquence]),
- l'alternative (si [condition] *alors* [séquence] *sinon* [séquence]).

Une séquence est une suite d'une ou de plusieurs opérations élémentaires telles que des calculs, une lecture ou une écriture, une affectation ou l'appel d'une fonction.

- Une structure logique élémentaire constitue une *procédure* qui ne comporte qu'un seul point d'entrée (le début) et qu'un seul point de sortie (la fin).
- On peut « décomposer » une procédure en d'autres structures élémentaires.

Voici comment on peut respecter ces règles dans la production d'une facture à un client.

Algorithme Produire facture

Obtenir les coordonnées du client
Imprimer les coordonnées du client
[Traiter la liste des produits]
[Produire le sommaire]
Arrêter

Procédure Traiter la liste des produits

Initialiser les totaux de la facture
TANT QU'il reste des produits à facturer au client FAIRE
Obtenir l'identification et la quantité du produit
Trouver le prix unitaire
 Prix = prix unitaire* quantité
 Imprimer les données de vente du produit
 Total = Total + prix)

Procédure Produire le sommaire

SI client privilégié
 ALORS % réduction = 10 %
 SINON
 % réduction = 0 %
Total = Total* (1 - % réduction)
Taxe = Total* % taxe
Montant à payer = Total + taxe
Imprimer le sommaire

L'intérêt d'une telle représentation est son haut degré de standardisation qui s'apparente à la rigueur et à la précision des langages informatiques. Des procédures si précises sont directement informatisables. Il faut cependant remarquer que certaines opérations « élémentaires » nécessiteraient d'être détaillées selon la capacité de représentation et de

traitement de la machine chargée de les exécuter. Ainsi, le niveau de généralité d'un algorithme est déterminé par la puissance de traitement de l'information de son exécutant.

Les trois éléments essentiels des connaissances procédurales sont des mécanismes de contrôle, des connaissances factuelles et des opérations à exécuter sur ces données. Ce sont des représentations opératoires des connaissances : elles précisent *comment* effectuer des opérations. On les appelle parfois *savoir-faire*. En effet, elles récupèrent des données brutes en vue de produire des informations plus signifiantes.

Bien que les connaissances déclaratives et les connaissances procédurales soient complémentaires, elles se distinguent par l'envergure de leur formulation et surtout par leur degré de structuration.

2.4 Connaissances conceptuelles

Nous venons de voir deux types de connaissances : le savoir et le savoir-faire. Ces deux représentations sont complémentaires mais insuffisantes pour rendre compte de tous les types de savoir. On a vu qu'elles expriment divers niveaux de généralité et qu'elles sont conformes à des règles de structuration. Mais leur dénominateur commun est de ne s'appliquer qu'à une seule règle ou à un nombre limité de cas concrets. Les connaissances déclaratives et procédurales n'ont qu'une portée restreinte, elles ne sont pas universelles.

Il y a pourtant un type de connaissances qui intègre les formes déclaratives et procédurales et qui permet de construire des ensembles structurés susceptibles de représenter un savoir assez général. On l'a déjà vu en parlant des concepts. Les auteurs Fortin et Rousseau définissent un *concept* comme « une représentation mentale abstraite qui nous permet de catégoriser des *objets* [...] ».

Un concept serait donc le résultat d'une construction mentale, une entité qui existe dans le cerveau et que l'on peut utiliser pour traiter un objet concret. La définition de concept fait également intervenir la notion de *catégorie*. Cela veut dire que la construction d'un concept fait intervenir une étape d'organisation, de classement et de structuration des connaissances qui le composent. L'exemple de la section 1.4 illustre ce point de vue.

L'organisation des connaissances est nécessaire. En effet, la quantité de savoir qu'acquiert une personne au cours d'une vie l'exige. Il faut procéder à des réorganisations du réseau des connaissances déjà acquises pour y incorporer de nouvelles connaissances. Si un cerveau manque de catégories générales (les concepts) qui lui permettent de classer de nouvelles données, il atteint rapidement un point de saturation, un haut niveau de confusion.

La figure 4 représente un exemple de connaissances conceptuelles :

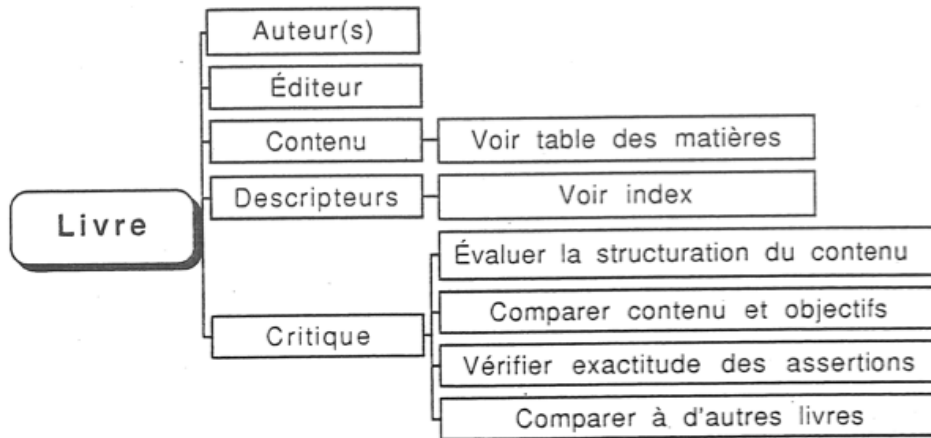


Figure 4 – Le concept de *livre*

On voit qu'un concept fait intervenir plusieurs types de connaissances :

- des connaissances déclaratives,
- des connaissances procédurales,
- des relations avec d'autres concepts.

3. Modèles pour la représentation des connaissances

Nous voilà dans le vif du sujet. Jusqu'à maintenant, nous avons tenté de cerner intuitivement la notion de connaissance. Désormais, nous allons employer un vocabulaire plus formel et utiliser des modèles opératoires, c'est-à-dire des modèles sur lesquels un programme informatique peut effectuer des transformations. Pour ce faire, nous allons utiliser un langage, la modélisation par objets typés pour lequel des outils de modélisation graphique ont été construits et sont largement utilisés. Nous présentons tout d'abord les bases de l'outil G-MOT qui permet de représenter graphiquement les divers types de connaissances de la section précédente.

3.1 Le langage graphique de modélisation par objets typés (MOT)

Comme tout langage, MOT se caractérise par sa grammaire et sa sémantique. Comme MOT est un langage graphique, sa grammaire sert à définir les règles d'utilisation des symboles graphiques qui composent l'alphabet ou le lexique du langage. Sa sémantique est la définition du sens, de l'interprétation donnée aux symboles.







La grammaire et la sémantique du langage MOT

L'alphabet ou lexique du langage MOT se compose de six types de connaissances² et de six types de relations entre elles. Les connaissances sont représentées par des figures géométriques qui en identifient le type. Elles peuvent être des connaissances abstraites ou des faits.

² Le concept de connaissance utilisé ici est plus large que dans la plupart des ouvrages qui traitent du sujet. Il inclut non seulement les faits et les connaissances abstraites, concepts, procédures, principes, mais aussi les habiletés mentales vues comme des métaconnaissances. En somme, est connaissance tout ce qui peut être appris par l'esprit humain, y compris les habiletés cognitives, motrices ou socio-affectives.

- Les *connaissances abstraites* sont des connaissances représentant des classes d'objets, permettant de classer les faits et de les organiser en des ensembles cohérents plus ou moins complexes. On distingue trois types de connaissances abstraites : les concepts, les procédures et les principes (voir tableau 1)
- Les *connaissances factuelles* ou *faits* sont les données, les observations, les exemples, les prototypes, ou les énoncés qui représentent des objets particuliers.

Tableau 1 – Symboles représentant les types de connaissances dans le langage MOT.

Type de connaissance	Connaissance abstraite		Connaissance factuelle	
Déclarative Le quoi des choses	Concept		Exemple	
Procédurale Le comment de choses	Procédure		Trace	
Stratégique Le pourquoi, le quand, le qui	Principe		Énoncé	

Du point de vue de la sémantique, les *concepts* ou connaissances conceptuelles décrivent la nature des objets d'un domaine (le quoi); ils représentent une classe d'objets définis par leurs propriétés communes, chaque objet de la classe se distinguant des autres par les « valeurs » que prennent ses propriétés.

Les *procédures* ou connaissances procédurales décrivent des ensembles d'opérations permettant d'agir sur les objets (le comment); elles s'intéressent aux combinaisons d'actions s'appliquant dans plusieurs cas, chaque cas se distinguant des autres par les objets auxquels les actions peuvent s'appliquer et les résultats des transformations qu'on leur fait subir.

Les *principes* ou connaissances stratégiques sont des énoncés permettant de décrire les propriétés des objets, d'établir des liens de cause à effet entre des objets (le pourquoi), ou de déterminer dans quelles conditions appliquer une procédure (le quand); les principes prennent le plus souvent la forme « Si telle condition Alors, telle condition ou telle action ».

Tel qu'indiqué sur le tableau 1, on distingue également trois types de faits correspondants aux trois types de connaissances abstraites : les exemples, les traces et les énoncés. Les *exemples* sont obtenus en spécifiant les valeurs de chacun des attributs d'un concept, obtenant un ensemble de faits décrivant un objet concret bien précis. Les *traces* sont obtenues en spécifiant les variables de chacune des actions qui composent une procédure, obtenant un ensemble d'actions particulières, une trace d'exécution. Les *énoncés* sont obtenus en spécifiant les variables d'un principe, obtenant ainsi un lien de cause à effet entre les propriétés d'objets particuliers ou entre des propriétés d'un objet particulier et une action précise à effectuer.

Les *relations* entre connaissances sont représentées par des liens orientés (flèches) munies d'une lettre désignant le type de la relation. Les relations entre connaissances appartiennent à six types dont voici la sémantique :

- Le *lien d'instanciation* (I) relie une connaissance abstraite à un fait obtenu en donnant des valeurs à tous les attributs (variables) qui définissent la connaissance abstraite. Chaque connaissance abstraite, concept, procédure ou principe « s'instancie » ainsi à un ensemble de faits appelé respectivement exemple, trace ou énoncé.

Exemple : « Les voitures Renault » a pour instance « La-voiture-de-Jean ».

- Le *lien de composition* (C) relie une connaissance à l'une de ses composantes ou de ses parties constitutives³. On peut spécifier ainsi les attributs d'un objet comme des composantes d'une connaissance en reliant celle-ci à chacun de ses attributs par un lien de composition « se compose de ».

Exemples : L'« Automobile » se compose d'une « Carrosserie » et d'une « Transmission ». L'automobile a un prix et une couleur.

- Le *lien de spécialisation* (S) met en relation deux connaissances abstraites de même type dont l'une est « une sorte de », un cas particulier de l'autre. Autrement dit, la seconde est plus générale ou plus abstraite que la première.

Exemple : Une « Décapotable » est une sorte d'« Automobile ».

- Le *lien de précédence* (P) relie deux procédures ou deux principes dont le premier doit être terminé, réalisé ou évalué avant que le second ne le soit.

Exemple : « Faire le plan » précède « Rédiger le texte ».

- Le *lien intrant-produit* (I/P) relie un concept et une procédure. Du concept vers la procédure, on dira que le concept est un intrant de la procédure. De la procédure vers le concept, on dira que la procédure a pour produit le concept, celui-ci représentant généralement la classe d'objets résultant de la procédure.

Exemple : « Le plan » est intrant de « Rédiger le texte ». « Rédiger le texte » a pour produit « le texte ».

- Le *lien de régulation* (R) s'utilise d'un principe vers une autre connaissance abstraite qui peut être un concept, une procédure ou un autre principe. Dans le premier cas, le principe définit le concept par des contraintes à satisfaire (parfois appelées contraintes d'intégrité), ou encore établit une loi ou une relation entre deux ou plusieurs concepts. D'un principe vers une procédure ou un autre principe, le lien de régulation signifie que le principe contrôle de l'extérieur (régit) l'exécution d'une procédure ou la sélection d'autres principes. Si le lien unit une connaissance à un principe qui la régit, il faut lire le lien « est régi par ».

Exemples : « Les règles de disposition sur la page » régissent « le plan ». « Les règles de contrôle du trafic » régissent « Faire décoller un avion ». « Les règles de gestion de projet » régissent « Les principes de design à appliquer »

L'utilisation de ces relations est soumise à des règles de grammaire. En voici quelques-unes.

- Si l'origine et la destination d'un lien est un concept, alors la relation peut uniquement être de type C (composition) ou S (spécialisation) »; la seconde signifie que

³ Soulignons que le lien de composition a un sens plus large que dans d'autres systèmes de représentation où il prend le sens de « a pour partie » (on lit toujours les liens de l'origine vers la destination). Ici, le lien C peut aussi être utilisé à cette fin comme dans « l'automobile se compose (a comme parties) la carrosserie et la transmission », mais aussi plus largement pour tout attribut d'un objet, par exemple « l'automobile (le concept) se compose d'une carrosserie, d'un prix, d'une couleur ».

- « Le seul lien possible d'un concept vers une procédure ou d'un procédure vers un concept est un lien I/P (intransit/produit) ».
- Le lien S et le lien C ne se placent qu'entre connaissances de même type.

Le langage MOT permet de construire des modèles graphiques en assemblant les connaissances et les relations de façon à respecter les règles de grammaire. À titre d'exemple, le réseau d'association ou carte conceptuelle de la section 1.4 est représenté par le modèle MOT de la figure 5 construit à l'aide du logiciel G-MOT.

Par rapport au modèle de la section 1.4, ce modèle G-MOT a le mérite de distinguer du premier coup d'œil les concepts (rectangles foncés), les attributs des concepts (hexagones), appelés aussi prédicats ou propriétés tels que « respire-par » et les exemples des concepts (en pointillé) tels que « Une Truite » ou « Une Baleine ». On note également l'utilisation de trois liens du langage MOT : le lien de spécialisation S entre concepts, le lien d'instanciation I entre les concepts et les membres de la classe ou exemples, et les liens R entre les concepts et les prédicats.

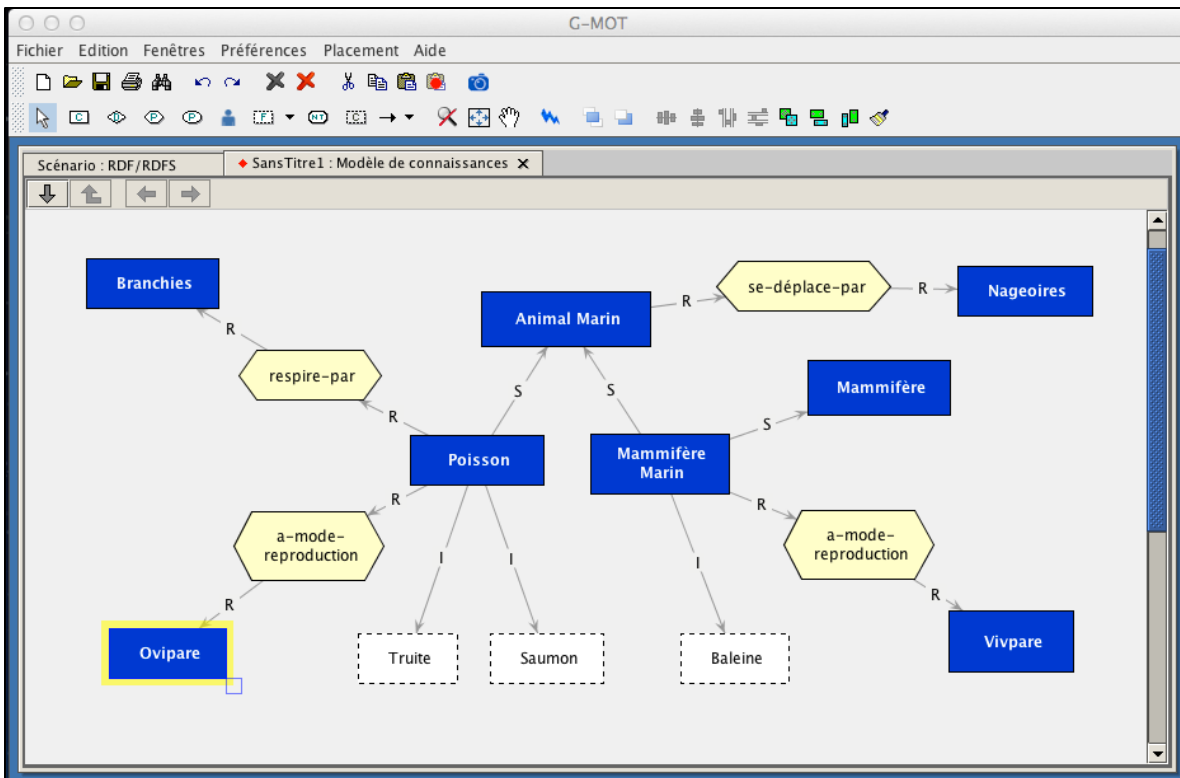


Figure 5 – Un réseau d'association de concepts.

Exemples de modèles MOT

Nous présentons maintenant trois exemples simples utilisant d'autres représentations connues et nous en donnons l'équivalent dans le langage MOT pour en illustrer la généralité.

Les *arbres sémantiques* servent à catégoriser des concepts et en fournir des exemples. Le format graphique met en évidence les liens hiérarchiques entre les concepts. En voici un exemple.

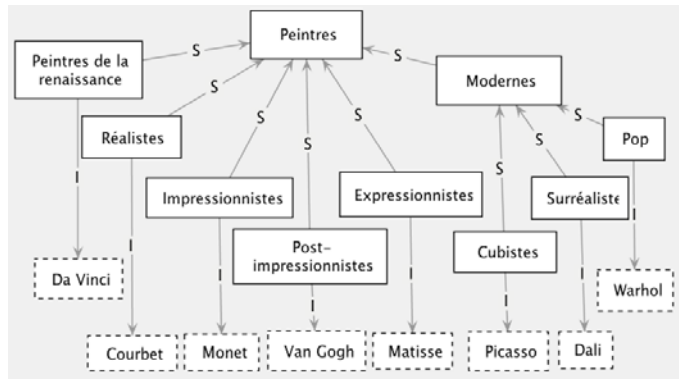
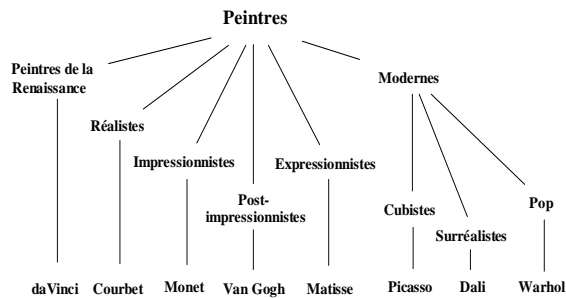


Figure 6 – Un arbre sémantique en histoire de l’art et un modèle MOT correspondant

Le graphe de gauche de la figure 6 fait ressortir une hiérarchie, de la catégorie la plus générale, « Peintres » jusqu’aux individus représentatifs de certains groupes ou écoles de peintures. Remarquons que ce type de graphe, en tant que langage, laisse plusieurs interprétations possibles car il ne précise pas la nature du lien qui est de plus non orienté. Dans des domaines moins bien connus ou totalement inconnus du lecteur, la distinction entre les deux types de liens passera probablement inaperçu. Les liens supérieurs indiquent une relation entre une classe et ses sous-classes (liens S), alors que les liens inférieurs sont des liens d’appartenance d’un individu à une classe (liens I) Le graphe correspondant en langage MOT fait mieux ressortir ces distinctions.

Les *ordinogrammes* sont un autre type de modèle largement utilisé pour représenter des algorithmes et plus généralement des procédures. La figure 7 présente les grandes lignes d’une méthode visant la validation des matériels pédagogiques par les apprenants. Le graphe se compose d’actions et de décisions qui s’enchaînent dans le temps. Il comporte un certain nombre de cycle ou de boucles, par exemple, lorsque la validation indique que les besoins sont satisfaits.

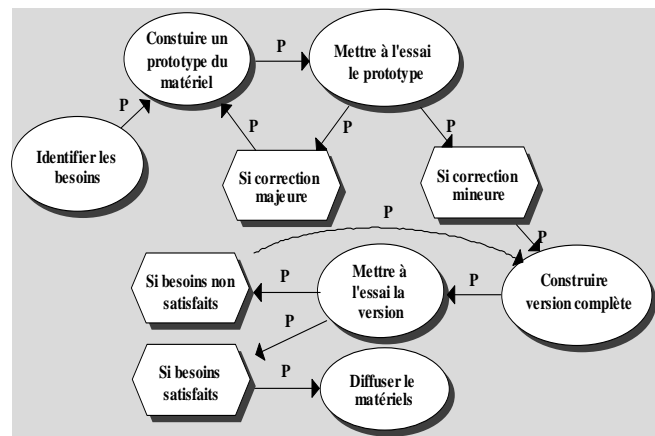
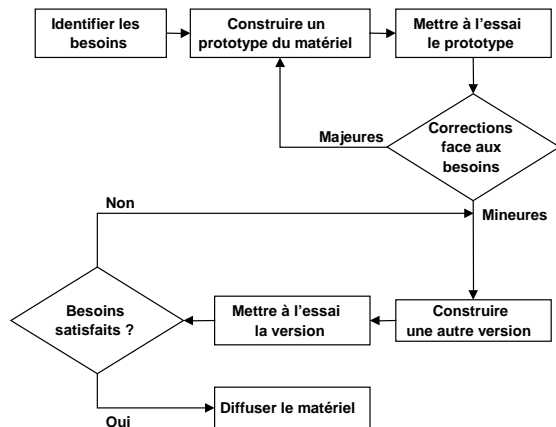


Figure 7 – Un exemple d’ordinogramme en langage MOT

La figure 7 présente un modèle correspondant en langage MOT qui comporte la même expressivité mais avec une signification plus précise en distinguant les tâches (procédures), des décisions et des types liens.

Les *arbres de décision*, utilisés dans le domaine des systèmes experts, servent à représenter une base de connaissances sous formes de règles. Dans le langage MOT, les règles sont des principes. On retrouve de tels arbres de décision dans de nombreux domaines : choix d'un véhicule financier à partir de certaines caractéristiques du client, identification de plantules de mauvaises herbes, détermination du statut d'un immigrant selon les lois de citoyenneté et d'immigration, etc. La figure 8 présente un exemple (fictif) visant à déterminer un taux d'imposition progressif pour un contribuable selon son revenu imposable, selon qu'il est marié ou non ou selon qu'il a des enfants ou non.

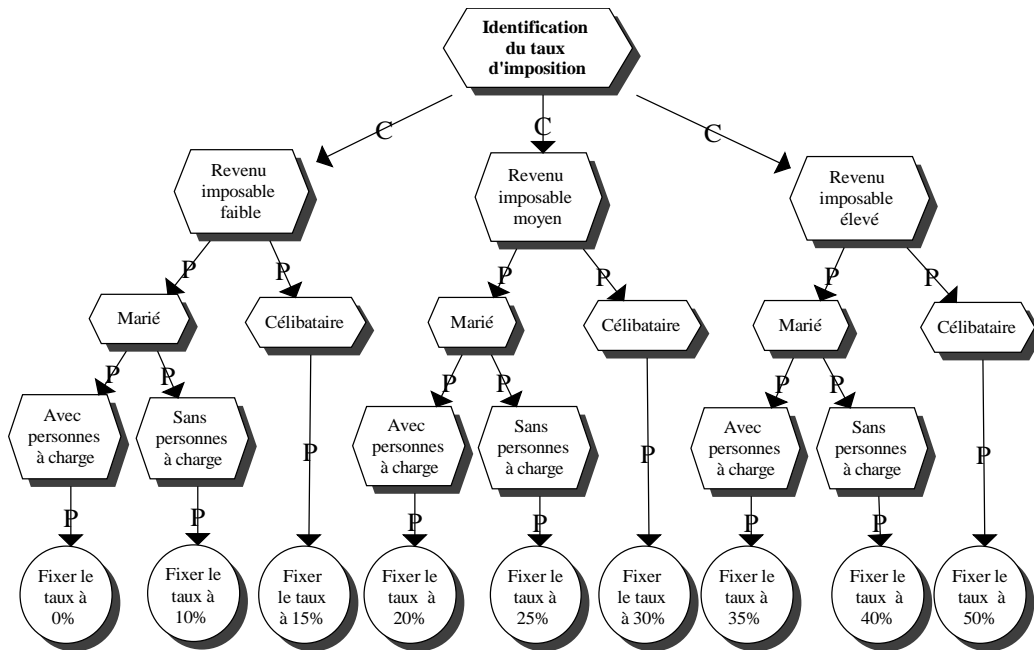


Figure 8 – Un arbre de décision pour le calcul de l'impôt en langage MOT

La base de règles se compose (lien C) de trois groupes de règles selon que le contribuable ait un revenu imposable faible moyen ou élevé. Dans chaque groupe on retrouve d'abord une condition selon que le contribuable est marié ou célibataire; puis une autre condition spécifiant si le contribuable a ou non des personnes à charge. Chaque groupe se compose ainsi de trois principes, chacun correspondant aux trois chemins principaux de parcours du graphe. Par exemple, le premier groupe (« revenu imposable faible ») se compose de trois règles qui se lisent ainsi :

- Si le revenu imposable est faible et le contribuable est marié avec personnes à charge, alors le taux est de 0 %
- Si le revenu imposable est faible et le contribuable est marié sans personnes à charge, alors le taux est de 10 %
- Si le revenu imposable est faible et le contribuable est célibataire, le taux est de 15 %

On le constate, les types de connaissances de base, concepts, procédures et principes, et les trois types de faits, peuvent se combiner aux 6 types de relations en des systèmes de plus en plus complexes de connaissances structurées. Le tableau 2 en donne une synthèse.

Tableau 2 – Catégories de modèles.

Classe	Définition	Exemples
Modèles factuels	La majorité des connaissances sont des faits	L'ensemble des valeurs des attributs décrivant une personne. Les traces d'exécution d'une procédure de traitement d'un formulaire. Un réseau de liens entre des personnes.
Modèles conceptuels	La majorité des connaissances sont des concepts	Une automobile, ses sous-systèmes et leurs composantes. La taxonomie du règne animal. Un schéma décrivant le concept de cube, ses composantes et les méthodes de calcul de sa surface et de son volume
Modèles procéduraux	La majorité des connaissances sont des procédures	Une recette de cuisine. Une méthode de calcul de l'impôt sur le revenu. La procédure de résolution des systèmes d'équation à deux inconnues. Le processus de fonctionnement d'un programme informatique.
Modèles prescriptifs	La majorité des connaissances sont des principes	Un « check list » de conditions permettant de choisir une maison en fonction de ses besoins. L'ensemble des lois de la gravitation. Un arbre de décision pour faire un diagnostic médical.

3.2 Les triplets objet-attribut-valeur

Tous ces modèles, très différents entre eux peuvent être représentés graphiquement à l'aide de l'outil G-MOT. Nous nous concentrons par la suite sur deux classes de modèles particulièrement importantes pour les systèmes de gestion des connaissances :

- les *bases de règles*, classe particulière de modèles prescriptifs qui permettent de représenter la logique procédurale d'un système ;
- les *ontologies*, classe particulière de modèles conceptuels (ou déclaratifs) qui permettent de décrire une théorie d'une domaine du savoir par ses concepts et leurs propriétés.

Mais auparavant, nous mettons en évidence une unité de base de la plupart des modèles de connaissances. Il s'agit des triplets objet-attribut-valeur.

Par objet, on entend une entité que l'on peut abstraire de la réalité. Ce peut être une personne, une plante, un animal, un compte de taxes, une facture, une machine, etc. La signification que l'on peut attribuer à un objet dépend du contexte d'où il est extrait. Par attributs d'un objet, on entend les caractéristiques, les qualités, les paramètres qui servent à le définir. Les types d'attributs d'un objet, leur nombre et leurs valeurs constituent un ensemble qui définit l'objet selon le contexte.

Un objet et sa liste d'attributs ne décrivent que les propriétés générales de plusieurs objets ayant ces propriétés en commun. Ils ne disent rien à propos d'un objet particulier, c'est-à-dire d'une *occurrence* de l'objet. Ce sont les *valeurs* d'attributs qui définissent concrètement un objet spécifique. On pourra représenter complètement toutes les occurrences réelles décrites par un objet abstrait et sa liste d'attributs par toutes les *combinaisons objet-attributs-valeurs* de cet objet. Une combinaison objet-attributs-valeurs décrit une seule occurrence d'objet concret. Voilà d'où vient l'appellation triplet objet-attributs-valeurs.

Selon cette approche, on représente un contexte en procédant en deux temps :

- identifier les objets et, pour chacun, faire la liste de ses attributs;
- faire l'inventaire complet de tous les triplets objet-attributs-valeurs.

Prenons pour exemple d'objet une université. Elle a des programmes, un nombre d'étudiants et bien d'autres attributs qui la caractérise.

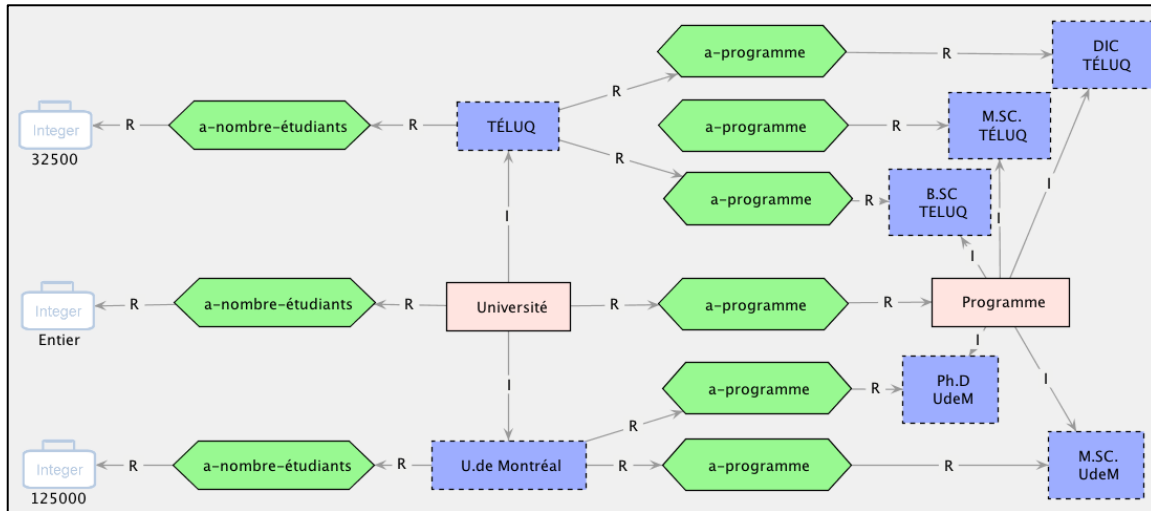


Figure 8 – Un vocabulaire et des triplets objet-attribut-valeur (données fictives)

La figure 8 présente un graphe qui entreprend cette description. Dans la partie centrale du graphe, on retrouve deux triplets (objet, propriété, valeur) génériques mais de nature différente, correspondant à deux attributs ou propriétés : « a-nombre-étudiants » et « a-programme ».

- Le premier triplet, (Université, a-nombre-étudiants, Entier) a comme troisième élément un entier; la propriété a-nombre-étudiants est une propriété de données, un attribut qui prend une valeur entière précise.
- Le second triplet, (Université, a-programme, Programme) relie deux concepts, ceux d'université et de programme; la propriété a-programme est une propriété relationnelle ou d'objet.
- Ensemble, les deux triplets fournissent un vocabulaire pour décrire les universités. Dans la terminologie du Web sémantique on dira qu'il s'agit d'un modèle RDFS (Resource Description Framework Schema).

Les autres triplets du graphe font partie de l'inventaire de tous les triplets particuliers à une université précise. Ils sont obtenus en donnant des valeurs particulières aux concepts « université » et « programme », ainsi que pour le nombre d'étudiants. Ces triplets constituent un graphe RDF selon le standard Ressource Description Framework.)

3.3 Modèles à base de règles (non structurés)

Nous avons déjà une idée intuitive de la notion de règle. En effet, on a vu que pour codifier les connaissances procédurales, nous prenons comme base le savoir déclaratif d'un contexte. Maintenant il nous reste à identifier des *relations* entre les objets et leurs attributs dans un

contexte précis. Ce type de relation peut s'exprimer par des règles de savoir dont le format est : si condition, *alors* conclusion.

Cette formulation met l'accent sur la signification d'une règle et surtout sur son potentiel de déduction. Il nous reste à préciser cette formulation « intuitive ». Or, on a déjà signalé en définissant une assertion qu'un fait rend compte de la valeur d'un attribut d'un objet. Par conséquent, on écrit une règle selon le format suivant :

SI [attribut] relation [valeur]

ALORS [attribut] relation [valeur]

Maintenant, on peut généraliser ce format en remarquant que la partie condition peut être constituée de plusieurs faits. Il peut en être de même pour la partie action. Autrement dit, la partie condition et la partie action sont des listes. Il s'agit d'une structure de données fréquente en informatique. En outre, on appelle parfois *prémisse* la partie condition et *conclusion* la partie action. La conclusion est l'élément essentiel d'une règle, car c'est précisément une *déduction* que l'on peut faire en appliquant une règle. Avec des règles, une déduction se traduit par l'affectation d'une valeur à un attribut. En fournissant des règles à un programme, celui-ci pourra donc déduire de nouveaux faits en exploitant l'information dont il dispose. N'est-ce pas là un type de comportement intelligent?

Nous adopterons les termes *condition* et *conclusion* pour désigner les deux parties d'une règle, car ils rendent bien compte des possibilités opératoires : elle permet de conclure ou de déduire quelque chose si une condition est jugée vraie. De plus, la conclusion peut prendre la forme d'une action ou d'une procédure. En somme, le format général d'une règle est :

SI partie condition (liste d'assertions)

ALORS partie conclusion (liste d'actions ou d'assertions)

On formera les listes en reliant assertions et actions avec les opérateurs logiques *et*, *ou* et *non*. Voici quelques exemples de règles qui pourraient constituer des données d'un programme qui effectuerait des calculs d'impôt.

SI état civil est marié ET
nombre d'enfants = 3

ALORS exemption = 11 000 \$

SI revenu_imposable > 15 000 \$ ET
revenu_imposable > 12 000 \$

ALORS taux_d'imposition = 8 %

Dans ces exemples, on peut déduire deux conclusions relatives à l'exemption et au taux d'imposition d'un contribuable. Mais on pourrait en déduire bien d'autres si l'on avait d'autres règles à appliquer. C'est ce que nous ferons par la suite : on traduira les données d'une situation en règles. Les réponses cherchées seront déduites par un programme qui appliquera des règles, c'est-à-dire qui affectera des valeurs à des attributs encore inconnus.

Une base de règles est un ensemble de plusieurs règles qui constitue un type de base de connaissances pour un système à base de connaissances (SBC). En fait, une base de règles (BR) est à un SBC ce qu'une base de données (BD) est à un système de gestion de bases de données (SGBD). Cependant, une BR est plus générale, car en plus de contenir des connaissances factuelles comme les données brutes dans une BD, elle renferme des

connaissances déclaratives et procédurales codées sous forme de règles. De ce point de vue, une BR est un prolongement d'une BD.

D'autre part, une base de règles forme un réseau de connaissances par l'effet du mécanisme suivant : un fait déduit dans la partie conclusion d'une règle peut être réutilisé dans la partie condition d'une autre règle qui, une fois appliquée, déduira à son tour un autre fait dans sa partie conclusion. L'application successive de plusieurs règles conduit à la déduction de plusieurs faits et constitue un enchaînement de déductions. C'est ce que l'on appelle un *mécanisme d'inférence*.

Revenons au contexte familier de la description de micro ordinateurs. À titre d'exemple, appliquons un mécanisme d'inférence en admettant que « Souris est standard » est un fait vrai. Voyons quelles sont les déductions possibles en appliquant les règles de la figure 9.

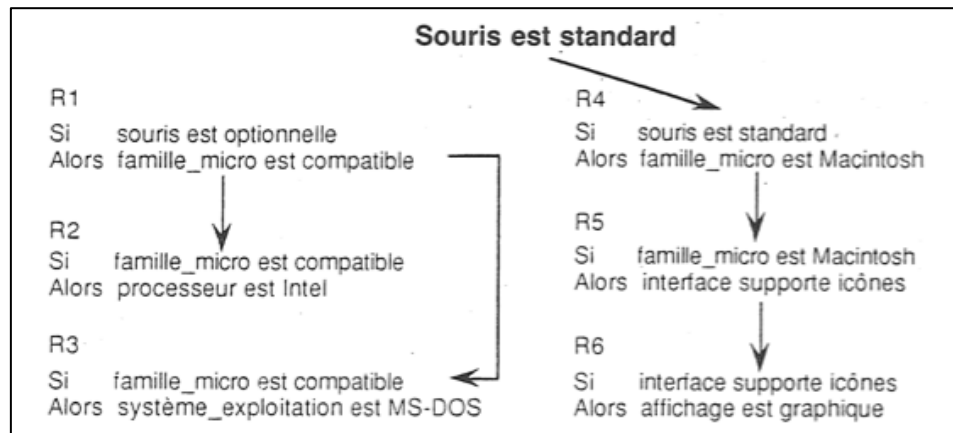


Figure 9 – Base de règles pour les micro-ordinateurs

Voici comment fonctionne un mécanisme d'inférence qui utilise ces règles comme base de connaissances et « Souris est standard » comme *l'hypothèse* de départ du processus de déduction.

- Recherche des règles qui possèdent « Souris est standard » dans leur partie condition. Seule la règle R4 est applicable.
- Application de la règle sélectionnée. Cela a pour conséquence de déduire le fait « famille_micro est Macintosh ». Le procédé se poursuit en prenant comme nouvelle hypothèse le fait déduit par la règle R4.
- Recherche des règles qui possèdent « famille_micro est Macintosh » dans leur partie condition. Seule la règle R5 est applicable.
- Application de la règle sélectionnée. Cela a pour conséquence de déduire le fait « interface supporte icônes ». Le procédé se poursuit en prenant comme nouvelle hypothèse le fait déduit par la règle R5.
- Recherche des règles qui possèdent « interface supporte icônes » dans leur partie condition. Seule la règle R6 est applicable.
- Application de la règle sélectionnée. Cela a pour conséquence de déduire le fait « affichage est graphique ». Le procédé se poursuit en prenant comme nouvelle hypothèse le fait déduit par la règle R6.
- Recherche des règles qui possèdent « affichage est graphique » dans leur partie condition. Il n'y a aucune règle applicable. Le processus de déduction s'arrête ici.

On peut décrire ce processus de manière procédurale. Voici un algorithme simplifié d'un mécanisme d'inférence :

Procédure INFÉRENCE

BASE DE FAITS <-un fait admis comme étant vrai

TANT QU'il reste des règles applicables FAIRE

- Sélectionner les règles dont la partie condition est dans BASE DE FAITS
- Appliquer les règles sélectionnées, c'est-à-dire déduire des faits
- BASE_DE_FAITS<----faits déduits par l'application de règles

Le point de départ d'une séquence de déductions est un fait considéré comme vrai. Une déduction est le résultat de l'application d'une règle. Cela se traduit par *une affectation d'une valeur à un attribut*. Dans notre exemple, la règle R4 affecte la valeur *Macintosh* à l'attribut *famille_micro*. Mais il y a plus car la conclusion d'une règle peut servir de donnée à une autre règle. Ainsi, la règle R5 affecte la valeur *icônes* à l'attribut *interface* et la règle R6 la valeur *graphique* à l'attribut *affichage*. De cette manière, on peut faire exécuter un enchaînement de déductions en appliquant successivement plusieurs règles comme le prescrit notre algorithme simplifié. C'est ainsi que l'on établit des relations entre les règles.

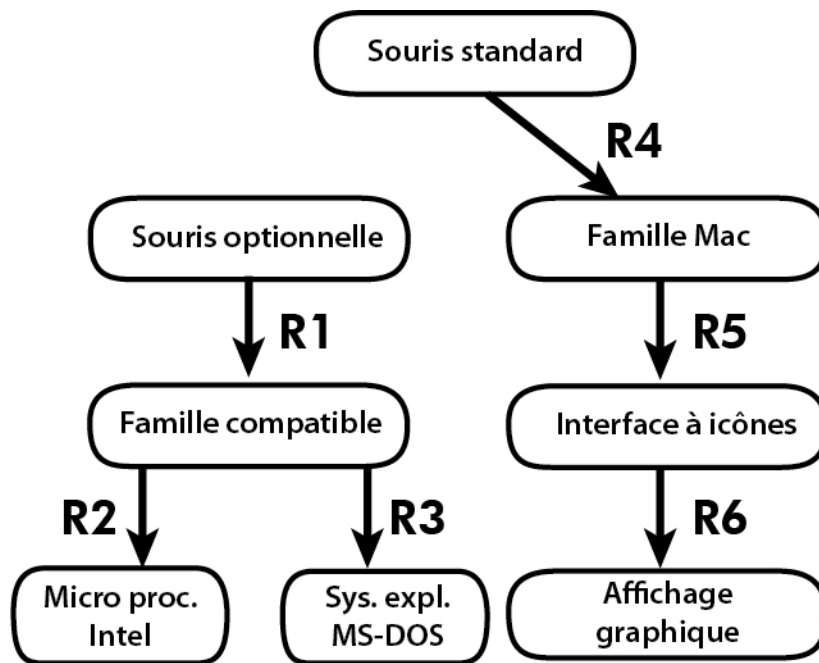


Figure 10 – Un réseau de faits obtenus par inférence sur une base de règles.

Nous venons d'illustrer un mécanisme d'application *automatique* des règles par affectation de valeurs d'attribut ou instanciation. Il s'agit d'une forme d'inférence. Ce phénomène peut se comparer à un parcours dans un réseau de faits que l'on relie par un enchaînement de déductions. Les faits constituent les nœuds du réseau et les règles en sont les arcs. La figure 10 montre le réseau correspondant à l'exemple des micro- ordinateurs.

En terminant, mentionnons que les bases de règles sont les bases de connaissances les plus répandues en IA parce qu'elles peuvent être implantées sur micro-ordinateur et parce que la

codification sous forme de règles est accessible aux non-spécialistes. D'autre part, la taille d'une base de règles, c'est-à-dire le nombre de règles qu'elle contient, dépend de la complexité de l'application. En général, une base de règles contient de 100 à 5 000 règles.

3.4 Modèles à base de schémas (structurés)

Cette représentation est connue depuis le milieu des années 70. Elle fut proposée par Marvin Minsky comme modèle de certains concepts mentaux. En intelligence artificielle, on l'adopta comme une structure de données à transmettre à un ordinateur. En anglais, on désigne un schéma par le terme *frame*. Voici ce qu'en écrivent les auteurs Harmon et King :

« Un schéma est une description d'un objet subdivisé en *compartiments* contenant l'information associée à l'objet. Comme les attributs, les compartiments peuvent emmagasiner des valeurs. Ils peuvent également contenir des valeurs par défaut, des pointeurs vers d'autres schémas, des règles ou des procédures qui peuvent générer des valeurs. »

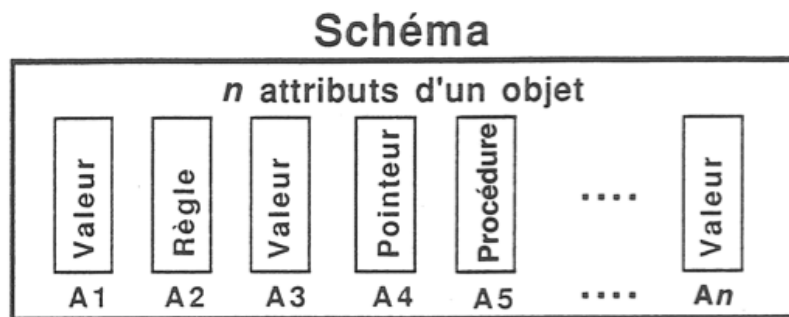


Figure 11 – Un schéma contient différents types d'attributs d'un objet.

Le schéma de la figure 11 illustre cette définition. Les compartiments libellés A1, A2, jusqu'à A *n* de la figure désignent des noms d'attributs de l'objet et leur contenu est indiqué ici par Valeur, Règle, Pointeur et Procédure. Il s'agit d'un modèle très général et souple pour représenter un concept. En effet, un schéma rend compte d'une bonne diversité de formes des connaissances. Il peut contenir des valeurs (connaissances déclaratives) et des procédures (connaissances procédurales). De plus, le nombre d'attributs qui définissent un objet n'est pas fixé à l'avance. Cela permet de construire des modèles souples et puissants. Enfin, le fait qu'un attribut puisse être un pointeur vers un autre schéma permet de relier plusieurs schémas entre eux. Un schéma est donc un modèle qui peut représenter une vaste gamme d'objets différents.

La figure 12 est un exemple de schéma qui pourrait représenter la déclaration des revenus simplifiée d'un particulier.

Déclaration des revenus pour particulier

Nom <input style="width: 90%;" type="text"/>	Exemption personnelle de base <input style="width: 90%; text-align: center;" type="text" value="6 000 \$"/>
Revenu brut <input style="width: 90%;" type="text"/>	Impôt à payer <input style="width: 90%;" type="text"/>
Revenu imposable <input style="width: 90%;" type="text"/>	Impôt payé <input style="width: 90%; text-align: center;" type="text" value="Voir feuillet T4"/>
Déduction pour famille <input style="width: 90%; text-align: center;" type="text" value="Remplir annexe B"/>	Remboursement ou Solde dû <input style="width: 90%;" type="text"/>

Figure 12 – Un exemple de schéma

Dans ce schéma de déclaration des revenus, les cases vides sont des attributs dont les valeurs seront remplies lors de la spécification d'un cas particulier. Par ailleurs, il y a des cases déjà remplies. La case « Exemption personnelle de base » contient une valeur par défaut. La case « Déduction pour famille » contient l'appel à une procédure et la case « Impôt payé » contient un pointeur à un autre schéma, le schéma « feuillet T4 ».

Il nous reste maintenant à montrer comment on peut relier plusieurs schémas pour représenter plusieurs objets concrets. On verra que les schémas peuvent d'une part, traduire de vastes ensembles de connaissances et d'autre part, être soumis à des programmes qui pourront les traiter comme données servant à déduire de nouvelles connaissances.

Les bases de schémas constituent probablement la représentation des connaissances la plus générale tout en contenant les valeurs d'attributs d'un objet. Ces valeurs peuvent être obtenues de diverses manières : par défaut, par l'application d'une règle, par l'exécution d'une procédure ou en ayant recours à un autre schéma.

Les objets représentés par des schémas sont définis par hiérarchie. En effet, on regroupe dans une *classe* tous les objets qui possèdent les mêmes attributs. D'autre part, certains attributs d'un objet sont transmis à tous ceux qui lui sont subordonnés. Il s'agit d'une réalisation simple de la notion *d'héritage*. Une base de schémas représente donc un ensemble hiérarchisé d'objets en vue d'assurer la transmission d'attributs des niveaux supérieurs aux niveaux inférieurs dans la hiérarchie. Il y a là une analogie intéressante avec la représentation d'un concept introduite à la section 1.4. Ainsi, la figure 3 montre que les concepts *poisson* et *mammifère marin* ont hérité des attributs *locomotion* et *milieu* du concept *animal marin*. Cela semble indiquer qu'un schéma est un bon modèle d'un concept mental.

Le phénomène d'héritage est la fonctionnalité la plus intéressante des bases de schémas. C'est là que réside la puissance de ces outils. La transmission de certaines caractéristiques d'un objet à un autre par héritage est gérée à l'aide d'un attribut commun à tous les schémas : l'attribut *parent*, c'est-à-dire le nom du seul objet qui est le « géniteur » logique de l'objet à définir. C'est justement l'objet parent qui transmet ses attributs à tous ses « descendants ». En fait, l'attribut parent de chaque objet tient lieu de *pointeur* pour relier entre eux les objets. Selon cette approche, une collection d'objets est représentée par un arbre hiérarchique, chacun des nœuds de l'arbre étant un objet.

D'autre part, pour répondre aux requêtes de l'utilisateur, les schémas doivent communiquer entre eux. Cela se fait par *échange de messages* entre les schémas. C'est ainsi que des schémas sont reliés entre eux pour former une base cohérente.

Exemple

Voyons maintenant comment on peut concevoir une base de schémas pour représenter un contexte simple et surtout illustrer la hiérarchie des schémas et l'héritage d'attributs. Nous allons représenter en partie la diversité des micro-ordinateurs domestiques par une base de schémas représentant certains d'entre eux.

La transmission d'attributs par héritage à la figure 13 s'effectue ainsi : le schéma de l'objet *Zénith* a hérité des attributs des objets *Compatible* et *Micro*. C'est ainsi que l'attribut *Souris* de *Zénith* hérite de la valeur *Option* de son parent *Compatible*. De même, il possède les valeurs *Intel* et *MS-DOS* sans que les attributs *Processeur* et *Système d'exploitation* ne lui soient associés. L'objet *Zénith* a hérité des valeurs des objets qui lui sont supérieurs en hiérarchie. De la même manière, cette base de schémas montre que le micro-ordinateur Tandy est muni d'un processeur Intel. Par ailleurs, on note que deux attributs sont communs à tous les schémas de cette base : le *nom* et le *parent* d'un objet. Il s'agit d'une règle générale à respecter dans la définition d'un schéma.

Pour illustrer le phénomène d'échange de messages entre les schémas et l'utilisateur, considérons le cas suivant :

- L'utilisateur veut savoir quel est le processeur du micro-ordinateur Tandy. Il s'agit d'une demande d'information. Pour y répondre, le système doit lire la valeur de l'attribut « Processeur ». La requête de l'utilisateur s'adresse d'abord au schéma « Tandy ».
- Puisque le schéma « Tandy » ne dispose pas de l'attribut « Processeur », il envoie un message à son parent « Compatible » pour qu'il se charge de la lecture de l'attribut « Processeur ». Le schéma « Compatible » dispose de l'attribut cherché. Sa valeur « Intel » est lue et retournée à l'utilisateur, en guise de message-réponse à sa requête initiale.

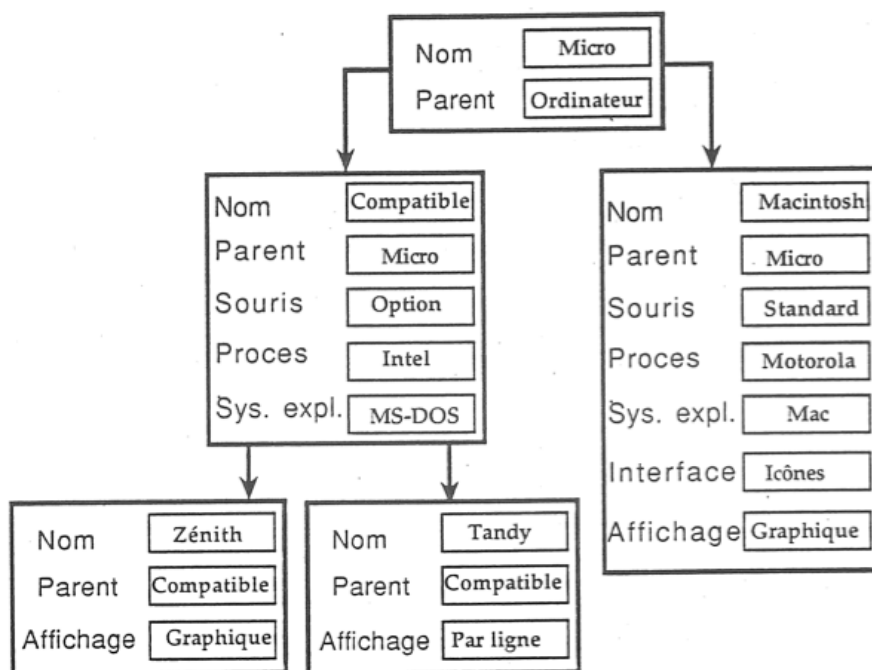


Figure 13 – Une base de schémas de certains ordinateurs (fin des années 80)

Bien que notre exemple soit très simple, il illustre les phénomènes d'héritage d'attributs et d'échange de messages, soit deux fonctionnalités essentielles d'une base de schémas. On peut généraliser pour suggérer la procédure générale de communication par envoi de messages entre schémas dans une base de schémas hiérarchisés. Brièvement, les schémas échangent des messages selon le procédé suivant :

- L'utilisateur fait une demande à la base de schémas. Cette requête (message initial à la base) peut en être une d'information ou de traitement. L'interface usager de la base de schémas identifie l'attribut qui caractérise la requête. Dans le cas d'une demande d'information, une valeur d'attribut est lue et retournée à l'utilisateur. Dans le cas d'une demande de traitement, une valeur est affectée à un attribut suite à l'exécution de procédures de traitement.
- Le schéma qui reçoit la demande est activé. Deux cas sont possibles :
 1. Le schéma qui a reçu la requête (le message) de l'utilisateur peut y donner suite. C'est le cas d'un schéma qui dispose des attributs qui caractérisent la demande de l'utilisateur. Ce sont les procédures attachées aux attributs du schéma qui seront exécutées pour répondre à la demande. Le résultat de cette exécution est retourné à l'utilisateur : il s'agit du message généré par le schéma.
 2. Le schéma qui a reçu la requête (le message) de l'utilisateur ne peut y donner suite. C'est le cas d'un schéma qui ne dispose pas de tous les attributs qui caractérisent la demande de l'utilisateur. Une recherche est entreprise pour retrouver le ou les schémas qui possèdent les attributs de la requête de l'utilisateur. Le schéma récepteur envoie un message à son schéma parent. Le message remonte dans la hiérarchie des schémas jusqu'à ce que la requête puisse être satisfaite. Ce sont les procédures attachées aux attributs des schémas appropriés qui seront appelées à répondre à la demande. Le résultat de cette exécution est retourné à l'utilisateur : il s'agit du message généré par la base de schémas.

Pour créer une base de schémas, on définit d'abord des objets généraux dont les attributs sont transmis par héritage. Ensuite, on définit progressivement les autres objets qui sont des cas particuliers des objets généraux tout en possédant leurs attributs. Il s'agit d'une démarche qui procède du général au particulier par raffinement graduel. On est ainsi amené à définir des classes. Une classe est un groupe d'objets qui possèdent des attributs communs.

On peut imaginer une base de schémas représentant l'ensemble du personnel d'une organisation. Ainsi, on pourrait définir un schéma général *employé* qui contiendrait des attributs communs à l'ensemble du personnel : identification, coordonnées, département, ancienneté et qualifications. Ensuite, on pourrait définir les schémas *temps plein* et *temps partiel* pour tenir des attributs particuliers à chacun de ces statuts. Ces deux derniers schémas auraient *employé* comme parent et, de ce fait, hériteraient de tous ses attributs. On pourrait continuer ainsi selon les besoins de l'application.

Nous venons de voir que les règles et les schémas peuvent traduire certains aspects du savoir déclaratif et procédural et qu'ils rendent compte des relations entre des données. Ainsi, ils peuvent représenter des *réseaux* de connaissances. Leurs dénominateurs communs sont l'emploi de la liste comme structure de données et la notion d'objet défini par ses attributs comme base conceptuelle. Rappelons que dans le modèle humain présenté à la section 1, on associait le mot *réseau* à la mémoire à long terme et le mot *liste* à la mémoire à court terme. S'agirait-il d'une coïncidence?

3.5 Avantages et limites des modèles présentés

La première qualité de toute représentation doit être son potentiel explicatif. D'autre part, elle doit faciliter la synthèse et la structuration des connaissances, être précise et concise. De ce point de vue, on peut affirmer que les représentations exposées dans cette section sont excellentes! De ces avantages découlent cependant certaines de leurs lacunes. En effet, les représentations déduites des notions d'objet et d'attributs sont nécessairement simplificatrices. Elles sont *réductionnistes* dans le sens où la réalité complexe qu'elles représentent se voit confinée dans des listes, des diagrammes simples ou des expressions très rigides. En somme, ces représentations ne rendent pas compte des nuances parfois nécessaires à la compréhension. C'est pourtant le prix à payer si l'on veut simplifier suffisamment la réalité pour la maîtriser et la codifier en termes informatiques.

Les métaphores inspirées du couple objet-attributs décrivent une réalité figée, c'est-à-dire qu'elles conduisent à des *images statiques* des objets. et qu'elles sont incapables d'exprimer l'évolution et les mouvements de ces objets. Ainsi, on peut représenter des objets stables, voire inertes, mais on ne peut rendre compte d'évènements. Par exemple, comment représenter la croissance d'un organisme? L'aspect dynamique de la réalité est plus difficile à cerner.

En outre, le mot *image* ne fait pas ou très peu partie du vocabulaire des représentations à base d'objets et d'attributs. C'est certainement l'une des raisons qui rendent les ordinateurs incapables de rivaliser avec un être humain quant à la signification et à la puissance des représentations qu'ils peuvent supporter. Les possibilités actuelles de la technologie sont telles que l'ordinateur est limité à la gestion de bases de connaissances dont les éléments primaires sont des entités logiques, des nombres ou de l'information textuelle. On ne peut que constater la faible intégration, voire l'absence, des images comme moyen de représentation des connaissances. On est limité au traitement de symboles. À quand les représentations reposant sur des *bases d'images*? C'est pourtant l'une des sources de la puissance de traitement du cerveau humain.

En somme, les principales faiblesses des représentations par objets sont :

- le réductionnisme, l'absence de nuances;
- l'aspect statique des modèles, l'absence de mouvements et de transitions;
- l'absence des négations et des contraintes;
- l'ignorance des images et de leur puissance de représentation.

Les constats précédents incitent à concevoir d'autres représentations qui combleraient les lacunes évoquées. Ainsi, on a proposé des modèles basés sur la notion de *script* pour représenter des *événements*. Un script a une structure analogue à celle d'un schéma. Il est donc constitué d'attributs pour décrire un événement : acteurs, lieux, temps, accessoires, buts, etc. Il est aussi une description d'un épisode dans une suite d'évènements, ce qui lui permet de raisonner sur ce qui pourrait survenir selon la situation. Ces outils remédient à certaines faiblesses des représentations par objets, notamment parce qu'ils rendent compte d'un point de vue dynamique, c'est-à-dire le changement ou le mouvement typique d'un événement. Mais ils ne sont pas des modèles universels. La solution à la représentation des connaissances réside probablement dans une *symbiose de plusieurs représentations*, car la diversité des formes de connaissances l'exige.

Conclusion

Dans ce deuxième texte, nous avons couvert un large éventail de notions et d'outils susceptibles de nous assister dans la tâche de représentation des connaissances. La *diversité* est d'ailleurs un trait caractéristique des connaissances et conséquemment des manières de les représenter. Mais cette diversité ne devrait pas nous empêcher de dégager des *thèmes unificateurs*, des idées maitresses.

Nous avons vu qu'il y a plusieurs types de connaissances. Ce sont des formes complémentaires de « la » connaissance. On tente de les intégrer dans les représentations qui permettent de construire un *modèle* abstrait des objets réels en vue de leur traitement informatique. Tous les modèles explicités dans ce chapitre reposent sur le postulat suivant :

On peut représenter une entité de la réalité par un objet auquel on associe une liste d'attributs.

L'objet sert de base conceptuelle commune à la plupart des représentations des connaissances. Le concept d'objet permet de définir les connaissances, c'est-à-dire montrer de *quoi* elles sont composées. D'autre part, il faut expliquer *comment* les choses fonctionnent et de quelle manière elles sont reliées. Pour y parvenir, on traduit les objets et leurs attributs en *règles* ou en *schémas* qui vont permettre de rendre opérationnelles les connaissances. On a vu que les règles et les schémas ont un effet d'organisation et de structuration des connaissances. Ils ont pour principale fonction d'établir des *liens* entre des connaissances isolées et de les exploiter pour en déduire de nouvelles par l'application de mécanismes d'inférence. Nous avons souvent mentionné qu'une base de connaissances prend la forme d'un *réseau* de données. À cet effet, nous avons succombé à des tentations anthropomorphiques. Nous avons « repiqué » des éléments d'un *modèle humain* d'organisation des données en mémoire pour les appliquer à une base de connaissances.

Dans une perspective d'informatisation, l'application des concepts dont nous discutons ici contribue à élaborer des *bases de connaissances* (BC) auxquelles on appliquera des méthodes. De plus, nous avons vu qu'une BC peut prendre la forme d'une *base de règles* ou d'une *base de schémas*. Pour aider à cerner les caractéristiques – on pourrait dire les attributs – d'une BC, nous proposons une analogie avec une base de données « classique ». Prenons d'abord la définition et le contenu d'un fichier dans un contexte SGBD. Une entité est définie par un fichier, lequel est décrit par une liste de champs. Le contenu d'un fichier est l'ensemble des valeurs des attributs appelées occurrences de champs.

À retenir

1. L'activité intelligente des humains peut être décrite par une succession d'évènements qui transforment des informations captées par les sens en actions exercées sur l'environnement. Cette activité dépend des connaissances constituées à partir d'un ensemble de faits simples et de structures générales basées sur ces faits.
2. L'expertise d'une personne est la capacité de mettre en pratique ses connaissances pour résoudre des problèmes et réaliser des tâches complexes.
3. Il y a différents types de connaissances : connaissances déclaratives, procédurales, conceptuelles et méta-connaissances.
4. Il faut construire des représentations des connaissances si l'on veut éventuellement leur appliquer des traitements informatisés.
5. Un objet défini par une liste d'attributs peut représenter une entité du monde réel. Le triplet objet-attributs-valeurs est la base conceptuelle de la plupart des représentations des connaissances.
6. Une règle est une relation *si... alors...* entre deux triplets objet-attributs-valeurs. Une base de règles peut représenter un ensemble complexe de connaissances; on peut en extraire de nouveaux faits par un mécanisme d'inférence.
7. Un schéma permet de représenter un concept ou un objet. On construit une base de schémas en regroupant des objets pour former des classes hiérarchisées. Une classe transmet par héritage ses attributs aux schémas qui lui sont subordonnés. Des schémas peuvent communiquer par échange de messages.
8. Chaque outil de développement informatique intègre un langage qui permet de codifier et d'exploiter une représentation particulière des connaissances. La représentation détermine le traitement.