

Qu'est-ce que la connaissance?

Où se trouve la connaissance?

Comment formuler la connaissance?

Introduction au débat : le point de vue d'un membre de la communauté de l'ingénierie de la connaissance = un ingénieur de la connaissance ?

[texte préparé pour une table ronde sur ce sujet aux journées ASTI 2005

<http://www.isima.fr/asti2005>]

Alain Mille

Qu'est-ce que la connaissance ?

Le point de vue d'un ingénieur, informaticien de surcroît, sur la connaissance est étroitement lié à l'histoire de ce qu'on appelle toujours les Techniques de l'Intelligence Artificielle.

Construction de la notion de connaissance en Intelligence Artificielle

- Acte de naissance : 1956, Darmouth College (New Hampshire, USA)
 - o Avec John McCarthy (tenant de la logique) et Marvin Minsky (tenant d'une approche par schémas)

La genèse de l'IA baigne dans l'idée de construction de « machines à penser » et la comparaison du cerveau avec les premiers ordinateurs est faite avec beaucoup d'enthousiasme.

- Il y a conjonction de faits qui confortent cette idée comme par exemple :
 - o Mc Culloch et Pitts : réseaux neuronaux artificiels (approche physiologique)
 - o Wiener : cybernétique
 - o Shannon : théorie de l'information
 - o Von Neumann : architecture d'un calculateur
 - o Turing : théorisation des fonctions calculables par machine (et de manière moins connue l'intuition des systèmes dynamiques représentés par des systèmes d'équations différentielles).

Newell, Simon et Shaw proposent un premier programme de démonstration de théorèmes en logique (1956!) et généralisent en proposant le General Problem Solver qui progresse dans la résolution en évaluant la différence entre la situation du solveur et le but à atteindre. C'est toujours ce qui constitue aujourd'hui les « Bases de l'intelligence artificielle ».

Il faut démontrer par des applications spectaculaires les possibilités ainsi ouvertes.

- Au travers des « jeux » au premier rang desquels on trouve le jeu d'échecs naturellement (premières idées en 1950 par Shannon!) -> première victoire sur un maître en 1997 Deep Blue bat Kasparov. Le terrain de recherche du jeu reste particulièrement actif et productif comme le récent prix Nobel d'économie à l'Américain Thomas Schelling et à l'Israélo-Américain Robert Aumann le démontre bien.
- Test « d'intelligence » (Evans 1963) : trouver la suite d'une série de figures.
- Résolution de problèmes par propagation de contraintes (Waltz 1975)
- Dialogue en « langage naturel » (Eliza, Joseph Weizenbaum 1965) (Système SHRDLU, Terry Winograd 1971) Kitano recevait un prix spécial pour ses travaux de traduction automatique en 1993 et un prix de même nature fut réservé à Wolfgang Wahlster en 2001 lors de IJCAI à Seattle.

Et par des « systèmes experts comme

- DENDRAL (en chimie)
- MYCIN (en médecine)
- Hearsy II (en compréhension de la parole)
- Prospector (en géologie)

On voit bien au travers de ce très rapide historique, que si la cognition est bien au cœur du débat, c'est tout d'abord la capacité à résoudre des problèmes « comme des humains » qui est recherchée et l'on parle plus d'heuristique que de connaissance.

C'est en 1982 que presque simultanément deux chercheurs « fondent » le concept de connaissance en Intelligence Artificielle.

Doug Lenat publie *Knowledge Based Systems in Artificial Intelligence* (1982, McGraw-Hill) après avoir imaginé un système capable d'acquérir continûment... toutes les connaissances que l'on peut lui fournir ! Cette entreprise s'est transformée progressivement en serveur de connaissances CYC = une encyclopédie « intelligente » (à visiter!)

MAIS, surtout Alan Newell a une idée géniale en inventant le « niveau connaissance » à distinguer du niveau symbolique, pour pouvoir travailler sur « les connaissances » en tant qu'objets manipulables. Cet appel fonde du même coup une communauté des « ingénieurs de la connaissance » et lance un programme de recherche toujours très actif.

Alors cette fois, on parle de connaissances (au pluriel) et de systèmes à base de connaissances mais il s'agit plus précisément d'une modélisation « au niveau connaissance » caractérisée par :

- Un processus de conceptualisation du domaine et des interactions sur ce domaine
- L'existence d'un langage partagé par l'expert et le « cognitif », langage capable « d'exprimer » la sémantique des « connaissances » mais nécessitant le passage à un langage exploitable également par « l'artefact » informatique, qui aura sa

sémantique propre et qui doit pouvoir garantir que les calculs seront menés en accord avec l'interprétation intentionnelle des experts..

On peut donc alors acquérir les connaissances, ce qui laisse supposer que l'on saurait répondre à la seconde question de cette table ronde.

Où se trouvent les connaissances ?

Les connaissances se trouveraient chez les experts, dans les documents et dans les données qui commencent à s'accumuler dans les systèmes informatiques. Pour les rendre « objectives », il convient d'observer les experts au travail, de considérer l'utilisation des documents et pour les données, on remobilise et rénove les techniques d'analyse de données pour les faire « parler » (les données !), c'est-à-dire en tirer des « règles » candidates pour représenter des connaissances. Acquisition de connaissances et apprentissage automatique se développent en se croisant parfois.

Comme on le constate facilement, il n'est pas possible de simplement « copier » les connaissances de ces différentes sources vers un système informatique. Il faut créer les modèles de connaissances à transcrire dans un environnement informatique.

Un des plus remarquables résultats du programme de recherche sur le niveau connaissances est l'approche KADS (1989-1993) et de nombreuses variantes d'ateliers autour des mêmes principes. Cette modélisation au niveau connaissance permet tout à la fois de documenter la connaissance et de permettre la conception de Systèmes à Bases de Connaissances plus facile à développer et à maintenir.

La connaissance est-elle cette fois localisée dans l'ordinateur ? Il est permis d'en douter et d'ailleurs personne ne l'affirmerait d'une manière aussi abrupte.

Ce programme de recherche n'arrive toutefois pas à résoudre la difficulté essentielle de construire des Systèmes à Base de Connaissances qui soient durablement utiles. Le moral baisse à nouveau, mais...d'une part la communauté des informaticiens s'est ouverte à des communautés SHS et d'autre part on comprend de mieux en mieux la difficulté qu'il y a à cerner les connaissances pour les modéliser !

Une autre « invention » dans les années 90 va relancer, pour un temps, les choses : il s'agit du concept de « Gestion des connaissances ». La partie « documentation de la connaissance » accessible par la modélisation au niveau « connaissance » permet en effet de capitaliser la connaissance ainsi modélisée et d'accéder plus facilement aux « conteneurs de connaissances » (documents, compétences) qui sont les sources de la modélisation. On renonce en grande partie à « l'engagement » inférentiel de la modélisation mais on espère ainsi pouvoir faire fructifier le capital connaissance de l'entreprise.

⇒ exploitation largement documentaire avec l'idée de pouvoir faire des requêtes « orientées connaissance »

Difficultés à nouveau, car la métaphore du Capital Connaissance a très bien marché d'un point de vue marketing et les attentes sont bien entendues déçues en terme de « retour sur investissement ». On déprime à nouveau un petit peu...mais...la communauté s'est ouverte largement sur les sciences du document et c'est salubre car la notion de connaissance n'y est pas du tout la même que dans l'acception informaticienne de l'IA.

Il n'y a pas si longtemps (mai 2001), Tim Berners Lee a lancé l'idée du « Web Sémantique » qui reprend le principe de l'indexation par le contenu conceptuel des ressources. Il n'y a qu'un pas à franchir pour tenter de s'appuyer sur les vertus de la logique...de description en s'attaquant à toute ressource accessible par le WEB. En d'autres termes, si on décrit logiquement les contenus atteignables sur Internet alors on (les moteurs de recherche) pourra exploiter ces descriptions pour faire des calculs et enrichir considérablement les requêtes faites sur la mine impressionnante atteignable. Plus pragmatiquement, il devient également possible de composer des services élémentaires pour peu que ces services soient décrits logiquement et le WEB devient source de services diversifiés composés automatiquement selon les requêtes utilisateurs. Ces « connaissances » sur le contenu sont supposées être partagées par les utilisateurs de ces contenus. Des groupes d'utilisateurs peuvent effectivement travailler pour décrire avec une sémantique commune les ressources d'un corpus partagé.

Nous sommes actuellement dans cette phase d'activité intense sur ces idées...

Ces différents travaux font avancer considérablement l'informatique moderne, mais est-ce que nous pouvons considérer que nous avons avancé sur la connaissance de la connaissance ?

Oui, probablement, même si à chaque fois qu'on croyait pouvoir la concrétiser en quelque modèle, on s'apercevait que l'on n'avait à nouveau qu'une inscription de la connaissance.

Je ne sais toujours pas où est localisée la connaissance, mais je vois bien que ça n'empêche pas de faire une ingénierie autour de la connaissance.

Alors ? Personnellement, je propose à mes étudiants quelques définitions simples pour s'y retrouver.

- L'inférence est l'élément de base associé à la connaissance.
- Connaître quelque chose c'est être capable de produire une inférence à partir de cette chose.
- La connaissance est donc révélée par cette capacité.
- Toutefois, l'inférence peut s'imaginer avec ou sans représentation de la connaissance (intelligence incorporée).
- Une inscription qui permet l'inférence est une inscription de connaissance.
- Une inscription formelle de connaissances dans un langage possédant une « sémantique » permet de faire des calculs dans un environnement informatique.
- C'est ainsi que l'on parlera le plus souvent de la « représentation de la connaissance ».

La connaissance résiste donc à la « mise en boîte » puisque tout ce que l'on pourra manipuler explicitement est une représentation symbolique dont on espère que les interprétations issues

de calcul ne seront pas trop éloignées des interprétations qui pourraient en être faites humainement parlant.

Nouvelle idée pour s'en sortir donc, l'ingénierie des inscriptions de la connaissance proposée par Bruno Bachimont lors d'une conférence récente (2004) sur l'ingénierie des connaissances !

Cette fois, la prétention est raisonnable : il ne s'agit plus de fabriquer un homonculus (cet être sans âme que les alchimistes espéraient créer), mais plus prosaïquement de profiter des propriétés d'un environnement informatique pour considérer des « inscriptions de connaissances » qui peuvent prendre la forme classique documentaire, mais aussi la forme des modèles précédemment imaginées au « niveau connaissance » dont la nature inférentielle serait alors exploitée explicitement par des utilisateurs « avertis » (au sens propre du terme, c'est-à-dire s'attendant aux inférences qui vont se mettre en place, mais aussi des formes relevant d'une herméneutique (art d'interpréter...) , c'est-à-dire privilégiant les échanges d'explications, les annotations, la glose à l'inscription dans un modèle définitoire.

Cette dernière approche, « l'art d'interpréter », peut se lire comme l'art de proposer de nouvelles inscriptions d'une connaissance révélée à l'interprétant par l'inscription initiale. L'idée est donc que la multiplication des inscriptions par différents interprétants donne nettement plus de chance à un nouvel interprétant de trouver un chemin « d'interprétation » d'une inscription donnée (tout le monde suit ?). Le revers de la médaille, c'est la multiplication des inscriptions de connaissance mais l'environnement informatique peut procurer de multiples façons de les gérer efficacement, une voie explorée aujourd'hui en associant les communautés de linguistes.

Comment « Formuler la connaissance » est donc peut-être la seule question à laquelle j'aurai répondu implicitement en introduisant la notion d'inscriptions de connaissances comme concept générique pour représenter aussi bien les discours transcrits sur la connaissance (bien représenté ici par Sacha Bourgeois-Gironde) que les modèles les plus formels (bien représenté ici par Pierre Lescanne).

Pour notre part, nous proposons de considérer en quoi l'utilisation d'un environnement informatique pourrait accompagner le sujet « connaissant » dans ses tâches lorsqu'elles sont médiées par cet environnement. En d'autres termes, comment proposer des environnements informatiques qui fonctionnent en « intelligence » avec l'utilisateur plutôt que de proposer des systèmes « intelligents ».

Un cas particulier d'inscription de connaissances exploitable « en situation » : les traces d'utilisation d'environnements informatiques.

Pour notre part, nous nous intéressons aux inscriptions de connaissances issues des interactions qu'un utilisateur entretient via un environnement informatique.

Un environnement informatique rassemble en effet des possibilités nombreuses et variées d'inter-actions¹ pour accompagner telle ou telle activité humaine dans des tâches plus ou moins formatées. Nous ne faisons pas d'hypothèse *a priori* sur la notion d'environnement

¹Inter-action : dénote le fait que les actions de l'utilisateur et les réactions de l'environnement sont dépendantes les unes des autres dans le cadre d'une activité particulière. L'utilisateur agit avec l'environnement comme intermédiaire avec lui-même d'abord, rejoignant ici les principes défendus par Vygotski par exemple [ref]. Nous ne considérons pas l'environnement comme un homonculus.

informatique qui peut être considéré à des niveaux de granularité très variables, depuis une fenêtre de configuration jusqu'au système d'exploitation : le contour ou le périmètre ressenti d'un environnement dépend fortement du contexte d'utilisation, ce contexte n'ayant de sens que pour l'utilisateur-observateur à un instant donné. La conception d'environnements informatiques se fonde sur l'analyse de l'activité de ses futurs utilisateurs et des tâches impliquées autant qu'elles puissent être anticipées, et s'attache à offrir différentes possibilités d'inter-action conformément à l'analyse faite. Le périmètre d'un environnement se révèle particulièrement difficile à stabiliser à l'avance : beaucoup d'environnements sont « ouverts » aux autres et il est parfaitement possible par exemple de décider de réaliser un document à partir d'un tableur mais d'exploiter intensivement les possibilités d'un autre environnement comme un traitement de texte dans ce tableur (et vice-versa naturellement). L'affirmation qu'il suffit de bien analyser les besoins de l'utilisateur pour concevoir un environnement qui lui convienne est donc difficile à tenir dans ces conditions. La question de l'accompagnement de l'activité de l'utilisateur par des « facilitateurs² » d'appropriation des possibilités offertes se pose donc dans la conception même de ce qu'est un environnement informatique.

Les environnements mis à disposition pour assister les activités sont donc conçus selon des « logiques³ » de conception qui sont le plus souvent le résultat de démarches « centrées utilisateur », c'est-à-dire en intégrant autant que possible l'observation d'utilisateurs dans le processus de conception [Rabardel, Theureau]. Ces logiques de conception s'expriment au travers de l'environnement développé par les possibilités d'actions offertes sur des « objets informatiques » ainsi manipulables par l'utilisateur, ces possibilités variant selon les différents « états » prévus de l'environnement. Il reste que quel que soit le soin apporté au processus de conception, l'utilisateur tentera d'appliquer ses propres « logiques d'utilisation ». Les logiques d'utilisation se constatent par l'observation et l'exploitation de « retours d'expérience » est la réponse habituelle pour tenter de réduire le hiatus alors constaté par de nouvelles conceptions intégrant ces logiques au mieux par de nouvelles possibilités d'action avec l'environnement. Cette boucle vertueuse peut se révéler source de désorientation pour l'utilisateur devant faire face à un « nouvel » environnement intégrant certes des logiques d'utilisation observées, mais pas forcément chez lui, alors qu'il a lui-même modifié ses logiques d'utilisation en s'adaptant à ce qu'il peut faire selon les logiques de conception de la version précédente. A contrario, il arrive aussi que l'on attende la prochaine version d'un environnement qui offrira enfin telle ou telle possibilité que l'on imagine possible par analogie⁴ avec les possibilités offertes par d'autres environnements.

L'environnement informatique constitue donc par lui-même un terrain de confrontation de logiques de conception s'exprimant au travers des possibilités d'action offertes et de logiques d'utilisation s'exprimant par les actions effectivement faites dans le cadre d'une tâche au sein d'une activité. Cette confrontation est aujourd'hui gérée par l'utilisateur qui découvre progressivement comment composer avec ce qui est possible pour avancer dans sa propre tâche. L'idée principale est donc de fonder un principe d'accompagnement de l'appropriation par une « facilitation » de l'interprétation de ce qui se passe dans le « cours d'actions » pour

2Facilitateur : nous préférons le terme « facilitateur » au terme plus courant « assistant » qui laisserait entendre qu'il y aurait quelque part une « expertise » disponible pour « orienter » l'utilisateur selon des scénarios pré-établis, donc pré-conçus.

3« Logique » (de conception et d'utilisation) : nous gardons ce terme logique pour référer à une certaine rationalité qui pourrait être revendiquée par les concepteurs ou les utilisateurs dans leurs choix. Ce terme est souvent mis au pluriel pour indiquer qu'il y a peu de chance qu'une conception ou une utilisation réponde à une seule « logique ».

4Nous ne considérons pas spécifiquement le cas des experts de la conception d'environnements informatiques qui pourraient imaginer de nouvelles possibilités à partir de leurs connaissances de spécialistes. Ils auraient alors une démarche de résolution de problème plutôt qu'une démarche analogique.

reprendre l'expression de Theureau. Cette démarche relève du constructivisme au sens que lui donnerait Varela plutôt que du cognitivisme plus classique en intelligence artificielle.

D'un point de vue pratique, il s'agit donc de fournir à l'utilisateur une « réflexion » de ses inter-actions « révélant » d'une certaine manière la confrontation en cours liée au hiatus entre logiques de conception et logiques d'utilisation. Les actions possibles et les états atteignables sont liés aux possibilités issues des logiques de conception, le processus de composition avec ces possibilités relève des logiques d'utilisations. Nous proposons de fournir à l'utilisateur une trace des inter-actions telles qu'elles sont observées par un « agent » de traçage. Pour que cette trace soit « lisible » et manipulable symboliquement par l'utilisateur il est nécessaire d'inter-agir également sur la trace en tant qu'objet informatique particulier qu'il doit donc s'approprier également.

L'intégration de cette exploitation de traces transforme alors *de facto* un environnement informatique S en un nouvel environnement informatique S'. L'utilisation du système S' (intégrant la gestion de traces explicites) peut donner lieu à une modélisation de nouvelles utilisations et leur traçage/présentation, pouvant donner lieu à la mise en place d'un système S'' (intégrant l'utilisation des traces de S dans S'). Il s'agit donc bien d'une dynamique d'utilisations nouvelles liées à la disponibilité de traces gérées par l'utilisateur, et la connaissance se construit inter-activement en étant symboliquement représentable par l'utilisateur lui-même à des fins de facilitations de ses propres tâches.