

L'ingénierie des connaissances ?

Une introduction
A Mille
2004-2005

IC A Mille 2004

1

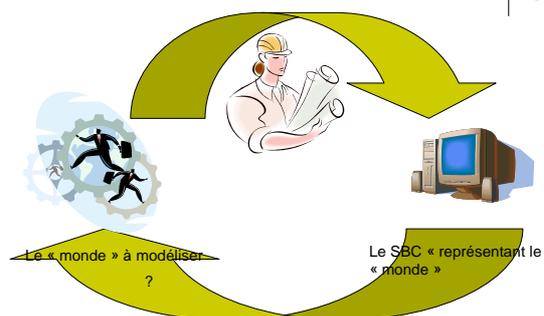
Plan

- Méthodes et outils de l'objectif historique : faciliter la réalisation de SBC
- Focus sur l'élicitation ontologique, avec l'objectif de « gestion des connaissances »

IC A Mille 2004

2

Objectif historique : faciliter la construction des Systèmes à Base de Connaissances



IC A Mille 2004

3

Première étape : les langages de représentation des connaissances

- Idée : Se rapprocher du niveau d'abstraction supposé des experts ⇔ modèle cognitif humain sous-jacent et donc :
- Concevoir des langages facilitant le « codage » de la représentation des connaissances conforme au modèle cognitif
 - LISP (approche résolution de problème -> expression simple de processus récurifs)
 - Prolog = langage de prédicats + moteur d'inférence
 - Smalltalk (représentation objets « réactifs » aux messages, réflexivité, auto-représentation)
 - Yafool, KL, ... schémas ⇔ langages de « frames »

IC A Mille 2004

4

La question de la validation d'un SBC

- Retour vers l'expert
- Tests sur un jeu de problèmes
 - Confronter Experts / SBC
 - Savoir résoudre le plus grand nombre de problèmes / savoir résoudre les problèmes les plus difficiles
 - Évaluer le « raisonnement » suivi

IC A Mille 2004

5

Evaluation de MYCIN

- 10 cas
- 8 experts + Mycin + solution du cas telle qu'elle a été appliquée (10 prescripteurs)
 - Chaque expert évalue toutes les prescriptions ⇔ Note/80
 - Le collège évalue toutes les prescriptions ⇔ Note/8
 - Le collège signale les solutions « inadmissibles »
- Mycin : meilleurs résultats des 10 prescripteurs
- Et pourtant MYCIN n'a jamais pu être implémenté de manière opérationnelle : POURQUOI ?

IC A Mille 2004

6

Une règle MYCIN

SI
la coloration de l'organisme est GRAM+
et si la morphologie de l'organisme est Cocci
et si le mode de développement de l'organisme est en colonies
ALORS
il existe une évidence (0.7) que l'identité de l'organisme soit Staphylococcus.

- Pour écrire une telle règle, le concepteur devra savoir comment MYCIN fonctionne. Ici, il faut savoir que cette règle, parce qu'elle conclut sur l'identité de Staphylococcus et parce que le moteur fonctionne en chaînage arrière, ne sera envisagée que lorsque le système se préoccupera de déterminer l'identité d'un organisme. Il faut de plus savoir qu'à chaque instant, le moteur s'intéresse à un « contexte » particulier, c'est à dire un organisme à retrouver. Un ordre des prémisses dans une autre règle fera en sorte que MYCIN s'intéressera d'abord à l'organisme courant avant de rechercher des organismes présents antérieurement. À chaque type d'organisme est associé un ensemble de caractéristiques (par exemple, identité de l'organisme pour l'organisme courant). À chaque caractéristique est associé un attribut « mis-à-jour-par » qui contient la liste des règles qui permettent d'établir la caractéristique en question (l'identité de l'organisme, par exemple). La règle R1 ci-dessus fait partie de cette liste de règles...

IC A Mille 2004

7

Dépasser les problèmes de MYCIN

- Stratégie de résolution implicite
- Capacités auto-explicatives pauvres
- Grande distance entre la pratique des experts et l'expression sous forme de règles liées à l'approche logique
- Dégradation forte aux limites du domaine couvert
- Connaissances de surface essentiellement
- Pas de généricité des systèmes construits
- Pas de modélisation « lisible » du domaine couvert

IC A Mille 2004

8

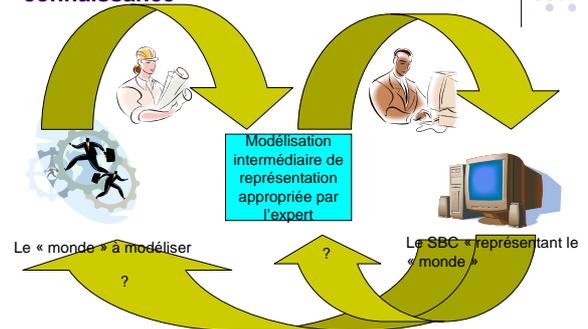
=> Mieux modéliser le raisonnement ?

- Neo Mycin : métarègles / stratégie
- Connaissances stratégiques : comment effectuer une tâche ?
- Connaissances du domaine : qu'est-ce qui est « vrai » dans le domaine et dans le contexte de la tâche considérée
- ⇔ Notion de Modèle Conceptuel = Connaissances stratégiques associées aux connaissances du domaine

IC A Mille 2004

9

Le virage des années 1980 : l'appel d'Alan Newell pour travailler « au niveau connaissance »



IC A Mille 2004

10

Qu'est-ce que la modélisation « au niveau connaissance » ?

- Conceptualisation du domaine et des interactions
- Niveau adapté pour la « spécification » d'un SBC donc :
 - 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 » (calculs en accord avec l'interprétation intentionnelle des experts).

IC A Mille 2004

11

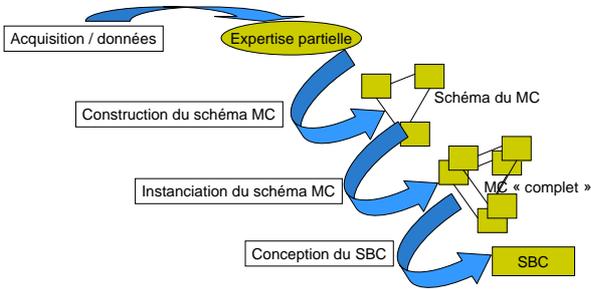
Modélisation au niveau connaissances

- Acquisition de connaissance et compréhension des méthodes de résolution à utiliser
 - Recueil, Analyse ⇔ modèle spécifique
 - Modèle d'expertise = étape dans le développement de SBC
- Modélisation indépendante des structures de représentation et des algorithmes
 - Langage intermédiaire non orienté implémentation, semi-formel
- Acquisition des connaissances guidée par la réutilisation de modèles et/ou de composants génériques

IC A Mille 2004

12

Acquisition de connaissances conduite par les modèles



IC A Mile 2004

13

Acquérir les connaissances ?

- Techniques de recueil des « données » pour construire une description de l'expertise
 - Méthodes informelles : interviews, résolutions à « haute voix », observations, enregistrements, recueil de documents, examen de bases de données, etc...
 - Méthodes formelles : grilles, questionnaires etc...
 - Méthodes statistiques : fouille de données, analyse de données, etc...

IC A Mile 2004

14

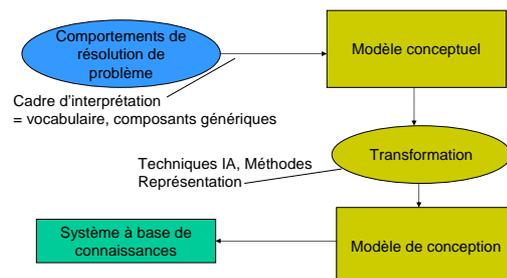
Modélisation conceptuelle

- Doit permettre :
 - D'exprimer comment va être effectuée une tâche. Utilise les connaissances du domaine.
 - D'exprimer les connaissances du domaine ↔ concepts manipulés + relations / tâche dans le domaine considéré

IC A Mile 2004

15

Exemple de Kads



IC A Mile 2004

16

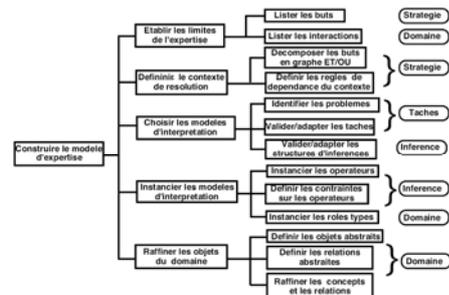
Kads : les quatre couches du modèle d'expertise

- Le niveau Stratégie / Plans, métarègles
 - Contrôle
- Le Niveau Tâche / Buts, description de tâches
 - Exploite
- Le Niveau Inférence / Sources de connaissances, Métaclasses
 - Décrit
- Le Niveau Domaine / Concepts et relations

IC A Mile 2004

17

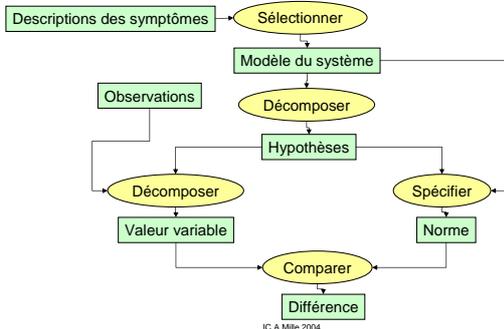
KADS : modèle d'expertise



IC A Mile 2004

18

Kads : Structure d'inférence du diagnostic



IC A Mille 2004

19

La structure de tâche de diagnostic systématique

TASK SYSTEMATIC DIAGNOSIS

Goal : Trouver le plus petit composant démontrant un comportement incohérent.

Control Terms :
 differential = ensemble des hypothèses actives en cours
 inconsistent_subsystem = subpart

Task-structure :

Systematic_diagnosis
 (symptoms-> inconsistent_sub_system) =

Select

(symptoms->system_model)
 generate_hypothesis
 (system_model -> differential)

REPEAT

test_hypothesis
 (differential-> inconsistent_subsystem)
 generate_hypothesis
 (inconsistent_subsystem-> differential)

UNTIL differential est vide

20

Les sous-tâches de génération et de test d'hypothèse

TASK GENERATE_HYPOTHESIS

Goal : générer un nouvel ensemble d'hypothèses par décomposition du système

Task structure :

Generate(system_model -> differential) =
 decompose (system_model -> differential)

TASK TEST_HYPOTHESIS

Goal : tester si une hypothèse de differential est contredite par une observation

Task structure :

test (differential -> hypothesis) =
 DO for EACH hypothesis in differential
 specify (hypothesis -> norme)
 select (hypothesis -> observable)
 OBTAIN (observable -> variable value)
 compare (norm + variable value -> difference)
 UNTIL difference = true

IC A Mille 2004

21

KADS : bibliothèque de modèles d'interprétation



22

Kads : bibliothèque de composants réutilisables

- Modèles et guides modélisation
 - Livre CommonKads, chapitre 6
 - INRIA -> webCOKACE
- Méthodes de résolution de problèmes
 - Inventaire de méthodes de résolution, d'opérateurs et de rôles dans une méthode dans le livre CommonKads, chapitre 13
 - Modèles de méthode
 - Voir <http://www.commonkads.uva.nl/>

IC A Mille 2004

23

Gestion de connaissances : gérer des abstractions partageables

- Lexiques
- Thésaurus
- Ontologies
- Modèles du domaine
- Modèles de cas
- Décomposition en tâches récurrentes
- Méthodes de résolution de problème
- Tâches applicatives abstraites

IC A Mille 2004

24

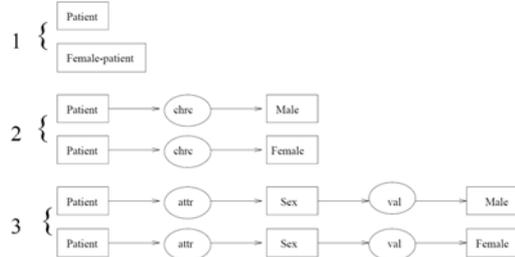
Focus sur la démarche de construction d'ontologies

- Nombreuses « ontologies » pour un système du monde
- Impossibilité de délivrer une ontologie universelle fusionnant les différents points de vue
- Une ontologie porte la trace des tâches pour lesquelles elles ont été construites et du raisonnement sous-jacent pour cette tâche

IC A Mille 2004

25

Exemple



IC A Mille 2004

26

Différentes classes d'ontologie

- Ontologies de « représentation »
 - Primitives de représentation (par exemple ONTOLINGUA)
- Ontologies génériques (« haut » réutilisable)
- Ontologies d'un domaine
- Ontologie d'une méthode de résolution de problème (ou de tâche)
- Ontologie d'application = spécialisation d'une ontologie de domaine + ontologie de méthode

IC A Mille 2004

27

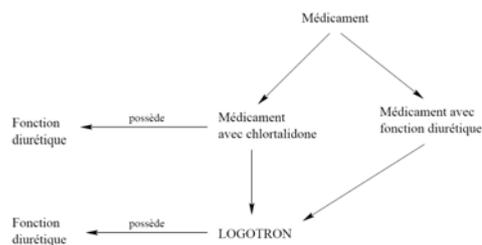
Ontologie pour représenter les connaissances => ajouter la sémantique

- Il s'agit de définir sans ambiguïté les fonctions et les relations qui correspondent à la représentation proposée
 - -> quels calculs/inférences peuvent être faits à partir de l'énoncé d'une fonction/relation ontologique
 - Ontologie ⇔ graphe conceptuel par exemple.
 - ENGAGEMENT ONTOLOGIQUE ?

IC A Mille 2004

28

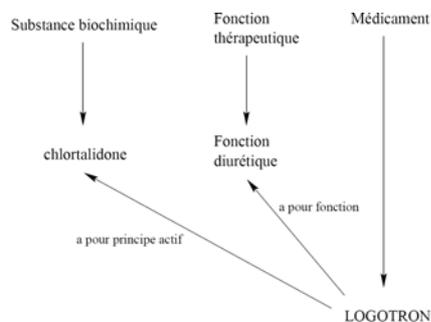
Exemple



IC A Mille 2004

29

Exemple



IC A Mille 2004

30

Méthodologie pour construire une ontologie (Bachimont)



- Hypothèse : domaines se formalisant peu et dont l'accès se fait par la langue naturelle (documents)
- Existence de corpus et d'outils terminologiques pour modéliser le modèle

IC A Mille 2004

31

Méthode



- Analyser le corpus
 - Extraire des candidats-termes, des relations, des verbalisations
 - -> base de connaissances terminologique
- Normalisation sémantique en fonction de la tâche
 - -> arbre de concepts (sorte-de)
 - -> engagement ontologique -> treillis formel
 - -> opérationnalisation dans un langage de représentation (graphes conceptuels, owl, ...)

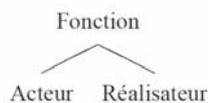
IC A Mille 2004

32

Utiliser des principes différentiels pour faciliter l'engagement ontologique



- Principe 1 de communauté avec le « père »
 - Fonction d'un être humain dans un film
 - Jouer un rôle du film
 - Le réalisateur qui fait le film, l'acteur qui y joue
 - apparaitre dans la film versus travailler sur le film (fonction sur le plateau)
- Principe 2 de différence avec le « père »
- Principe 3 de différence avec les « frères »
- Principe 4 de communauté avec les frères



IC A Mille 2004

33

Pour en savoir plus...



- Une liste étonnante de liens vers des documents, des systèmes, des groupes de recherche (anglo-saxon)... <http://www.cs.utexas.edu/users/mfkb/related.html>
- LE site français sur le sujet de l'ingénierie des connaissances
- <http://www.irit.fr/GRACQ/index-bib.html>
- Une série de cours d'initiation sur le sujet (et sur quelques autres par un collègue de Nice) <http://www.irit.fr/GRACQ/COURS/CoursFabienGandon.htm>

IC A Mille 2004

34