



**HAL**  
open science

## Ontologies informatiques

Fabien Gandon

► **To cite this version:**

| Fabien Gandon. Ontologies informatiques. Interstices, 2006. inria-00080851

**HAL Id: inria-00080851**

**<https://inria.hal.science/inria-00080851>**

Submitted on 22 Mar 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Ontologies informatiques

CULTURE & SOCIÉTÉ

**Dans un système d'information, un petit peu de sémantique peut vous emmener très loin...  
Nous vous invitons à la découverte de la notion d'ontologie et de ses applications.**

## 1. De l'Ontologie à l'ontologie

Le terme « ontologie », construit à partir des racines grecques *ontos* (ce que existe, l'existant) et *logos* (le discours, l'étude), est un mot que l'informatique a emprunté à la philosophie au début des années 1990. En philosophie, l'Ontologie est une branche fondamentale de la Métaphysique, qui s'intéresse à la notion d'existence, aux catégories fondamentales de l'existant et étudie les propriétés les plus générales de l'être. Si vous ouvrez un dictionnaire tel que le Petit Larousse Illustré, la définition n'éclairera probablement pas beaucoup votre lanterne quant à l'intérêt d'importer cette notion en informatique : « (1) Etude de l'être en tant qu'être, de l'être en soi (2) Etude de l'existence en général, dans l'existentialisme ».

Pourtant, à y regarder de plus près, nous pourrions, à l'extrême inverse, penser que beaucoup d'ingénieurs en informatique sont des « messieurs Jourdain » de l'ontologie. Par exemple, lorsque pour implanter une application, les ingénieurs en informatique conçoivent un schéma de classes, ils s'interrogent sur les objets que cette application va manipuler, les classes qui les regroupent, les caractéristiques communes à tous les objets de chaque classe, les relations qui peuvent exister entre ces objets, etc. En d'autres termes, ces ingénieurs s'interrogent sur ce qui définit ces classes d'objets, ce qui permet d'identifier qu'un objet appartient à une classe, ce que cette appartenance signifie en termes de contenu ou de manipulations possibles. Bref, ils s'interrogent sur la définition existentielle des classes d'objets mobilisés dans les scénarios de l'application qu'ils développent.

Vu sous cet angle, l'ingénieur qui conçoit ses représentations logicielles n'est-il pas plus proche de

l'ontologue qui interroge nos conceptualisations du monde qu'il ne le semblait initialement ? Si la programmation orientée objets présente cette ressemblance avec la notion d'ontologie informatique, c'est qu'elles ont un ancêtre commun : les systèmes de l'intelligence artificielle symbolique. Les débuts de cette branche de l'intelligence artificielle se confondent avec les débuts de l'informatique, car dès ses prémices, l'informatique a perpétué le rêve des concepteurs d'automates de simuler voire dépasser l'intelligence humaine avec des systèmes artificiels.

## L'histoire d'une notion à la recherche d'un nom

La branche de l'intelligence artificielle à laquelle nous nous intéressons ici est qualifiée de symbolique parce qu'elle repose sur des représentations formelles des connaissances, sous la forme de symboles que le système peut stocker et manipuler (par exemple, langages et opérations logiques, structures et opérations de graphes). Contrairement à d'autres approches, ces représentations sont à la fois compréhensibles par les humains et manipulables par les systèmes, en appliquant des règles de manipulation définies sur les symboles de ces représentations et dont l'interprétation simule, par exemple, un raisonnement.

Ainsi, dès les années 1970, la notion d'ontologie existait, sans être nommée et de façon transversale, dans les différents systèmes de représentation de connaissances : c'est la TBox des logiques de description, où l'on décrit les types de termes qui existent dans notre représentation et leurs caractéristiques ; c'est le support des graphes conceptuels, où l'on décrit des hiérarchies de multi-héritage entre des types de concepts ou des types de relations ; ce sont enfin les schémas des « Frames » et les classes des langages de représentation par objets.

Il aura pourtant fallu attendre les années 1990 pour que le mot « ontologie » soit adopté par toute la communauté, et sa définition fait encore couler de l'encre (électronique).

## Aux grands mots les grands remèdes

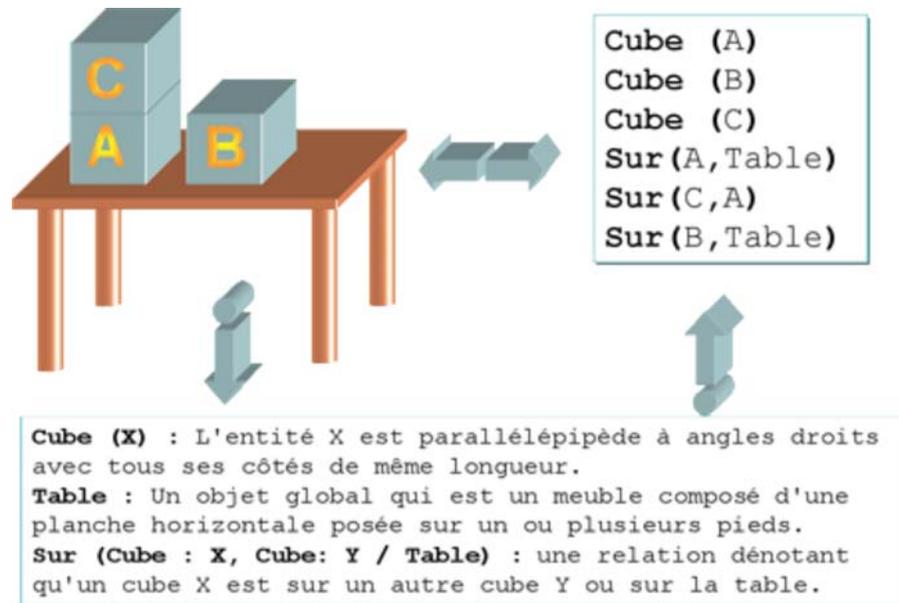
Comble de l'histoire donc, cette notion d'ontologie, qui s'attache tant à la définition précise des concepts que nous manipulons, a longtemps cherché une définition et un nom. Ce retard est probablement largement dû à la nature même de la notion d'ontologie, qui est abstraite.

Pour tenter de définir la notion d'ontologie informatique, il est utile de rappeler que l'Ontologie désigne l'étude des propriétés générales de ce qui existe. En important cette notion en informatique, nous sommes passés de la science (l'Ontologie) à un objet (une ontologie).

Une ontologie informatique est une représentation de propriétés générales de ce qui existe dans un formalisme supportant un traitement rationnel. C'est le résultat d'une formulation exhaustive et rigoureuse

de la conceptualisation d'un domaine. Cette conceptualisation est souvent qualifiée de partielle car, en l'état de l'art, il est illusoire de croire pouvoir capturer dans un formalisme toute la complexité d'un domaine. Notons aussi que le degré de formalisation d'une ontologie varie avec l'usage qui en est envisagé.

C'est pour cet aspect de description de l'existant et de ses catégories que les ontologies informatiques ont emprunté leur nom à l'Ontologie philosophique. De ce rapprochement vient aussi la possibilité d'adapter des méthodes de la philosophie pour proposer des méthodes d'ingénierie d'ontologies.



Un exemple schématique d'ontologie : l'incontournable exemple des cubes.

Pour faire simple, considérons l'exemple de la figure ci-dessus. On y voit une certaine « scène du monde ». La description de cette scène demande deux choses : (1) un vocabulaire non ambigu, aussi appelé vocabulaire conceptuel ou ontologie ; (2) une énonciation des faits de la scène, reposant sur l'utilisation du vocabulaire de l'ontologie.

D'un point de vue pratique, une ontologie informatique permet, en particulier grâce aux travaux de l'intelligence artificielle symbolique sur les systèmes à base de connaissances et les moteurs d'inférence, d'implanter des mécanismes de raisonnement déductif, de classification automatique, de recherche d'information, et d'assurer l'interopérabilité entre plusieurs systèmes de ce type.

## 2. Que met-on dans une ontologie ?

Une ontologie définit des concepts (principes, idées, catégorie d'objet, notions potentiellement abstraites) et des relations. Elle inclut généralement une organisation hiérarchique des concepts pertinents et des relations qui existent entre ces concepts, ainsi que des règles et axiomes qui les contraignent.

L'ensemble des propriétés d'un concept s'appelle sa compréhension ou son *intension*, et l'ensemble des objets ou êtres qu'il englobe, son *extension*. Prenons un concept volontairement anonyme noté  $C$ , nous pouvons lui associer :

une intension : c'est un ensemble de propriétés qualitatives ou fonctionnelles communes aux individus auxquels le concept s'applique, et permettant de définir le concept, par exemple : «  $C$  est une sous-catégorie de véhicules de transports automobiles, conçus et aménagés pour le transport d'un petit nombre de personnes (7 ou moins) ainsi que d'objets de faible encombrement, et dotés d'au minimum trois roues » ;

une extension : un ensemble d'entités qui entrent dans cette catégorie, par exemple : { la Twingo de Rose, le Kangoo d'Olivier, la 306 d'Isabelle, la Clio d'Alain... }.

Pour exprimer, communiquer un concept, nous choisissons une représentation symbolique, souvent linguistique et verbale, parfois iconique. (Peirce distingue par exemple trois types de signes : l'indice, l'icone, le symbole.)

Symbole :

voiture

Icone :



Indice :



Dans le cas précédent, nous pouvons donner comme exemples de représentations linguistiques, les termes de « voiture », « automobile », « auto » ou « véhicule automobile », ou encore « tacot », « bagnole », « tire » ou « caisse ». Nous dissociions donc les concepts et leurs manifestations linguistiques. Un terme n'est pas un concept, et vice-versa. Un terme peut être ambigu, alors qu'un concept n'a qu'un seul sens, une seule définition. Il faut alors gérer les problèmes de synonymie (un concept dénoté par plusieurs termes) et d'homonymie (un terme dénotant plusieurs concepts).

De la même façon que pour les concepts, l'ontologie définit des relations pouvant exister entre les instances de ces concepts. Prenons une relation volontairement anonyme notée  $R$ , nous pouvons aussi lui associer :

une intension, par exemple : «  $R$  est une relation entre une personne ou un groupe qui a créé un document, et son contenu intellectuel, son arrangement ou sa forme » ;

une extension, par exemple : { (Hugo, Notre Dame de Paris), (Jean Markale, Le cycle du Graal)... } ;

des représentations linguistiques : « a écrit », « auteur de »...

Les relations possèdent en plus une « signature », une liste spécifiant les types d'instances qu'elles relient, soit pour notre exemple : (Personne ou groupe, Document).

## Avec les meilleures intensions...

Dans une ontologie, les intensions sont organisées, structurées et contraintes pour représenter notre conception du monde et de ses contraintes (par exemple, une voiture est forcément un véhicule). L'ontologie capture les intensions et les lois qui les régissent, afin de rendre compte des aspects de la réalité choisis pour

leur pertinence dans les scénarios d'application considérés. La représentation des intensions et de l'ontologie peut faire appel à des langages plus ou moins formels (graphes, logiques, langue restreinte), selon l'utilisation envisagée pour l'ontologie. La construction formelle de l'intension donne une représentation précise et non ambiguë de la manière dont on peut concevoir son sens, ce qui permet sa manipulation logique et son utilisation comme une primitive de représentation de connaissances pour décrire et structurer, par exemple, des données, des logiciels, des utilisateurs, des communautés, etc.

Dans l'ontologie, les intensions sont habituellement organisées en taxinomie ou hiérarchie de types. On appelle « subsomption » le fait de placer une catégorie sous une autre ; c'est aussi le lien qui en résulte entre la sous-catégorie subsumée et la catégorie mère. L'importance de l'organisation taxinomique se justifie par le fait que la classification ou identification (le fait de déterminer si quelque chose appartient à une classe) et la catégorisation (le fait d'identifier les catégories existantes) sont des inférences élémentaires que nous faisons à longueur de journée. Prenons l'exemple simple d'une conversation entre deux personnes :

« – Tu connais un *restaurant* proche ?  
 – Il y a une *pizzeria* au coin de la rue.  
 – Merci. »

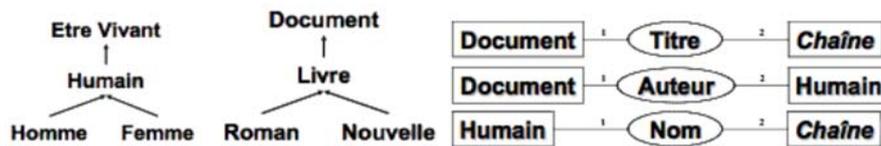
Dans une conversation aussi banale, la première personne a généralisé sa requête au concept de *restaurant*, qui représente la catégorie la plus abstraite recouvrant toutes les formes de réponses acceptables. La deuxième a, probablement sans même y prêter attention, utilisé sa taxinomie de concepts pour en déduire qu'une *pizzeria* est un *restaurant*, et que par conséquent sa réponse est pertinente. Le fait que cette connaissance taxinomique soit partagée est implicite, puisque la deuxième personne suppose que sa réponse sera comprise sans préciser qu'une *pizzeria* est un *restaurant*, et que c'est effectivement le cas. Le recours à des conceptualisations partagées et aux inférences qu'elles permettent est donc au cœur d'activités aussi simples que cet échange d'information. Le fait de rendre explicites les connaissances ontologiques et de s'assurer de leur nature consensuelle est un des problèmes majeurs de l'ingénierie ontologique.

Ainsi, dans un système d'information, le simple ajout de cette connaissance peut permettre d'améliorer considérablement les capacités des machines. Prenons l'exemple, très simplifié, où l'on recherche des livres écrits par un certain « Hugo ». Si votre système d'information se contente de travailler au niveau textuel, avec les mots clefs « Hugo » et « Livre », vous verrez apparaître plusieurs problèmes :

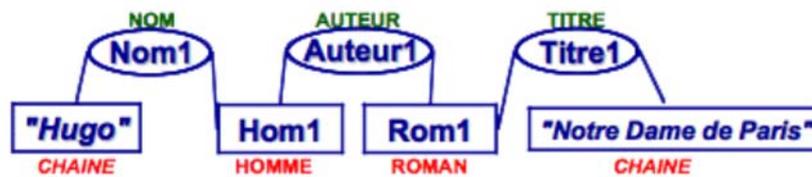
le bruit : le système ne saura pas faire la différence entre le nom de famille « Hugo », le prénom « Hugo » ou le nom de rue « Hugo » ;

le silence : le système, s'il rencontre le terme « R-o-m-a-n », ne saura pas qu'il est pertinent pour votre requête, car il cherche le mot « L-i-v-r-e ».

Si maintenant, vous expliquez au système quelques aspects de notre réalité sur les humains (Homme et Femme sont des sous-types d'Humain, qui est lui-même un sous-type d'Être Vivant), les documents (Roman et Nouvelle sont des sous-types de Livre, qui est lui-même un sous-type de Document) et les relations entre les deux, avec leurs signatures (par exemple, il existe une relation Auteur, qui peut s'établir entre un Document et un Humain.)...



puis que vous utilisez ce vocabulaire pour décrire la réalité, ici, qu'un homme dont le nom est « Hugo » a écrit un roman intitulé « Notre Dame de Paris »...



alors vous pouvez formuler une requête non ambiguë avec ce même vocabulaire pour rechercher les documents écrits par un certain « Hugo ».



Et en utilisant la logique de votre langage, le système peut inférer qu'un roman est un livre, un livre est un document, donc un roman est un document, et que la réponse « Hugo a écrit le roman Notre Dame de Paris » est valide.

## Ne pas confondre ontologie et taxinomie

En voyant cet exemple, il ne faut cependant pas confondre ontologie et taxinomie. Les connaissances ontologiques dépassent largement les connaissances taxinomiques. Ainsi, on peut trouver dans une ontologie :

des connaissances de composition, par exemple : en chimie (catégories d'éléments), en production (catégories de pièces), en médecine (catégories anatomiques), etc. ;

des définitions complètes, par exemple : une personne est un directeur si et seulement si il existe une organisation dirigée par cette personne ;

des contraintes d'intégrité, par exemple : un livre édité a un et un seul ISBN, un parent ne peut pas être plus jeune que ses enfants ;

des fonctions de calcul, par exemple : le rythme cardiaque conseillé pour une personne lors d'un effort cardio-vasculaire est égal à  $(220 - \text{âge}) \times 0.65$  ;

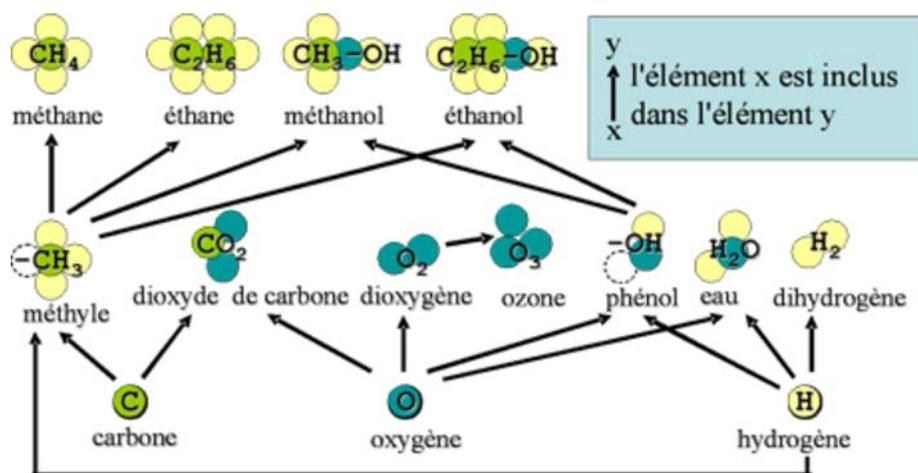
des propriétés algébriques, par exemple : la relation « est marié avec » est symétrique, cela signifie que si Thomas est marié avec Stéphanie, alors le système peut aussi déduire que Stéphanie est mariée avec Thomas, et vice-versa ;

des connaissances par défaut, par exemple : par défaut une voiture a quatre roues ;

des relations inverses, par exemple : « faire partie de » est l'inverse de « inclure », c'est-à-dire que si

une portière fait partie d'une voiture, alors la voiture inclut la portière, et vice-versa;

des règles spécifiques au domaine considéré, par exemple : en biologie, pour chaque récepteur qui active une fonction moléculaire, si cette fonction joue un rôle dans le fonctionnement de l'organisme, alors le récepteur joue le même rôle.



Exemple d'ontologie en chimie : composition de molécules.

Le contenu d'une ontologie varie aussi avec le type d'ontologie considéré. Une ontologie de domaine contiendra des connaissances propres à un domaine de connaissances (par exemple, l'aviation). Une ontologie de tâche contiendra des connaissances propres à une activité (par exemple, le diagnostic). Une ontologie de haut niveau contiendra des connaissances abstraites très générales, destinées à rassembler d'autres ontologies (par exemple, des notions d'entité, d'événement, de rôle, etc.). Le contenu dépendra aussi du degré de formalisation (langue naturelle, langage restreint, formalisme simple, logiques complexes).

## Pourquoi avoir séparé ces connaissances des autres ?

Les raisons en sont multiples. Tout d'abord, une ontologie permet de factoriser des connaissances. Dans un modèle, les connaissances ontologiques sont des connaissances toujours vraies, quels que soient l'état du système et les descriptions faites. L'ontologie permet de les factoriser et ainsi de ne pas avoir à les répéter pour chaque occurrence. Par exemple, on dira dans l'ontologie qu'une voiture est un véhicule (car c'est toujours vrai), mais on ne lui donnera pas une couleur, car cela change d'une voiture à une autre.

Un autre avantage est de pouvoir réutiliser et échanger des connaissances. Les connaissances ontologiques étant séparées, elles peuvent être réutilisées dans plusieurs applications, et ces réutilisations (totales ou partielles) peuvent constituer la base d'une interopérabilité entre différents systèmes.

Enfin, il est possible de compiler les connaissances et d'optimiser les inférences. Les connaissances ontologiques peuvent faire l'objet de traitements particuliers pour leur donner des structures efficaces, certifier leur cohérence et optimiser les inférences qui les exploitent. Par exemple, une ontologie permet le

calcul d'une fermeture transitive comme « si un coupé est une voiture, et une voiture est un véhicule, alors un coupé est un véhicule ».

## 3. Rendre l'implicite explicite : quelques applications

Beaucoup d'entités sociales ont à gérer et maintenir des connaissances, que ce soit leur raison d'être (réseaux d'intérêt, équipes de recherche, écoles, etc.) ou un résultat de leur fonctionnement (entreprises, administrations, associations...). De l'agilité de ces entités à détecter, mémoriser, se remémorer et activer leurs connaissances dépend leur agilité à répondre au monde extérieur (innovation des recherches, temps de réponse au marché, qualité des formations, etc.). Dans ce contexte, la connaissance est un capital et un système d'information performant est un atout primordial. Au royaume de l'information, les connaisseurs sont rois.

Mais les systèmes d'information collectifs sont des applications singulièrement contraintes. Les différences d'expérience, les différentes formations, les différentes cultures, les différents besoins, les différents points de vue, les différentes langues ou jargons, les différents médias et formats, les différents contextes d'utilisation, les différents droits d'accès, etc., posent une diversité de contraintes qui peuvent entraîner un système dans un cercle vertueux ou dans un cercle vicieux, selon leur degré d'importance pour les usages envisagés et la façon dont elles sont respectées par les solutions choisies.

## Des ontologies pour les systèmes d'information

L'introduction d'une ontologie dans un système d'information vise à réduire, voire éliminer, la confusion conceptuelle et terminologique et à tendre vers une compréhension partagée pour améliorer la communication, le partage, l'interopérabilité et le degré de réutilisation possible. Une ontologie informatique offre un cadre unificateur et fournit des « primitives », des éléments de base pour améliorer la communication entre les personnes, entre les personnes et les systèmes, et entre les systèmes.

Intégrer une ontologie à un système d'information permet donc de déclarer formellement un certain nombre de connaissances utilisées pour caractériser les informations gérées par le système et de se baser sur ces caractérisations et la formalisation de leur signification pour automatiser des tâches de traitement de l'information.

Dans un moteur de recherche c'est, par exemple, pouvoir améliorer la précision de la recherche d'information, en évitant des ambiguïtés au niveau terminologique (provenant de l'homonymie) ; le taux rappel de cette recherche d'information, en intégrant des notions plus précises ou équivalentes (en utilisant

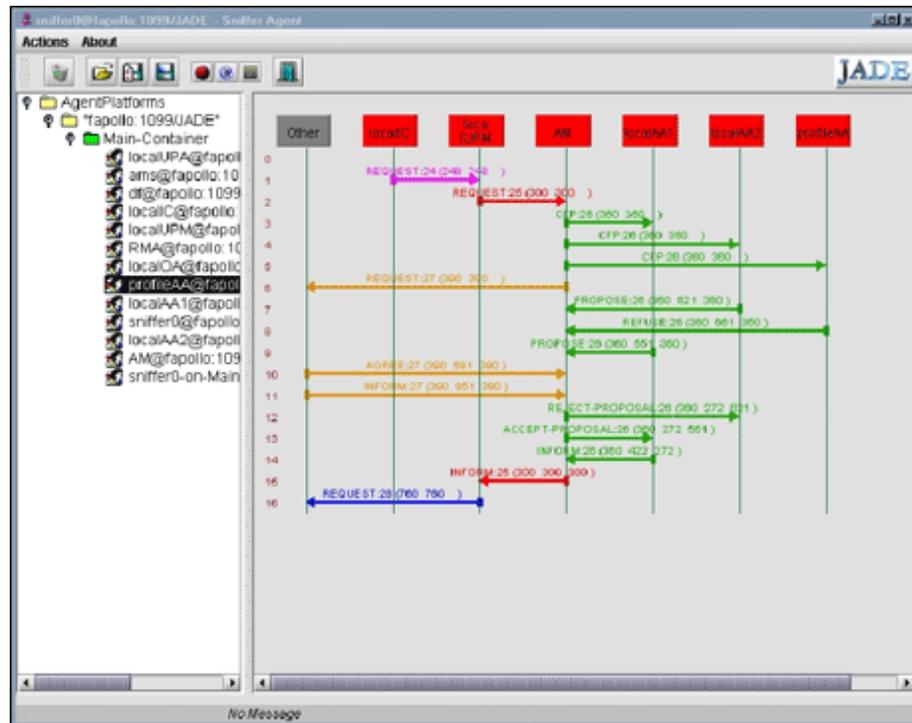
la synonymie, l'hyponymie) ou en déduisant des connaissances implicites (par exemple, des règles d'inférence) ; en relaxant des contraintes trop strictes en cas d'échec de la requête (par généralisation) ; en regroupant des résultats trop nombreux selon leur similarité pour les présenter de façon plus conviviale (regroupement ou clustering conceptuel). Le moteur de recherche Corese applique cette approche dans de nombreux domaines.

## Application à la gestion de documents dans le domaine du bâtiment

Le logiciel Aprobatom permet de guider la recherche de documents lors de la conception d'un nouveau bâtiment, afin d'identifier et de réutiliser des documents traitant de la réalisation de bâtiments ressemblant au nouveau projet. L'ontologie est utilisée pour guider l'utilisateur dans l'expression d'une requête (par exemple, en montrant les contraintes des signatures des relations, c'est-à-dire les instances qui peuvent être reliées). Elle sert aussi pour caractériser et comparer les différents documents gérés (par exemple, pour vérifier les types en prenant en compte la subsomption, c'est-à-dire les liens hiérarchiques entre catégories).

## Application à la distribution des connaissances

Dans une entreprise, les connaissances, leurs sources et leurs utilisations sont distribuées. Le système CoMMA a été conçu pour gérer l'annotation distribuée de documents.



Dans le système CoMMA, négociation entre agents logiciels gérant la distribution des connaissances (ici, des enchères pour décider de la meilleure base d'archivage pour une nouvelle annotation).

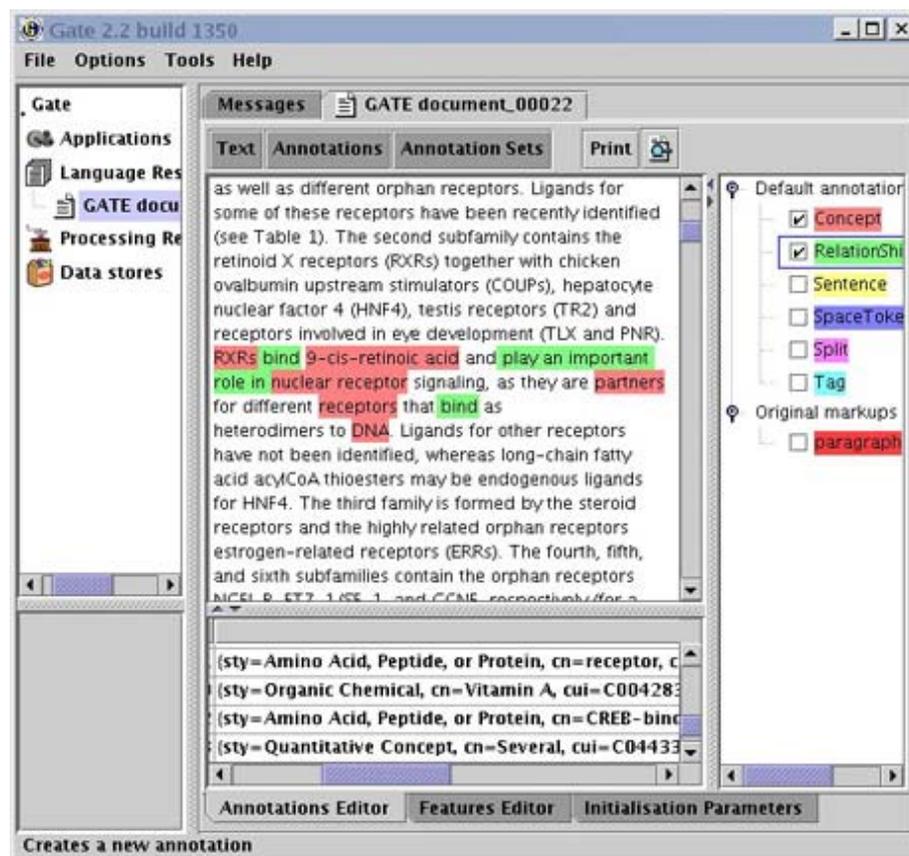
Prenons par exemple deux scénarios où l'ontologie est utilisée dans le système CoMMA. Le premier scénario met en scène un employé dans le cadre d'une activité de veille technologique, qui identifie et annote un document intéressant pour son entreprise. L'ontologie est ici utilisée pour caractériser ce document, archiver cette annotation avec des annotations similaires pour maintenir la spécialisation des archives, identifier les profils de personnes potentiellement intéressées par ce document et leur envoyer un message de notification. Pour définir formellement la notion de similarité, le graphe de subsomption de l'ontologie est utilisé comme espace métrique fournissant une distance pour évaluer, par exemple, la proximité sémantique de deux annotations.

Le deuxième scénario concerne un nouvel employé qui recherche des informations sur les activités et structures de son entreprise. Sa requête met en jeu différentes sources distribuées et l'ontologie est utilisée pour identifier les sources pertinentes et découper et distribuer la requête (par exemple, comparaison logique entre les contraintes de typage de la requête et les statistiques sur le contenu des bases en prenant en compte la hiérarchie des types et les signatures des relations).

## Application à la gestion d'une mémoire d'expériences

Dans une communauté comme celle de la biologie cellulaire, la quantité de publications scientifiques est telle que l'on conçoit aisément l'enjeu que représente l'intégration des connaissances pour améliorer la recherche d'information. Les communautés de la biologie et de la médecine sont d'ailleurs parmi les

pionnières du développement d'ontologies. Cependant, les larges corpus existant font qu'une méthode d'annotation purement manuelle ne passerait pas à l'échelle.



MeatAnnot utilise la plateforme de traitement de la langue naturelle GATE et des ontologies pour permettre l'annotation automatique d'articles en biologie.

En couplant les ontologies avec des outils d'analyse de la langue naturelle, l'outil MeatAnnot permet d'annoter formellement des textes avec des connaissances qu'ils décrivent. Par exemple, nous pouvons extraire d'un article une relation de causalité entre un gène et une maladie et annoter cet article avec cette connaissance, afin de l'utiliser d'une part dans des raisonnements d'analyses d'expériences et pour la confrontation de résultats, et d'autre part de garder une trace de la source de chaque connaissance.

Dans un premier temps, l'ontologie pilote donc l'analyse de texte en fournissant les termes à chercher (représentations linguistiques associées aux concepts et aux relations), le sens qui peut leur être associé (les intensions), et les structures qui peuvent être extraites (les signatures des relations); ces contraintes dirigent et focalisent l'analyse. Dans un deuxième temps, l'ontologie est utilisée, comme dans les exemples précédents, dans des inférences de recherche d'information.

## Application à l'enseignement assisté par ordinateur

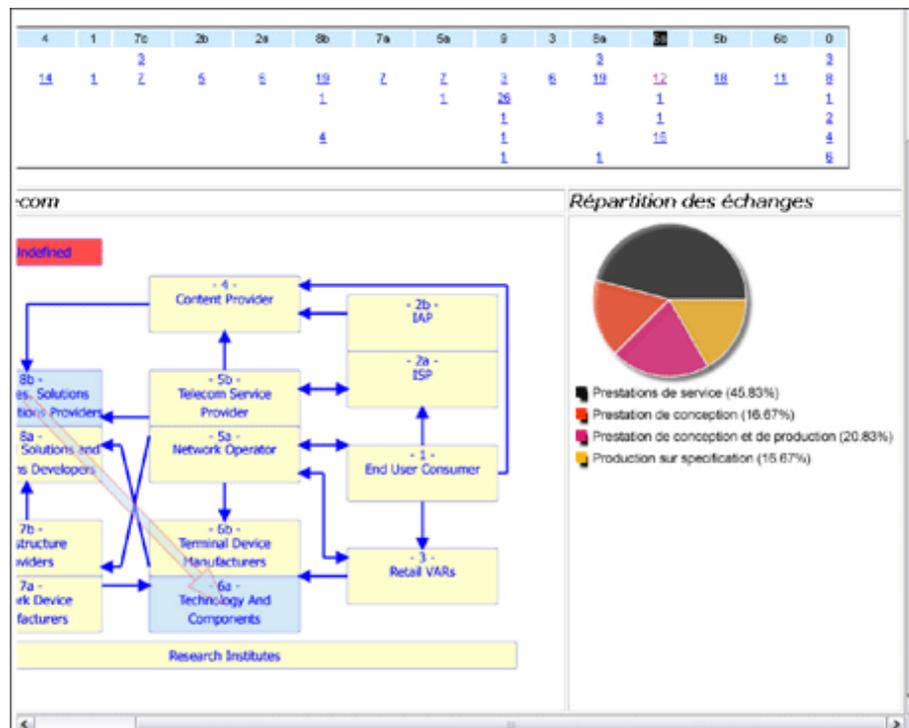
Chaque étudiant assistant à un cours l'assimile selon un parcours qui lui est propre. Les acquis et les impasses sont différents d'un étudiant à un autre. C'est pourquoi, dans le système QBL5, la structure d'un support de cours est enrichie à l'aide d'une ontologie des éléments pédagogiques (définition, théorème, exemple, question, etc.), afin de générer dynamiquement une interface de travaux dirigés ou de travaux pratiques assurant une grande souplesse de navigation et suivant la progression de l'étudiant dans l'ensemble des notions à acquérir. Dans cette application, l'ontologie définit et organise essentiellement les rôles des notions pédagogiques, par exemple elle définit les notions : « illustration », « définition », « exemple », etc. La sémantique de ces notions est utilisée dans la génération des supports pour distinguer les aspects fondamentaux (à présenter en priorité) des aspects qui ne le sont pas. La sémantique fixée permet aussi, d'un point de vue ergonomique, de définir des conventions pour les interfaces (par exemple, un code de couleurs) ou de définir formellement des structures récurrentes pour créer des vues abstraites (par exemple, rendre plus synthétique le graphe des parcours des étudiants sur un support pédagogique).

## Application au suivi médical et à la coopération autour du dossier d'un patient

Une décision thérapeutique peut impliquer plusieurs experts médicaux et peut aussi nécessiter une connaissance détaillée des antécédents du patient. Le système Ligne de vie repose sur une ontologie pour décrire les symptômes, les diagnostics, les options et les choix thérapeutiques, intégrer différentes contributions au dossier d'un patient et permettre une collaboration non ambiguë des personnes et des systèmes. L'ontologie est utilisée ici comme référentiel commun dans une activité collaborative.

## Application à la gestion des compétences

Ce dernier exemple s'applique à la Telecom Valley de Sophia Antipolis. Dans ce parc technologique, qui compte environ 70 membres, représentant plus de 10 000 emplois, la variété des compétences et des échanges possibles est non seulement élevée mais aussi dynamique et hautement spécialisée. Dans ce contexte, comprendre le paysage industriel pour les institutionnels régionaux ou trouver un partenaire pour les acteurs locaux sont de véritables problèmes d'intercompréhension et d'intégration d'informations.

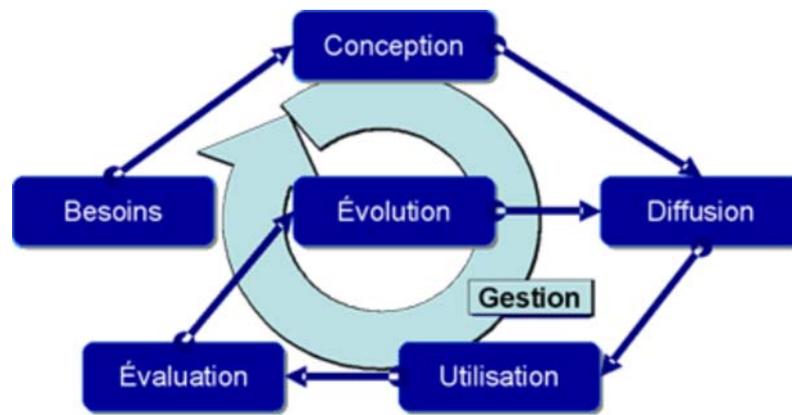


Dans le portail KmP, visualisation de l'analyse des échanges entre les membres de la Telecom Valley.

Le portail KmP permet la visualisation et l'analyse des échanges entre les membres de la Telecom Valley. Ce portail public, reposant sur une ontologie des compétences, permet à chaque acteur de décrire ses atouts et ses besoins afin d'améliorer la visibilité du parc et de faciliter de nouveaux partenariats. Là aussi, l'ontologie est utilisée pour piloter la génération d'interfaces, par exemple en fournissant des contraintes de typage pour la génération des choix multiples dans un formulaire. À partir des interfaces générées, les échanges peuvent être analysés, et le système peut trouver des partenaires possibles (inférences de types dans la recherche d'information), même si le partenaire parfait n'existe pas (utilisation de l'ontologie comme espace métrique pour la recherche approchée). Enfin, l'état du parc, ses points forts et ses niches, peuvent être visualisés en temps réel (utilisation de l'ontologie dans la définition des similarité dans un algorithme de clustering).

## 4. La vie rêvée des ontologies

Les ontologies sont des objets vivants, et chaque étape de leur cycle de vie pose des problèmes de recherche. Ce cycle de vie rassemble sept activités : détection des besoins, conception, gestion et planification, évolution, diffusion, utilisation, évaluation.



Le cycle de vie d'une ontologie.

## Besoins et évaluation

L'activité de détection des besoins, préalable à la conception, et l'activité d'évaluation, lorsqu'une ontologie est utilisée, posent des problèmes méthodologiques de recueil (analyse d'entretiens, questionnaires et sondages, étude de l'ergonomie et des usages) et d'identification (par exemple, modélisation par scénarios). En complément, la phase de détection des besoins demande un état des lieux initial approfondi, car elle ne peut reposer sur des études précédentes ou des retours d'utilisation, comme c'est le cas pour l'évaluation.

## Conception et évolution

La phase de conception initiale et la phase d'évolution ont elles aussi en commun un certain nombre de problèmes :

- spécification des solutions (conception participative, maquettage, prototypage) ;
- acquisition des connaissances nécessaires (analyse de textes, traitement automatique de la langue naturelle, plateformes collaboratives) ;
- conceptualisation et modélisation (*design pattern* ontologiques, méta-ontologies, entretien avec les experts) ;
- formalisation (méthodes et outils de l'Ontologie formelle, logiques de description et algorithmes de tableaux, analyse formelle de concepts, graphes conceptuels, formalismes du web sémantique RDF/S et OWL) ;
- intégration de ressources existantes (alignement automatique d'ontologies, traduction) ;
- implantation (graphes conceptuels, logiques de description, formalismes objets).

Un autre problème de conception et d'évolution est l'obtention et le maintien d'un consensus sur les choix de représentation et de conceptualisation faits dans l'ontologie. Suivant les usages, ce problème appelle des « collecticiels » et des outils de gestion des points de vue, des terminologies, des langues et des jargons

différents.

Notons aussi que l'évolution pose le problème de la maintenance de ce qui repose déjà sur l'ontologie. En effet, une ontologie est à la fois un objet vivant intéressant en soi et un ensemble de « primitives » pour décrire des faits du monde et des algorithmes sur ces faits. Lorsque l'ontologie change, ses changements ont un impact sur tout ce qui a été construit au-dessus. Le maintien de la cohérence dans une ontologie et au-dessus d'une ontologie, l'historique et la gestion des versions, la ré-ingénierie et la propagation des changements après modification, sont des questions de recherche encore largement ouvertes. La maintenance de l'ontologie soulève donc des problèmes d'intégration technique et des problèmes d'intégration aux usages.

## Diffusion

La phase de diffusion s'intéresse au déploiement et à la mise en place de l'ontologie. Les problèmes de cette phase sont fortement contraints par l'architecture des solutions. Dans un contexte d'application web, on reposera sur des technologies idoines. Pour le partage de fichiers, des architectures pair à pair ou autres architectures distribuées peuvent être utilisées. Pour l'intégration d'applications, des architectures de services web peuvent être une solution. Dans toutes ces architectures (serveurs web, services web, pair à pair, agents, etc.) la distribution des ressources (données, modèles, applications et utilisateurs) et leur hétérogénéité (syntaxes, sémantiques, protocoles, contextes, etc.) posent des problèmes de recherche sur l'interopérabilité (alignement et médiation) et le passage à l'échelle (larges bases, optimisation d'inférences, propagation de requêtes, syndication de données, composition de services, etc.).

## Utilisation

La phase d'utilisation regroupe toutes les activités reposant plus ou moins directement sur la disponibilité de l'ontologie, par exemple, l'annotation de ressources (traitement de la langue, rétroingénierie de base de données, etc.), la résolution de requête (algorithme de projection de graphes avec contraintes), la déduction de connaissances et l'aide à la décision (moteurs d'inférence à base de règles), la navigation assistée et les services contextuels (analyse de contexte, identification et composition de services), l'analyse de gros volumes de connaissances (clustering, recherche de motifs récurrents, veille).

Toutes ces activités ont en commun de poser le problème de la conception des interactions avec l'utilisateur et de leur ergonomie (interfaces dynamiques, lien sémiotique-sémantique, profils et contextes d'utilisation). Sur ce point, l'ontologie apporte à la fois de nouvelles solutions (par exemple, les inférences exploitent les ontologies pour la génération dynamique d'éléments d'interfaces) et de nouveaux problèmes (par exemple, la complexification des modèles de données engendre des problèmes pour leur représentation et l'interaction avec ces représentations).

# Gestion

L'activité permanente de gestion et planification souligne qu'il est important d'avoir un travail de suivi et une politique globale pour détecter ou déclencher, préparer et évaluer les itérations du cycle et s'assurer que l'on reste dans le cercle vertueux des systèmes d'information (où se succèdent contribution, utilisation, création).

## 5. Une notion pleine d'avenir

La notion d'ontologie, qui précède largement l'utilisation du mot, ne semble pas prête à disparaître. Au contraire, le spectre d'applications et de domaines s'intéressant aux ontologies ne cesse de s'élargir.

Anciennement réservée aux systèmes experts simulant des raisonnements humains dans des domaines spécifiques, l'ontologie se retrouve maintenant dans une large famille de systèmes d'information. Elle est utilisée pour décrire et traiter des ressources multimédia ; asseoir l'interopérabilité d'applications en réseaux ; piloter des traitements automatiques de la langue naturelle ; construire des solutions multilingues et interculturelles ; permettre l'intégration de sources hétérogènes d'information ; décrire des protocoles d'interactions complexes ; vérifier la cohérence de modèles ; permettre les raisonnements temporel et spatial ; faire des approximations logiques ; etc.

Ces utilisations des ontologies se retrouvent dans de nombreux domaines d'application : intégration d'informations géographiques, gestion de ressources humaines, aide à l'analyse en biologie, commerce électronique, enseignement assisté par ordinateur, bibliothèques numériques, échanges commerciaux entre partenaires industriels, suivi médical informatisé, etc.

## Le web sémantique

Un courant particulièrement prometteur pour l'expansion des systèmes à base d'ontologies est celui du web sémantique. Il s'agit d'une extension du Web actuel, dans laquelle l'information se voit associée à un sens bien défini, améliorant la capacité des logiciels à traiter l'information disponible sur le web. L'annotation de ces ressources d'information du web repose sur des ontologies elles aussi disponibles et échangées sur le web. Grâce au web sémantique, l'ontologie a trouvé un formalisme standard à l'échelle mondiale et s'intègre dans de plus en plus d'applications web, sans même que les utilisateurs ne le sachent. Cela se fait au profit des logiciels qui, à travers les ontologies et les descriptions qu'elles permettent, peuvent proposer de nouvelles fonctionnalités.

De ce fait, de plus en plus d'ontologies de domaines sont disponibles : ontologie de la génétique, ontologie de la géométrie, ontologie pour les musées, ontologie médicale, ontologie pour l'enseignement, ontologie pour le bâtiment, ontologie de systèmes documentaires, ontologie pour la gestion, ontologie dans le secteur automobile, etc.

À l'heure où l'ontologie se dote d'une ingénierie, cette expansion est loin d'être finie. Parmi ses dernières évolutions, l'ontologie qui s'appliquait essentiellement à des données (documents, images, vidéos) est maintenant utilisée pour décrire des logiciels (par exemple, des services web), leurs caractéristiques fonctionnelles (types d'entrées, types de sorties), et non fonctionnelles (coût, qualité). Elle pourrait ainsi permettre l'identification, l'invocation et la composition dynamique d'applications à l'échelle du web.

De même, l'ontologie commençait déjà à être utilisée pour décrire les utilisateurs et s'étend maintenant à la description du contexte d'interaction, pour doter les applications de ce que l'on appelle une conscience du contexte. Cela concerne les préférences de l'utilisateur (langue, goûts, droits, etc.), les caractéristiques du terminal (mobile, vocal, etc.), la situation géographique (à l'étranger, dans une salle avec imprimante, etc.), l'activité en cours (au volant, en présentation, etc.), l'historique d'utilisation.

Enfin, si l'ontologie est actuellement utilisée pour faciliter l'accès à des informations et des applications, on pressent aussi son utilisation dans la description et l'application de règles de sécurité et de confidentialité décrites à de hauts niveaux d'abstraction, permettant de restreindre les accès avec une grande flexibilité. Ainsi, dans un système d'information à base d'ontologies, la confidentialité et ses règles reposent aussi sur la sémantique des ontologies et les inférences qu'elle permet pour contrôler l'accès à l'information et la précision de l'information diffusée.

## À quand la programmation orientée ontologie ?

Nos systèmes d'information sont de plus en plus complexes. Cette complexité, même si elle est artificielle, puisqu'il s'agit de technologie, pose des défis scientifiques ardues qu'il nous faudra relever pour voir l'expansion technologique continuer, par exemple, dans l'assistance des sciences de la vie.

La possibilité de concevoir des systèmes qui se reconfigurent, s'adaptent au contexte, détectent leurs fautes, et même se corrigent dans une certaine mesure, apparaît comme un facteur de passage à l'échelle pour la croissance technologique. Rendre explicites les conceptualisations du monde sur lesquelles se basent les architectures logicielles, les structures de données, les choix de conception, c'est aussi participer à cette évolution des applications et de leur programmation.

Le défi actuel des ontologies est de passer dans les pratiques d'ingénierie logicielle, pour que les conceptualisations actuellement sous-jacentes, en filigrane dans le code, dans ses commentaires ou dans sa documentation dans le meilleur des cas, soient le plus souvent possible rendues explicites et capturées dans des formalismes. Ainsi exposées, elles offriraient des possibilités d'inférence aux systèmes informatiques, de réflexivité sur leurs connaissances et leurs traitements, d'introspection, d'alignement dynamique pour

permettre l'interopérabilité, d'évolution dynamique et de description de l'« affordance » des composants logiciels, c'est-à-dire leur capacité à indiquer comment s'interfacer avec eux et les utiliser, afin de rendre leurs interactions plus dynamiques et leur gestion plus automatique, et d'aller vers une informatique plus autonome.

Alors que les applications web s'infiltrant dans tous nos systèmes d'informations, le web est définitivement passé du statut de base documentaire à celui d'une machine virtuelle universelle combinant des ressources de tous horizons. On peut imaginer un nouveau paradigme de programmation, où les structures de données seraient des représentations basées sur des ontologies partagées, et où les applications seraient obtenues par composition de services (logiciels personnels, services en ligne, appels à des grilles, etc.).

Après la programmation orientée objet, la « programmation orientée ontologie » ?

*Remerciements à tous les membres de l'équipe ACACIA, en particulier aux autres membres permanents (Olivier Corby, [Rose Dieng-Kuntz](#), Alain Giboin) et aux doctorants dont les travaux sont cités ici (Sylvain Dehors, Khaled Khelif).*