Les représentations des connaissances par objets

Amedeo Napoli* — **Jérôme Euzenat**** — **Roland Ducournau*****

* *LORIA — B.P. 239, 54506 Vandoeuvre les Nancy*
** *INRIA Rhône-Alpes — 655 avenue de l'Europe, 38330 Montbonnot-Saint Martin*
*** *LIRMM — 161, rue Ada, 34392 Montpellier Cedex 5*
napoli@loria.fr, euzenat@inrialpes.fr, ducournau@lirmm.fr

RÉSUMÉ. *La finalité des systèmes de représentation des connaissances par objets est de représenter des connaissances autour de la notion centrale d'objet. Cet article décrit l'origine et l'évolution de ces systèmes, ainsi que la place et l'avenir qui leur semblent réservés.*

ABSTRACT. *The goal of object-based knowledge representation systems is to represent knowledge in an object-oriented way. This paper describes the origin and the evolution of these systems, as well as their place and the future that seems to be associated with these systems.*

MOTS-CLÉS : *Représentation des connaissances par objets, raisonnement, système classificatoire, logique de descriptions, gestion des connaissances, objet, inférence de valeur, classification.*

KEYWORDS: *Object-based knowledge representation, reasoning, classification scheme, description logics, knowledge management, object, inference, classification.*

1. Introduction et historique

Représenter des connaissances. En toute généralité, représenter des connaissances propres à un domaine particulier consiste à décrire et coder les entités de ce domaine sous une forme qui puisse être exploitée en machine [KAY 97]. Un système de représentation des connaissances est utilisé pour mener à bien des raisonnements et résoudre des problèmes dans le domaine étudié. Il existe plusieurs familles de langages de représentation des connaissances et un certain consensus semble s'être fait sur ce que doivent être les caractéristiques d'un tel langage : les expressions du langage reposent sur une *syntaxe* — un procédé de construction de formules bien formées — et une *sémantique*. La sémantique fournit un sens aux formules et justifie la validité des opérations effectuées par le système de représentation sur les formules, opérations qui s'appliquent ainsi en adéquation avec le sens des formules. À l'heure actuelle, parmi les familles de langages de représentation, se détachent les formalismes de représentations qui s'expriment directement en logique (des propositions, des prédicats, modale, non monotones, etc.), les logiques de descriptions (LD) [NEB 90, DON 96] les graphes conceptuels [SOW 84] et enfin les représentations des connaissances par objets (RCO). C'est sur ces derniers formalismes qu'essaie de faire brièvement le point cet article.

Objets. Le succès des « objets informatiques » est justifié habituellement par les qualités que recherche et promeut le génie logiciel : modularité, extensibilité ou réutilisabilité sont des références obligées. Pourtant, une autre raison est sans doute aussi importante : les « objets informatiques » ont une capacité naturelle de représentation « des objets du monde réel ». Cette capacité est non seulement à l'origine de la représentation des connaissances par objets en intelligence artificielle, mais aussi des méthodes d'analyse et de conception en génie logiciel ou encore des bases de données à objets [CAL 99]. Ainsi, parallèlement à la croissance de l'approche objet en génie logiciel, une famille de langages de représentation s'appuyant sur des principes similaires mais empruntant certaines de leurs caractéristiques aux LD — essentiellement leur traitement de la sémantique des expressions — a commencé à se développer pour conduire aux systèmes de RCO et aux systèmes classificatoires [DUC 98].

Histoire. Historiquement, les formalismes généraux de représentation des connaissances ont été nombreux [BRA 85, SOW 91]. Parmi eux, le formalisme des réseaux sémantiques se caractérise par une organisation des connaissances en un graphe orienté dont les sommets et les arcs sont étiquetés. Le formalisme ultérieur des *frames* — ou schémas — organise les connaissances autour de la notion de *frame* [MAS 89] : un nom auquel est associé un ensemble d'*attributs*, chaque attribut se voyant à son tour associer un ensemble de *facettes* permettant de le caractériser.

Cependant, ces formalismes sont dotés d'une sémantique peu claire et non véritablement fixée, ce qui conduit les utilisateurs à commettre certains péchés : (i) considérer la même entité, tantôt comme un individu, tantôt comme une classe, (ii) confondre propriétés descriptives et propriétés typiques, (iii) confondre la *définition* d'une classe donnée par des conditions nécessaires et suffisantes d'appartenance à la classe (notion de classe *définie*), la *description* d'une classe où les conditions d'appartenance sont seulement nécessaires (notion de classe *primitive*), et le *prototype* où les conditions sont typiques et données avec un degré de plausibilité ou de typicité. C'est sous l'impulsion de William Woods et Ronald Brachman [BRA 85] que ces problèmes ont été clairement mis en avant et que des solutions ont été recherchées. Les travaux initiaux ont mené à la définition du formalisme des *logiques de descriptions*, qui traite de manière satisfaisante les questions de sémantique et d'inférences, en introduisant notamment la relation de subsomption et le processus de classification.

La sémantique des langages de représentation a mis du temps à s'imposer en tant que telle : l'évolution entre les premiers travaux relativement peu formels jusqu'aux logiques de descriptions a été lente. Aussi, le développement de systèmes a continué à progresser, sans soucis de description formelle des systèmes, mais sous la houlette de concepteurs conscients de l'importance d'une définition claire des langages.

Les *langages hybrides* s'inspirent des réseaux sémantiques, des *frames*, mais aussi des langages de programmation par objets (LPO) [MAS 89] : la notion de classe et la distinction classe – instance sont adoptées ; la relation de *spécialisation* permet d'organiser les classes en un *graphe (d'héritage)* ; la définition et l'utilisation des facettes sont clarifiées et rationalisées (typage et calcul des valeurs des attributs). L'héritage, qui gère le partage de propriétés dans le graphe des classes, et l'attachement procédural sont les moyens d'inférence privilégiés.

Vers les représentations des connaissances par objets. C'est à partir de ces langages hybrides que s'est développée la notion de (système de) représentation des connaissances par objets, comme un compromis entre les langages de programmation par objets et les logiques de descriptions, avec les concepts et les fonctionnalités procédurales des premiers et les aspects sémantiques des secondes. La notion de RCO se caractérise plus par la disponibilité et l'application de systèmes que par un corpus théorique uniforme. Mais la recherche théorique sur la RCO évolue vers une formalisation croissante qui conduit à définir une sémantique dénotationnelle des systèmes de RCO, à évaluer la décidabilité et la complexité des mécanismes d'inférences disponibles, et à concevoir des algorithmes corrects pour ces mécanismes. Cependant, l'effort de formalisation ayant poussé à une simplification des systèmes existants, de nombreuses opérations et innovations — souvent très utiles sur le plan pratique — sont laissées dans l'ombre par le travail de formalisation. Même les systèmes développés sur la base des travaux théoriques offrent des traits qui sortent de la théorie, mais qui rendent le système utile. Il est donc important de s'intéresser à tous les aspects d'un système de RCO et non uniquement à ceux qui sont dûment formalisés.

2. Les systèmes de représentation des connaissances par objets

À proprement parler, il n'y a pas encore aujourd'hui de principes intangibles sur lesquels reposent les systèmes de RCO. Il est cependant possible de décrire à gros traits ce qu'est un système typique. La fonction d'un tel système est de stocker et d'organiser les connaissances autour de la notion d'objet et de fournir des services inférentiels destinés à compléter l'information disponible.

Objets et classes. Les entités élémentaires manipulées par les RCO sont proches de celles d'un LPO [CAL 99]. Les objets élémentaires du domaine étudié sont décrits par des « objets informatiques » regroupés en classes, classes qui sont elles-mêmes organisées en un graphe. Un objet possède une identité et se décrit par un ensemble de couples attributs-valeurs. La valeur d'un attribut peut être constituée d'un seul élément ou d'un ensemble d'éléments ; elle peut être connue ou pas. Cette valeur peut relever d'un type primitif comme *entier*, *booléen*, ou *chaîne de caractères*, ou être une instance d'une autre classe : dans ce cas, l'attribut matérialise une *relation* entre la classe à laquelle il est attaché — le *domaine* de la relation — et la classe de l'instance qui constitue sa valeur, le *co-domaine* de la relation. Dans les classes, des facettes (ou descripteurs) sont associées aux attributs. Les facettes sont en nombre restreint et leur rôle est clairement identifié : elles servent principalement à préciser le type des valeurs des attributs, à introduire des contraintes ou des mécanismes capables de calculer la valeur d'un attribut, à gérer les relations et leurs inverses. Ainsi, les facettes de typage permettent de restreindre l'ensemble des valeurs possibles d'un attribut — le *domaine* de l'attribut — en précisant les valeurs admissibles par une énumération de valeurs ou une union d'intervalles, en introduisant des exceptions, en restreignant la cardinalité dans le cas de collections de valeurs, ou encore en déclarant un filtre n'acceptant que des instances satisfaisant les conditions du filtre. En outre, certaines facettes spécifient des comportements *réflexes* exécutés dès que la valeur d'un attribut est modifiée, lue ou écrite, ce qui est la base de la *programmation dirigée par les accès*.

Spécialisation, héritage et sous-typage. Les classes sont organisées par l'intermédiaire d'une relation d'ordre partiel, la *spécialisation*, parfois nommée relation d'héritage [LEN 91]. Le graphe de la relation peut être un graphe orienté sans circuit ou un arbre. Si c'est un arbre, une classe hérite les propriétés de son unique ascendant direct — sa *super-classe* — et l'héritage est dit simple. Si c'est un graphe, une classe peut hériter les propriétés de plusieurs super-classes et l'héritage est dit multiple [DUC 95]. La sémantique de la relation de spécialisation est celle de l'inclusion ensembliste : une classe s'interprète comme un ensemble d'individus et les individus appartenant à l'interprétation d'une classe doivent appartenir à celle des super-classes ; les ensembles d'individus sont imbriqués parallèlement à la spécialisation des classes. De cette interprétation découlent les principes suivants : (i) plus une classe est spécifique, plus les domaines de ses attributs sont restreints ; (ii) si une classe est sous-classe de deux classes, les domaines de ses attributs sont forcément inclus dans (ou égaux à) l'intersection des domaines de ces deux classes ; (iii) si un objet est instance d'une classe, il l'est aussi de toutes les super-classes, etc. Le point (ii) représente une opposition importante entre les RCO et les LPO. Pour reprendre le vocabulaire de la théorie des types, il s'agit d'une « redéfinition covariante » des attributs, alors que ladite théorie des types impose une « redéfinition invariante » pour les attributs, et même contravariante pour les arguments des méthodes [CAS 96]. Cette opposition peut s'expliquer par l'argument suivant : la théorie des types s'intéresse plutôt à la substituabilité d'un type par un sous-type, ce qui sous-entend une quantification universelle. En revanche, la RCO s'intéresse plutôt à rechercher la valeur d'un attribut, ce qui sous-entend une quantification existentielle.

Inférence de valeurs. Un autre trait qui différencie les LPO des systèmes de RCO est la présence de fonctionnalités liées au raisonnement. Les opérations principales qui sont à la base du raisonnement sont : (i) le partage de propriétés et le calcul de la valeur d'un attribut, (ii) le maintien de la cohérence des attributs interdépendants, (iii) le test de spécialisation et la classification. Le partage de propriétés se fait de façon classique par l'intermédiaire du mécanisme d'héritage. Le calcul de la valeur d'un attribut se fait quant à lui par le biais des facettes d'inférences, dont les plus courantes sont : (1) la valeur par défaut qui fournit une valeur donnée en l'absence d'autre valeur, (2) le passage de valeur qui contraint la valeur cherchée à être celle d'un autre attribut, (3) le filtrage qui donne pour valeur un ensemble d'objets satisfaisant certaines contraintes, (4) l'attachement procédural qui provoque l'exécution d'une fonction de calcul retournant la valeur cherchée. Ces mécanismes peuvent être attachés simultanément à un même attribut et il faut alors décider d'une stratégie pour les appliquer dans un certain ordre.

Des problèmes de *conflits d'héritage* peuvent apparaître dans le cas de l'héritage multiple [DUC 95] : de quelle super-classe hériter lorsqu'il existe plusieurs super-classes incomparables ? Le problème est très discuté dans le monde des LPO, mais aussi dans le monde de l'intelligence artificielle pour les problèmes liés à la non monotonie (l'ajout d'information peut venir invalider ce qui était tenu pour valide).

Classification. Le test de spécialisation consiste à vérifier qu'une classe en spécialise une autre sur la base de la structure des deux classes. Ce test est la base du processus de classification qui, avec une interprétation définitionnelle, se présente sous deux formes : (i) la *classification de classes* consiste à placer une nouvelle classe dans le graphe des classes en recherchant ses super-classes les plus spécifiques et ses sous-classes les plus générales, (ii) la *classification d'instances* qui permet d'identifier la classe d'un individu. Une classe ou un individu peuvent donc être créés avec des classes de référence très générales, à partir desquelles seront retrouvées les classes de référence les plus spécialisées possibles, avec le processus de classification. Le mécanisme de classification n'est pas présent dans tous les systèmes de RCO, mais est toutefois devenu un mécanisme de raisonnement incontournable pour tout système se voulant opérationnel.

Méthodes et messages. Dans ce qui précède, il n'a pas été question de méthodes attachées aux classes comme il s'en trouve dans les langages hybrides et les LPO, car ce point reste encore un sujet de controverse. Lorsque des méthodes sont attachées aux classes, leur fonctionnement est le même que dans les LPO et peut s'envisager sur plusieurs niveaux selon les effets de bords résultants, qui sont ou ne sont pas contrôlables dans le cadre de la sémantique standard. Un premier niveau peut être dédié à des services qui ne produisent pas d'effets de bords comme l'impression, la visualisation, la recherche de caractéristiques et éventuellement l'instanciation. Un second niveau peut être dédié à des méthodes de calcul produisant des effets de bords sur les classes et les instances ; la responsabilité de l'emploi de ce dernier niveau est à la discrétion de l'utilisateur. Le débat sur de tels principes reste toutefois ouvert.

Systèmes classificatoires. À l'instar des LD, un système de RCO peut se voir comme un système logique, qui fournit un langage de description des classes et des individus, et des procédures d'inférences qui reposent sur la spécialisation. Une fonction d'interprétation est définie par induction sur la structure du langage. Elle donne la sémantique attachée à une classe : cette sémantique correspond à l'ensemble des objets faisant partie de l'extension de la classe. Les faits dans lesquels interviennent les individus sont pris en compte par l'intermédiaire d'un ensemble d'*assertions*. Ces caractéristiques constituent la définition d'un *système classificatoire*, qui se veut une abstraction des diverses approches prises par les systèmes de RCO [DUC 98]. La justification de « classificatoire » est double : ces systèmes offrent une *classification des espèces*, au sens du graphe des classes, et ils se prêtent à une *classification des individus*, au sens du mécanisme d'inférences qui reconnaît qu'un individu appartient à une classe. La théorie des systèmes classificatoires essaie d'établir une comparaison constructive avec la théorie des LD, mais aussi de clarifier et de formaliser les capacités inférentielles effectives des systèmes de RCO, en particulier en étudiant l'incertitude qui résulte fatalement de tentatives de classification avec des classes primitives. La théorie des systèmes classificatoires revêt une importance considérable car elle vise à fournir une référence formelle sur laquelle un concepteur ou un utilisateur peuvent s'appuyer pour comprendre et mesurer les capacités d'un système de RCO.

3. Les futurs de la représentation des connaissances par objets

Jusqu'à présent, la RCO est restée confinée à des systèmes très particuliers. Cela tient sans doute en partie à la difficulté — et donc au coût — de créer des bases de connaissances, ce qui, contrairement à l'alimentation de bases de données, est consommateur de ressources intellectuelles : la gestion de ces ressources constitue justement la problématique de la gestion des connaissances.

Représentation et gestion des connaissances. Le futur des systèmes de RCO passe certainement par ce qu'il est convenu d'appeler maintenant la *gestion des connaissances* et les *systèmes intégrés de traitement de l'information* [FOU 97, CHO 98]. Le but de tels systèmes est d'intégrer et de rendre accessibles données, connaissances et savoir-faire d'une organisation (mais aussi ceux provenant de l'extérieur). Ils entérinent la convergence des technologies de représentation des connaissances, de gestion de bases de données et de gestion de documents. La diffusion s'appuie principalement sur les technologies de l'Internet (même pour un système interne). Ces systèmes ne sont que la face technologique d'une évolution organisationnelle.

Si l'on se borne à considérer l'aspect informatique, il est possible de distinguer, en fonction de leur provenance, deux types d'information à manipuler : (i) une information interne qui devrait pouvoir être maîtrisée, (ii) une information externe qui ne peut pas être maîtrisée tant du point de vue de sa structure, que de celui de son contenu ou de la confiance qui peut lui être accordée. Ces deux types d'information peuvent partager leur support et leur format, mais ils sont fondamentalement différents : du point de vue de l'utilisateur, le premier a (devrait avoir) un caractère organisé et le second un caractère désorganisé. Il s'ensuit que leur impact sur les RCO est aussi différent.

Représentation et acquisition des connaissances. La connaissance provenant de l'extérieur, mais aussi parfois de l'intérieur, peut être considérée comme plutôt textuelle, bruité voire contradictoire, et non structurée ou « semi-structurée ». Afin de la synthétiser, il faut faire appel à des techniques linguistiques et statistiques de gestion et de recherche d'information. En particulier, des modules de fouilles de données s'appuyant sur des techniques symboliques et numériques d'analyse des données peuvent être utilisés. De plus, en présence de sources d'information hétérogènes, il faut aussi être en mesure de prendre en compte des *données semi-structurées*, un autre enjeu majeur à l'heure actuelle. Du point de vue des systèmes de RCO, cela peut impliquer d'intégrer plus de souplesse ou plus d'expressivité, afin de prendre en compte — entre autres — la disjonction de classes, les objets sans classes et la multi-instanciation (le fait qu'un objet puisse être une instance de plusieurs classes), points qui ne font généralement pas partie des systèmes actuels de RCO.

Représentation et diffusion des connaissances. La diffusion de la connaissance sous forme de documents — fussent-ils HTML — est un obstacle à leur réappropriation par un système informatique et nécessite le développement des technologies évoquées au paragraphe précédent. Or, la communication de la connaissance formalisée, ou, au pire, structurée, est utile pour exploiter la connaissance acquise dans un système à base de connaissances, pour manipuler de la connaissance distribuée, pour que des agents tirent parti de cette connaissance ou pour la recherche d'un document sur la base de son contenu. Si la connaissance est formalisée, il vaut mieux la diffuser sous cette forme plutôt que sous la forme de documents peu structurés. Mais, afin d'assurer l'interopérabilité, il faut être capable de comprendre le sens de la connaissance communiquée. Cela pose des problèmes importants auxquels la recherche en représentation des connaissances s'attelle :

– en ce qui concerne la terminologie utilisée, les travaux se penchent sur la conception et la diffusion de modèles objets communs ou « ontologies » [GUA 98] ;

– en ce qui concerne les formalismes de représentation des connaissances, il semble acquis que la structure sera diffusée à l'aide du langage XML. Mais qu'en est-il de la sémantique ? Devant la faillite de l'adoption de langages standards, due à la diversité des besoins, et le développement de nombreux langages de RCO, il semble nécessaire d'élaborer un moyen de diffuser une description de la sémantique des formalismes, comme est diffusée une description de leur syntaxe, qui puisse être appréhendée par la machine.

4. Bibliographie

[BRA 85] BRACHMAN R., LEVESQUE H., Eds., *Readings in Knowledge Representation*, Morgan Kaufmann Publishers, Los Altos (CA), USA, 1985.

[CAL 99] CALVANESE D., LENZERINI M., NARDI D., « Unifying Class-Based Representation Formalisms », *Journal of Artificial Intelligence Research*, vol. 11, 1999, p. 199–240.

[CAS 96] CASTAGNA G., « Le modèle fondé sur la surcharge : une visite guidée », *Technique et science informatiques*, vol. 15, n° 6, 1996, p. 673–708.

[CHO 98] CHOMICKI J., SAAKE G., Eds., *Logics for databases and Information Systems*, Kluwer Academic Publishers, Boston (MA), USA, 1998.

[DON 96] DONINI F.-M., LENZERINI M., NARDI D., SCHAERF A., « Reasoning in Description Logics », BREWKA G., Ed., *Principles of Knowledge Representation*, p. 191–236, CSLI Publications, Stanford (CA), USA, 1996.

[DUC 95] DUCOURNAU R., HABIB M., HUCHARD M., MUGNIER M.-L., NAPOLI A., « Le point sur l'héritage multiple », *Technique et science informatiques*, vol. 14, n° 3, 1995, p. 309–345.

[DUC 98] DUCOURNAU R., EUZENAT J., MASINI G., NAPOLI A., Eds., *Langages et modèles à objets — État des recherches et perspectives*, Collection Didactique D-019, INRIA, Le Chesnay, 1998.

[FOU 97] FOUET J.-M., Ed., *Connaissances et savoir-faire en entreprise*, Hermès, Paris, 1997.

[GUA 98] GUARINO N., Ed., *Formal Ontology in Information Systems*, IOS Press, Amsterdam, 1998.

[KAY 97] KAYSER D., *La représentation des connaissances*, Hermès, Paris, 1997.

[LEN 91] LENZERINI M., NARDI D., SIMI M., Eds., *Inheritance Hierarchies in Knowledge Representation and Programming Languages*, John Wiley & Sons, Chichester (West Sussex), UK, 1991.

[MAS 89] MASINI G., NAPOLI A., COLNET D., LÉONARD D., TOMBRE K., *Les langages à objets*, InterEditions, Paris, 1989.

[NEB 90] NEBEL B., *Reasoning and Revision in Hybrid Representation Systems*, Lecture Notes in Artificial Intelligence 422, Springer-Verlag, Berlin, 1990.

[SOW 84] SOWA J., *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, Reading (MA), USA, 1984.

[SOW 91] SOWA J., Ed., *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, Morgan Kaufmann Publishers, San Mateo (CA), USA, 1991.

***Amedeo Napoli** est chargé de recherches CNRS au LORIA (Nancy). Il s'intéresse depuis longtemps à l'approche objet sous toutes ses formes et particulièrement aux systèmes de* RCO*, du point de vue de la représentation des connaissances et du raisonnement. Les aspects théoriques et les liens avec les* LD*, mais aussi l'extraction de connaissances dans les bases de données, la conception de systèmes intégrés de traitement de l'information et la gestion de données semi-structurées font partie de ses préoccupations actuelles.*

***Jérôme Euzenat** est chargé de recherches à l'INRIA Rhône-Alpes. Il a dirigé, au sein du projet* SHERPA*, le développement et l'application du modèle de représentation des connaissances* TROEPS *tout en poursuivant des travaux sur les fondements des modèles de* RCO*. Ses travaux actuels concernent les relations qu'entretiennent plusieurs représentations, qu'elles soient ou non décrites dans le même langage, afin d'assembler des représentations (travail collaboratif) ou encore d'engendrer des représentations dans un but précis (serveurs de connaissance).*

***Roland Ducournau** est professeur à l'Université Montpellier II depuis 1994. Il était auparavant ingénieur dans la SSII Sema Group, où il a conçu et réalisé l'environnement de développement de systèmes à base de connaissances* Y3 *autour du langage d'objets* YAFOOL*. Son domaine de recherche est « l'objet », sous toutes ses formes, mais surtout du point de vue de ses capacités de représentation : en représentation des connaissances, bien sûr, mais aussi en programmation ou en bases de données. La structure classificatoire des systèmes d'objets, avec l'héritage et la classification, est au cœur de ses travaux.*

Annexe pour le service de fabrication

Article pour la revue :
Technique et science informatiques

Auteurs :
Amedeo Napoli — Jérôme Euzenat** — Roland Ducournau****

Titre de l'article :
Les représentations des connaissances par objets

Titre abrégé :
Les représentations par objets

Traduction du titre :
Object-based knowledge representation

Date de cette version :
11th February 2000

Coordonnées des auteurs :

– téléphone : 03 83 59 20 68
– télécopie : 03 83 27 83 19
– Email : napoli@loria.fr

Logiciel utilisé pour la préparation de cet article :
LaTeX, avec le fichier de style `article-hermes.cls`,
version 1.3 du 26/09/1999.

Formulaire de copyright :
Joindre le formulaire de copyright signé, récupéré sur le web à l'adresse
`http://www.hermes-science.com`