

# Édition sémantique d'arbres de décision pour l'oncologie avec KcatoS

Thomas Meilender, Nicolas Jay, Jean Lieber, Fabien Palomares

► **To cite this version:**

Thomas Meilender, Nicolas Jay, Jean Lieber, Fabien Palomares. Édition sémantique d'arbres de décision pour l'oncologie avec KcatoS. 1ère édition du Symposium sur l'Ingénierie de l'Information Médicale - SIIM 2011, Jun 2011, Toulouse, France. hal-00646843

**HAL Id: hal-00646843**

**<https://hal.inria.fr/hal-00646843>**

Submitted on 30 Nov 2011

**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.

# Édition sémantique d'arbres de décision pour l'oncologie avec KCATOS

Thomas Meilender\*,\*\* Nicolas Jay\*\*,  
Jean Lieber\*\*, Fabien Palomares\*

\*A2ZI - 61 ter rue de Saint-Mihiel - 55200 Commercy  
{prenom.nom}@a2zi.fr,  
<http://www.a2zi.fr>

\*\*UHP-Nancy 1 – LORIA (UMR 7503 CNRS-INPL-INRIA-Nancy 2-UHP)  
{prenom.nom}@loria.fr

**Résumé.** Au cours des deux dernières décennies, l'intérêt pour les guides de bonnes pratiques informatisés (GPBI) n'a cessé d'augmenter pour devenir une problématique majeure en informatique médicale. Les GPBI reposent généralement sur des recommandations de pratique clinique (RPC) créées par des experts médicaux. La transformation des RPC en GPBI se fait par une étape de formalisation, l'acquisition des connaissances. Cette tâche est généralement coûteuse en temps et en ressources, nécessitant des personnes qualifiées en médecine et en ingénierie des connaissances. Pour structurer les GPBI, plusieurs formalismes ont été proposés tels que Arden Syntax, GLIF, PROforma et Asbru. Cependant, aucun d'eux ne s'est imposé comme un standard et la plupart souffre d'un manque d'outils pour leur utilisation. Dans ce papier est présenté KCATOS, un éditeur sémantique d'arbres de décision. Utilisant un langage graphique simple, il permet l'export de RPC vers un langage de représentation des connaissances compréhensible par une machine, compatible avec le standard du Web sémantique OWL.

## 1 Introduction

Les recommandations de pratique clinique (RPC) sont définies par la Haute Autorité de Santé<sup>1</sup> «comme des propositions développées méthodiquement pour aider le praticien et le patient à rechercher les soins les plus appropriés dans des circonstances cliniques données». Construit à partir de consensus d'expert ou reposant sur les principes de la médecine fondée sur les faits, ils ont pour but d'améliorer la qualité des soins et de réduire les coûts. Les RPC sont le plus souvent disponibles sous forme de documents papiers, ce qui les rend difficiles à utiliser dans un contexte clinique. De plus, ils peuvent contenir des ambiguïtés ou des incohérences. Les guides de bonnes pratiques informatisés (GPBI) reposent généralement sur

---

1. Site de la Haute Autorité de Santé : [http://www.has-sante.fr/portail/jcms/c\\_418716/methodes-de-laboration-des-recommandations-de-bonne-pratique](http://www.has-sante.fr/portail/jcms/c_418716/methodes-de-laboration-des-recommandations-de-bonne-pratique). Dernière consultation : février 2011.

les RPC et permettent d'éviter plus facilement ces défauts. Cependant, leur développement est chronophage, coûteux et fastidieux (Latoszek-Berendsen et al., 2010). Il nécessite de disposer de personnes ayant acquis une expertise dans différents domaines, en particulier l'ingénierie des connaissances. De nombreux *frameworks* pour la création des GPBI ont été développés tels qu'Arden Syntax, GLIF, PROforma, Asbru et EON, mais aucun ne s'est imposé comme un standard universel (Clercq et al., 2004). Tous ces formalismes souffrent d'un manque d'implémentation logicielle et nécessitent pour la plupart des compétences spécifiques en gestion des connaissances et parfois même en algorithmique. D'après la littérature, «à ce moment, il n'existe pour les GPBI ni *framework* dominant, ni système répandu qui soit en utilisation clinique en dehors de l'institution dans laquelle il a été développé » (Sonnenberg et Hagerty, 2006).

Le projet de recherche KASIMIR<sup>2</sup> a démarré en 1997 et vise à créer des outils logiciels pour l'aide à la décision et plus généralement pour la gestion des connaissances décisionnelles en oncologie. KASIMIR est mené en partenariat avec Oncolor<sup>3</sup>, un réseau de soins rassemblant des professionnels de santé impliqués en cancérologie dans la région Lorraine. Les travaux récents du projet KASIMIR se concentrent principalement sur l'utilisation du Web sémantique (Berners-Lee et al., 2001) pour la formalisation, l'échange et l'exploitation des connaissances (D'Aquin et al., 2004). La dernière version du *toolkit* KATEXOWL, en particulier le *framework* EDHIBOU (Badra et al., 2008), utilise les technologies du Web sémantique telles que SPARQL et OWL. Son but est de générer des interfaces pratiques pour l'aide à la décision grâce à des formulaires simples qui guident les utilisateurs dans la base de connaissances.

Cependant, seuls quelques référentiels sont disponibles pour EDHIBOU. En effet, les RPC proposés par Oncolor doivent être formalisés, c'est-à-dire transformés en un langage exploitable par un moteur d'inférences. Jusqu'aujourd'hui, cette étape complexe nécessite deux types d'experts : des experts médicaux pour écrire les référentiels et des ingénieurs des connaissances pour les formaliser. Toutefois, il semble que si les experts médicaux pouvaient formaliser eux-mêmes les référentiels, le processus serait simplifié. Idéalement, un type de langage compréhensible par les experts du domaine et par une machine devrait être utilisé. Ce langage serait le chaînon manquant entre le Web sémantique et les experts médicaux.

Cet article présente une étape vers cet idéal : KCATOS est un éditeur sémantique d'arbres de décision qui permet de dessiner des arbres selon un langage simple et qui dispose d'un algorithme transformant ces arbres en connaissances décisionnelles formalisées, utilisant le standard du Web sémantique OWL.

La Section 2 décrit le langage des arbres de décision de KCATOS. La transformation de ce langage vers OWL est présentée dans la Section 3. L'éditeur de KCATOS, basé sur le langage de KCATOS, est présenté dans la Section 4 avec son interface utilisateur, ses facilités d'export et son utilisation avec EDHIBOU. La Section 5 mentionne les travaux antérieurs relatifs aux GPBI tandis que la Section 6 donne une conclusion et plusieurs travaux en cours et à venir.

---

2. Site du projet KASIMIR : <http://kasimir.loria.fr>. Dernière consultation : février 2011.

3. Site institutionnel d'Oncolor : <http://www.oncolor.fr>. Dernière consultation : février 2011.



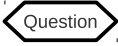
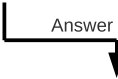
| Forme   | Commentaires  |
|---|---|
|  | Les rectangles aux angles arrondis représentent des situations médicales. Une situation médicale peut être définie comme l'état d'un patient décrit par un ensemble de variables telles que les résultats à des examens, sa physiologie, etc. |
|  | Les recommandations sont symbolisées par des rectangles classiques. Ils contiennent l'aide à la décision des RPC à destination du praticien.  |
|  | Les hexagones représentent les questions dont les réponses vont permettre de décrire la situation dans le but d'obtenir la recommandation adéquate.   |
|  | Les liens sont représentés par des arcs orientés. Lorsqu'ils relient un hexagone à un autre nœud, les liens sont typés, ce qui signifie qu'ils contiennent une réponse à la question les précédant directement.                               |

FIG. 1 – *Les formes et leur signification.*

## 2 Le langage d'arbre de décision de KCATOS

### 2.1 Présentation générale

Le langage d'arbre de décision de KCATOS est une représentation graphique basée sur un petit ensemble de formes géométriques connectées par des arcs orientés. De cette manière, les formes sont considérées comme les nœuds d'un arbre de décision. Un arbre commence par un nœud particulier appelé racine. En suivant un arc, le premier nœud est le parent et le second est appelé fils.

Du point de vue de la sémantique, chaque type de nœud a sa propre signification, comme présenté dans le tableau de la Figure 1.

Cette représentation est inspirée directement de la charte graphique de notre partenaire Oncolor. En effet, les RPC d'Oncolor utilisent des représentations graphiques, dont la plupart peuvent être vues comme des arbres de décisions. L'avantage d'utiliser ces formes géométriques est que les experts d'Oncolor les connaissent déjà. Nous cherchons donc à conserver la sémantique des graphiques d'Oncolor pour faciliter les futures créations et mises à jour de RPC.

### 2.2 Syntaxe et sémantique

#### 2.2.1 Syntaxe

Afin d'éviter les ambiguïtés et d'assurer la consistance des GPBI, les règles classiques des arbres de décision sont utilisées : un nœud a exactement un parent, à l'exception de la racine qui n'en a pas, et chaque arbre a une seule racine.

De plus, quelques règles ont été ajoutées pour garantir une sémantique correcte aux arbres :

- la racine est nécessairement une situation,
- une question a forcément au moins une réponse,
- chaque lien suivant une question doit être typé,
- un texte sur un lien n'est considéré comme réponse que si ce lien suit directement une question.

Un arbre respectant toutes ces règles est considéré comme syntaxiquement correct. L'exemple d'un tel arbre est présenté dans la Figure 2(a).

### 2.2.2 Sémantique

Un arbre de décision de KCATOS peut être exporté en OWL, comme le montre la section suivante. Sa sémantique est la même que celle de la base de connaissance exportée.

## 3 Export du langage de KCATOS vers OWL

### 3.1 Une brève introduction aux logiques de descriptions et à OWL

OWL est une recommandation du W3C pour représenter des ontologies<sup>4</sup>. C'est un formalisme inspiré des logiques de descriptions (LD (Baader et al., 2003)) dans un format XML-RDF. OWL est bien plus expressif que le langage de KCATOS mais, alors qu'il n'est pas raisonnable de supposer qu'une personne sans une expérience sérieuse en informatique et en logique puisse représenter des connaissances « proprement » en OWL, nous espérons que la représentation dans le langage graphique de KCATOS est suffisamment simple et intuitive pour que toute personne avec une bonne culture scientifique, comme le sont les médecins, puisse l'apprendre rapidement. Dans ce papier, la version 1.0 de OWL DL, qui correspond à la LD  $SHOIN(D)$ , est utilisée. Les LD forment une famille de logiques équivalentes à des fragments décidables de la logique du premier ordre (avec prédicat d'égalité et avec domaines concrets). Ci-dessous, seule une partie de  $SHOIN(D)$  suffisante pour comprendre cette section est présentée.

Les entités de représentation de  $SHOIN(D)$  sont les classes, les propriétés objets, les propriétés de type de données, les instances et les formules. Une interprétation est un couple  $\mathcal{I} = (\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$  où  $\Delta_{\mathcal{I}}$  est un ensemble non vide et où  $\cdot^{\mathcal{I}}$  est une fonction d'interprétation.

Une *instance*  $a$  représente un élément du domaine d'interprétation :  $a^{\mathcal{I}} \in \Delta_{\mathcal{I}}$ . Les instances de  $SHOIN(D)$  sont atomiques : ce sont des noms d'instances.

Une *propriété objet*  $p$  représente une relation binaire sur un domaine d'interprétation :  $p^{\mathcal{I}} \subseteq \Delta_{\mathcal{I}} \times \Delta_{\mathcal{I}}$ . Les propriétés de  $SHOIN(D)$  sont atomiques : ce sont des noms de propriétés.

Une *propriété de type de données*  $g$  représente une relation binaire sur  $\Delta_{\mathcal{I}} \times \Delta_D$  où  $\Delta_D$  est un type de donnée (p. ex., les booléens, les entiers, les flottants, les caractères ou les chaînes de caractères).  $\Delta_D$  est le co-domaine de  $g$ .

Une *classe*  $C$  représente un sous-ensemble du domaine d'interprétation :  $C^{\mathcal{I}} \subseteq \Delta_{\mathcal{I}}$ . Une classe est soit une *classe atomique* (i.e., un nom de classe), soit une expression de classe. Les expressions de classe introduites dans cette section sont d'une des formes suivantes :  $C \sqcap D$ ,  $\exists p.C$ ,  $\exists p.a$  et  $\exists g.v$  où  $C$  et  $D$  sont des classes (atomiques ou pas),  $p$  est une propriété objet,  $a$

<sup>4</sup>. Recommandation du W3c : <http://www.w3.org/TR/owl-features/>. Dernière consultation : février 2011.

est une instance,  $g$  est une propriété de type de données de co-domaine  $\Delta_D$  et  $v$  est une valeur du type  $\Delta_D$ . La sémantique de ces expressions de classe est la suivante :

$$\begin{aligned} (C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\ (\exists p.C)^{\mathcal{I}} &= \{x \in \Delta_{\mathcal{I}} \mid \exists y \in \Delta_{\mathcal{I}}, (x, y) \in p^{\mathcal{I}} \text{ et } y \in C^{\mathcal{I}}\} \\ (\exists p.a)^{\mathcal{I}} &= \{x \in \Delta_{\mathcal{I}} \mid (x, a^{\mathcal{I}}) \in p^{\mathcal{I}}\} \\ (\exists g.v)^{\mathcal{I}} &= \{x \in \Delta_{\mathcal{I}} \mid (x, v) \in g^{\mathcal{I}}\} \end{aligned}$$

Par exemple, la classe suivante représente l'ensemble des patients de 40 ans dont un parent au moins a un antécédent de cancer :

$$\text{Patient} \sqcap \exists \hat{\text{age}}.40 \sqcap \exists a\text{PourParent}.\exists a\text{PourAntécédent}.\text{Cancer}$$

(Patient et Cancer sont des classes atomiques, aPourParent et aPourAntécédent sont des propriétés objet et âge est une propriété de type de données de co-domaine le type des entiers naturels.)

Parmi les formules de  $\mathcal{SHOIN}(D)$ , les suivantes sont utilisées dans la section (avec  $C$  et  $D$ , deux classes,  $a$ , une instance,  $p$  une propriété objet et  $r$  une propriété objet ou une propriété de type de données) :

- $C \sqsubseteq D$  ( $C$  est plus spécifique que  $D$ ) :  $\mathcal{I}$  satisfait cette formule si  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ .
- $C \equiv D$  ( $C$  est équivalente à  $D$ ) :  $\mathcal{I}$  satisfait cette formule si  $C^{\mathcal{I}} = D^{\mathcal{I}}$ .
- $C(a)$  est satisfaite par  $\mathcal{I}$  si  $a^{\mathcal{I}} \in C^{\mathcal{I}}$ .
- «  $r$  est fonctionnelle » est satisfaite par  $\mathcal{I}$  si  $r^{\mathcal{I}}$  est une fonction partielle, c'est-à-dire que pour tout  $x \in \Delta_{\mathcal{I}}$  il existe au plus un  $y$  tel que  $(x, y) \in r^{\mathcal{I}}$ .
- «  $C$  est domaine de  $r$  » est satisfaite par  $\mathcal{I}$  si pour tout  $(x, y) \in r^{\mathcal{I}}$ ,  $x \in C^{\mathcal{I}}$ .
- «  $D$  est co-domaine de  $p$  » est satisfaite par  $\mathcal{I}$  si pour tout  $(x, y) \in p^{\mathcal{I}}$ ,  $y \in D^{\mathcal{I}}$ .

## 3.2 Algorithme d'export

### 3.2.1 Vue générale

L'algorithme d'export de KCATOS s'appuie sur deux classes : Situation et Recommandation. La première est relative au patient et la deuxième est relative aux décisions.

Ces deux classes sont liées de la façon suivante :

$$\text{Situation} \sqsubseteq \exists a\text{PourRecommandation}.\text{Recommandation}$$

Cela signifie que pour toute situation  $\sigma$  ( $\sigma \in \text{Situation}^{\mathcal{I}}$ ) il existe une recommandation  $\varrho$  ( $\varrho \in \text{Recommandation}^{\mathcal{I}}$ ) qui est associée à  $\sigma$  ( $(\sigma, \varrho) \in a\text{PourRecommandation}^{\mathcal{I}}$ ). Cette propriété aPourRecommandation lie une situation à une recommandation : elle a la classe Situation pour domaine et la classe Recommandation pour co-domaine.

Des sous-classes de Situation et Recommandation sont définies lors du processus d'export. Par exemple, considérons un patient qui a une céphalée et dont la recommandation est de prendre une aspirine. La classe PatientAvecCéphalée, sous-classe de Situation

## Édition sémantique d'arbres avec KCATOS

et la classe `PrescriptionDAspirine`, sous-classe de `Recommandation`, peuvent être introduites de la façon suivante :

$$\text{PatientAvecCéphalée} \equiv \text{Situation} \sqcap \exists \text{aPourSymptôme.SymptômeDeCéphalée}$$
$$\text{PrescriptionDAspirine} \sqsubseteq \text{Recommandation}$$

Puis, la formule suivante formalise l'assertion (médicalement discutable, mais c'est pour l'exemple) « Pour chaque patient avec une céphalée, une prescription d'aspirine est recommandée. » :

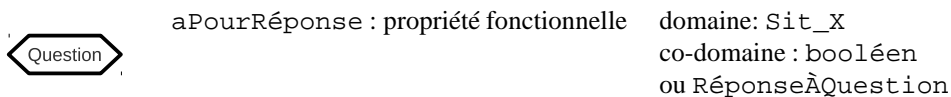
$$\text{PatientAvecCéphalée} \sqsubseteq \exists \text{aPourRecommandation.PrescriptionDAspirine}$$

### 3.2.2 Règles de traduction

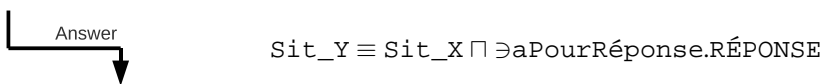
Un arbre de décision de KCATOS doit être lu selon un parcours en profondeur. Chaque nœud est transformé en utilisant des règles prenant en compte ses ancêtres. Ces règles sont expliquées ci-dessous :



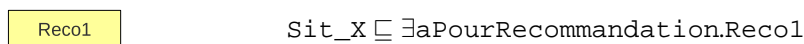
Une forme de situation permet de créer une classe `Sit_Y` comme sous-classe de `Sit_X`, où `Sit_X` est la plus proche sous-classe de `Situation` (*i.e.*, la sous-classe associée au nœud parent ou de l'arc parent).



Une forme de question introduit une nouvelle propriété fonctionnelle, `aPourRéponse`, ayant `Sit_X`, la plus proche sous-classe de situation, comme domaine. Si les réponses sont "vrai" et "faux" ou "oui" et "non", `aPourRéponse` est une propriété de type de données avec un co-domaine booléen. Sinon, c'est une propriété objet ayant une nouvelle sous-classe, `Réponse`, comme co-domaine.

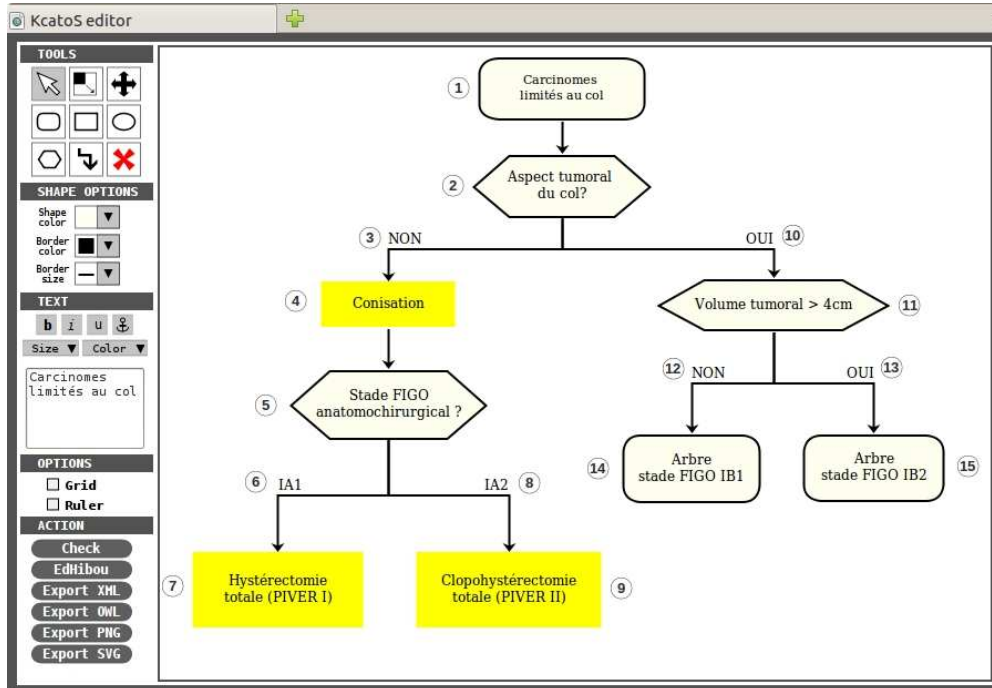


Comme cela a été écrit dans la section 2.2, un lien contient une réponse `RÉPONSE` à la question `aPourRéponse` qui le suit directement. Il introduit une nouvelle sous-classe de `Sit_X`, en spécifiant sa valeur de propriété.



Une forme de recommandation indique qu'une classe de situation `Sit_X` est liée à une recommandation `Reco1` par la propriété `aPourRecommandation`.

Un exemple d'export d'un arbre de décision vers OWL est montré à la figure 2.



(a) Un arbre de décision syntaxiquement correct édité par KCATOS.



(b) Représentation OWL de l'arbre de décision présenté ci-dessus.

FIG. 2 – Un extrait du référentiel Oncolor pour le cancer du col de l'utérus limité au col (a) et sa traduction en OWL (b).



## Édition sémantique d'arbres avec KCATOS

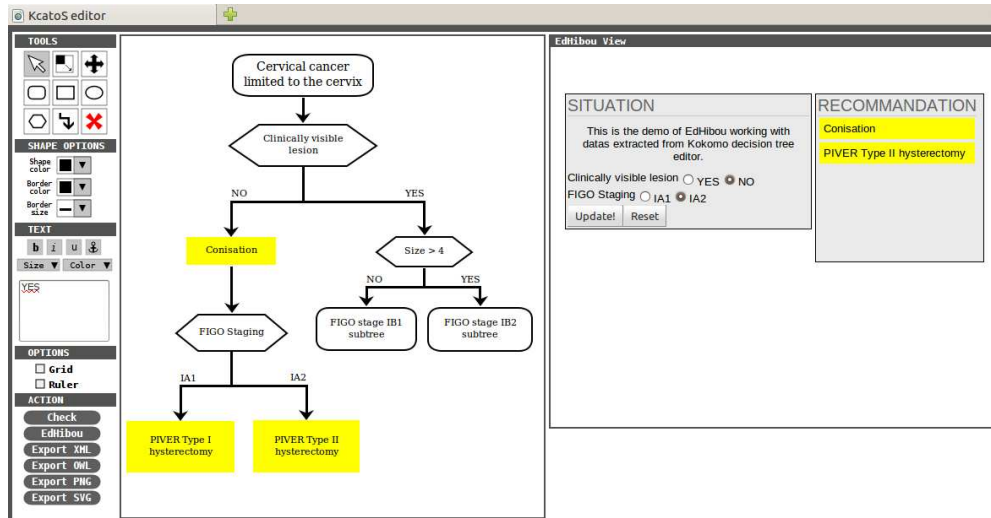


FIG. 3 – KCATOS combiné à EDHIBOU.

## 4 KCATOS en action

### 4.1 Présentation

KCATOS est une application Web reposant sur les langages PHP et Ajax, et utilisant les technologies SVG et Javascript pour la gestion des dessins. Ainsi KCATOS est un format ouvert à une utilisation collaborative et aux services Web. Son *framework* peut être intégré à la plupart des systèmes de gestion de contenu (SGC). Par exemple, des tests ont déjà été effectués avec succès pour le SGC Mediawiki<sup>5</sup>. Une copie d'écran de la version la plus simple fonctionnant de manière autonome est présentée dans la figure 2(a). Cette version sera prochainement disponible sous une licence libre. Actuellement, le fonctionnement de KCATOS a été validé pour les navigateurs Mozilla et Chrome.

Un module de validation syntaxique est proposé pour vérifier que les arbres dessinés respectent les règles définies dans la Section 2. En sortie, différents formats sont proposés : image matricielle (PNG), image vectorielle (SVG) et ontologie (OWL).

De plus, il est prévu dans un futur proche d'intégrer certains standards internationaux des GPBI, en particulier Arden syntax et GLIF dont la sémantique peut correspondre à celle de KCATOS.

### 4.2 KCATOS et EDHIBOU

EDHIBOU est un *framework* pouvant être implémenté en tant que service Web. Il fournit une interface utilisateur qui permet aux praticiens de décrire une situation liée à un patient pour obtenir la recommandation la plus adéquate présente dans le GPBI.

5. Site institutionnel de Mediawiki : <http://www.mediawiki.org>. Dernière consultation : février 2011.

Techniquement, EDHIBOU crée une instance de `Situation` dans l'ontologie et considère les propriétés liées comme des questions auxquelles l'utilisateur peut répondre dans l'interface. Combiné à KCATOS, il crée un formulaire correspondant à l'arbre de décision construit par l'utilisateur (cf. Figure 3). Ainsi, l'utilisateur peut vérifier que l'interprétation faite par KCATOS correspond à ce qu'il souhaite modéliser.

## 5 État de l'art

Plusieurs formalismes ont été élaborés pour mettre en œuvre les GPBI. Certains d'entre eux ont été comparés par Peleg et al. (2003) ou plus récemment par De Clercq et al. (2008). Arden syntax (Clayton et al., 1989) est probablement un des plus connus : il définit des règles textuelles indépendantes traduisant chacune une simple recommandation. Plus complets, GLIF (Ohno-Machado et al., 1998) et *PROforma* (Fox et al., 1998) font appels à des graphes et des langages objets, avec un nombre de types limité. Asbru (Shahar et al., 1998) propose une approche temporelle, centrée sur des objectifs, mais ne comporte pas de mécanisme d'inférence. On peut également citer les systèmes PRODIGY (Johnson et al., 2000) dans le domaine des pathologies chroniques ou encore GUIDE (Quaglioni et al., 2001) dont l'implémentation est basée sur des réseaux de Petri.

Malgré un certain nombre de points forts, ces formalismes utilisent des mécanismes, des formats, des outils qui leur sont propres et manquent d'interopérabilité. De plus, ils ne sont pas toujours suffisamment implémentés, et apparaissent aujourd'hui éloignés des standards du web sémantique.

D'autres systèmes utilisent explicitement une logique de description pour automatiser des référentiels de bonnes pratiques. C'est par exemple le cas de ASTI (Séroussi et al., 2009). Schulz et Hahn (2004) proposent également un système basé sur la logique *ALCN*.

Les travaux s'approchant le plus du système que nous présentons ici ont été publiés dans (Chen et al., 2006). Les auteurs s'appuient sur un module de Protégé permettant de tracer des arbres de décision. Ceux-ci sont directement implémentés en langage objet puis exportés au format XML et enfin exploités dans un logiciel spécifique. Cependant, ce formalisme n'est pas forcément utilisable par d'autres applications. De plus, les experts du domaine doivent avoir des compétences particulières en ingénierie des connaissances et savoir utiliser Protégé. Par ailleurs, ce système manque d'expressivité en ne considérant que des critères booléens.

## 6 Conclusion

De nombreuses études récentes ont montré l'importance des GPBI en ingénierie médicale. Alors que plusieurs formalismes ont été proposés, aucun ne s'est imposé comme un standard universel. Dans ce papier, l'éditeur sémantique d'arbre de décision KCATOS a été présenté. Basé sur les standards du Web sémantique et simple à utiliser, il propose un format ouvert, potentiellement compatible avec la plupart des formats existants pour les GPBI.

KCATOS fait partie d'un travail plus large sur l'édition collaborative des GPBI dans le cadre du projet KASIMIR. L'objectif global est de fournir des outils qui puissent assister les spécialistes médicaux à éditer, stocker et maintenir des connaissances décisionnelles dans le domaine de l'oncologie. De manière à permettre la réutilisation des connaissances dans les

systèmes de recommandation, la technologie des wikis sémantiques (Völkel et al., 2006) est utilisée. Pour parvenir à une intégration complète de KCATOS dans un *framework* de wiki sémantique, plusieurs développements sont nécessaires.

D'abord, l'édition des référentiels pourrait s'appuyer sur des systèmes de terminologie ou des ontologies existantes. Les futures améliorations de notre approche impliquent un alignement d'ontologies. Pour le moment, les créations de classes et de propriétés sont contextuelles. Par exemple, la propriété `avolumeSup4` de la Figure 2 est spécifique à ce référentiel. Toutefois, une telle propriété pourrait être généralisée et réutilisée dans d'autres arbres de décision.

Un autre problème est que, dans certains cas, les diagrammes représentés dans les RPC papier ne sont pas des arbres syntaxiquement correct au sens de KCATOS : par exemple, un nœud peut avoir plusieurs parents ou les transitions peuvent ne pas être données explicitement. Dans ce cas, KCATOS doit pouvoir prendre en compte des structures plus générales.

Enfin, nous prévoyons de travailler sur l'export des arbres de décision KCATOS vers différents formats tels GLIF ou Arden Syntax. Des solutions quant à l'alignement entre différentes logiques ayant des expressivités différentes doivent être trouvées.

## Références

- Baader, F., D. Calvanese, D. McGuinness, D. Nardi, et P. Patel-Schneider (Eds.) (2003). *The Description Logic Handbook*. UK : Cambridge University Press.
- Badra, F., M. d'Aquin, J. Lieber, et T. Meilender (2008). EdHibou : a customizable interface for decision support in a semantic portal. In *Proceedings of the Poster and Demonstration Session at the 7th International Semantic Web Conference (ISWC2008), Karlsruhe, Germany, October 28*.
- Berners-Lee, T., J. Hendler, et O. Lassila (2001). The semantic web. *Scientific American* 284(5), 34–43.
- Chen, K., C.-H. Chen, S.-Y. Chang, C. Chen, et Y.-C. Li (2006). A rapid development environment for synthesizing guideline-based decision support systems. In J. Ni et J. Dongarra (Eds.), *IMSCCS*, pp. 548–554. IEEE Computer Society.
- Clayton, P., T. Pryor, O. Wigertz, et G. Hripcsak (1989). Issues and structures for sharing medical knowledge among decision-making systems : The 1989 arden homestead retreat. In L. Kingsland (Ed.), *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pp. 116–21.
- Clercq, P. D., J. Blom, H. Korsten, et A. Hasman (2004). Approaches for creating computer-interpretable guidelines that facilitate decision support. *Artificial Intelligence in Medicine* 31(1), 1–27.
- D'Aquin, M., S. Brachais, J. Lieber, et A. Napoli (2004). Decision Support and Knowledge Management in Oncology using Hierarchical Classification. In Kaiser et al., pp. 16–30.
- De Clercq, P., K. Kaiser, et A. Hasman (2008). Computer-Interpretable Guideline formalisms. *Studies in health technology and informatics* 139, 22–43.
- Fox, J., N. Johns, et A. Rahmzadeh (1998). Disseminating medical knowledge : the PRO-forma approach. *Artificial Intelligence in Medicine* 14, 157–181.

- Johnson, P., S. Tu, N. Booth, B. Sugden, et I. Purves (2000). Using scenarios in chronic disease management guidelines for primary care. *Proceedings of the AMIA Symposium*, 389–393.
- Kaiser, K., S. Miksch, et S. Tu (Eds.). Volume 101 of *Studies in Health Technology and Informatics*, Prague, Czech Republic. Silvia Miksch and Samson W. Tu : IOS Press.
- Latoszek-Berendsen, A., H. Tange, H. J. van den Herik, et A. Hasman (2010). From clinical practice guidelines to computer-interpretable guidelines. A literature overview. *Methods of information in medicine* 49, 550–570.
- Ohno-Machado, L., J. Gennari, S. Murphy, N. Jain, S. Tu, D. Oliver, E. Pattison-Gordon, R. Greenes, E. Shortliffe, et G. Barnett (1998). The guideline interchange format : a model for representing guidelines. *Journal of the American Medical Informatics Association : JAMIA* 5, 357–372.
- Peleg, M., S. Tu, J. Bury, P. Ciccarese, J. Fox, R. Greenes, R. Hall, P. Johnson, N. Jones, A. Kumar, S. Miksch, S. Quaglini, A. Seyfang, E. Shortliffe, et M. Stefanelli (2003). Comparing computer-interpretable guideline models : A case-study approach. *JAMIA* 10.
- Quaglini, S., M. Stefanelli, G. Lanzola, V. Caporusso, et S. Panzarasa (2001). Flexible guideline-based patient careflow systems. *Artificial Intelligence in Medicine* 22, 65–80.
- Schulz, S. et U. Hahn (2004). A Description Logics approach to CPGs. In Kaiser et al., pp. 137–141.
- Séroussi, B., J. Bouaud, D. Denké, H. Falcoff, et J. Julien (2009). Using knowledge modelling to measure how clinical practice could actually be evidence-based : A preliminary analysis with arterial hypertension management. In K.-P. Adlassnig, B. Blobel, J. Mantas, et I. Masic (Eds.), *MIE*, Volume 150 of *Studies in Health Technology and Informatics*, pp. 668–672. IOS Press.
- Shahar, Y., S. Miksch, et P. Johnson (1998). The Asgaard project : a task-specific framework for the application and critiquing of time-oriented clinical guidelines. *Artificial Intelligence in Medicine* 14, 29–51.
- Sonnenberg, F. et C. Hagerty (2006). Computer-interpretable clinical practice guidelines. Where are we and where are we going? *IMIA Yearbook 2006 : Assessing Information - Technologies for Health*, 145–158.
- Völkel, M., M. Krötzsch, D. Vrandečić, H. Haller, et R. Studer (2006). Semantic wikipedia. In *WWW '06 : Proceedings of the 15th international conference on World Wide Web*, New York, NY, USA, pp. 585–594. ACM.

## Summary

During the two last decades, the interest for computer-interpretable guidelines (CIGs) has kept growing to become a major issue in medical informatics. CIGs generally rely on clinical practice guidelines (CPGs) created by medical experts. Transforming CPGs into CIGs is done through a formalisation step called knowledge acquisition. This is usually a time and resource consuming task requiring skilled personnel trained both in medicine and in knowledge engineering. To structure CIGs, some formalisms have been proposed such as the Arden Syntax, GLIF, *PROforma*, and *Asbru*. Meanwhile, none of them has emerged as a standard

## Édition sémantique d'arbres avec KCATOS

and many lack implementation tools. In this paper, KCATOS, a semantic decision tree editor, is presented. Using a simple graphical language, it allows exporting CPGs to machine-understandable knowledge, compatible with the semantic Web standard OWL.