



Des besoins des utilisateurs à la recherche de services web : Une approche sémantique guidée par les intentions

Isabelle Mirbel, Pierre Crescenzo

► To cite this version:

Isabelle Mirbel, Pierre Crescenzo. Des besoins des utilisateurs à la recherche de services web : Une approche sémantique guidée par les intentions. Revue des Sciences et Technologies de l'Information - Série ISI : Ingénierie des Systèmes d'Information, 2010, 15 (4), pp.89-112. 10.3166/isi.15.4.89-112 . hal-01137683

HAL Id: hal-01137683

<https://hal.science/hal-01137683>

Submitted on 31 Mar 2015

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.

Des besoins des utilisateurs à la recherche de services web : une approche sémantique guidée par les intentions

Isabelle Mirbel — Pierre Crescenzo

Université de Nice Sophia-Antipolis
Laboratoire I3S (CNRS / UNS)
930 route des Colles - BP 145
F-06903 Sophia-Antipolis cedex - France
{Isabelle.Mirbel, Pierre.Crescenzo}@unice.fr

RÉSUMÉ. Cet article présente le projet SATIS dont l'objectif principal est d'offrir la capacité à des utilisateurs finaux non informaticiens de décrire leurs démarches de recherche de services web pour opérationnaliser un processus métier. Dans cette approche, les démarches sont décrites sous forme d'intentions et de stratégies, desquelles sont dérivées des requêtes permettant de rechercher des services web. Nous nous appuyons sur les modèles et langages du web sémantique afin de proposer des moyens de raisonnement et d'explication des services web trouvés pour implémenter un besoin métier. Une démarche de recherche est vue comme un ensemble de fragments implémentés sous forme de règles exploitées en chaînage arrière, favorisant ainsi le partage, la réutilisation et la fertilisation croisée des savoir-faire. Dans cet article, l'approche SATIS est illustrée dans une communauté de neuroscientifiques.

ABSTRACT. This paper presents the SATIS project, which aims at providing to final users the capability to capitalise their know-how about how to search for web services to operationalise business processes. In SATIS, search procedures are elicited through intentions and strategies from which queries dedicated to search for web services are derived. We rely on semantic web models and languages in order to provide means to reason and explain why web services have been selected. A search procedure is seen as a set of building blocks implemented as rules exploited in backward chaining in order to support sharing, reuse and cross-fertilization of know-how. In this paper, the SATIS approach is illustrated in the context of a community of neuroscientists.

MOTS-CLÉS : services web, web sémantique, modélisation intentionnelle, règles, réutilisation
KEYWORDS: web services, semantic web, intentional modeling, rules, reuse

1. Introduction

Les services web sont de plus en plus utilisés lors du développement d'applications internet et l'une des difficultés de ce type de développement réside dans la recherche des services web adéquats. Cette difficulté va s'accroître avec l'augmentation du nombre et de la variété des services web disponibles en ligne. Dans ce contexte, le but de notre approche est d'aider les utilisateurs finaux d'une application développée à partir de services web à trouver des services web pertinents. Des approches existantes concernant le processus de recherche de services web considèrent que les buts des utilisateurs finaux ont déjà été identifiés, capturés, spécifiés et formalisés à l'aide d'un modèle dédié à la recherche de services. D'autres approches considèrent encore que les utilisateurs finaux, souvent spécialistes de leur domaine d'application, sont aussi des informaticiens ou des connaisseurs des services web. Nous considérons ces hypothèses trop fortes.

Aussi, l'ambition du projet SATIS¹ présenté ici est de permettre aux utilisateurs finaux d'exprimer leurs besoins et de les aider à trouver les services web disponibles qui correspondent à leurs besoins. Mais ce problème extrêmement complexe ne peut être résolu par une approche générale. Aussi, nous nous focalisons sur un domaine métier particulier dans lequel des services web sémantiques (décrits à l'aide d'annotations sémantiques) et une ou plusieurs ontologies sont disponibles. C'est le cas du domaine des neurosciences, qui illustre nos recherches et cet article.

Dans SATIS, nous proposons un environnement permettant de mettre à la disposition des membres d'une communauté d'intérêt des fragments réutilisables de démarche pour implémenter leurs buts métiers. Pour cela, notre approche s'appuie sur la spécification de haut niveau d'activités métiers à l'aide d'un modèle intentionnel. Nous nous intéressons également au partage de ces fragments de démarche à l'intérieur d'une communauté d'utilisateurs.

Notre travail s'inscrit dans la famille des approches de recherche de services web par les buts. La plupart de ces approches (Norton, 2007; Vukovic *et al.*, 2005; Zhang *et al.*, 2006; da Silva Santos *et al.*, 2008) portent sur la spécification de buts dans l'optique de recherche des services web satisfaisant ces buts. Dans ces approches, différents modèles ont été proposés pour spécifier des buts sans s'attacher au problème de leur capture. Au contraire, dans SATIS, notre objectif est de proposer des moyens d'assister les utilisateurs finaux dans l'explicitation de leurs buts. Par ailleurs, l'approche GODO (Gomez *et al.*, 2006) propose des modèles et des outils pour capturer les buts des utilisateurs avec l'aide d'une ontologie ou en langage naturel. Nous nous distinguons de cette approche par la proposition d'un processus incrémental de raffinement des besoins des utilisateurs dans le but de spécifier les caractéristiques des services web recherchés, comme c'est également le cas dans (Kaabi, 2007). Notre approche se distingue cependant de (Kaabi, 2007) par le fait que nous nous appuyons sur les modèles et langages du web sémantique pour enrichir la description des be-

1. SATIS : *Semantically AnnotAted Intentions for Services*.

soins des utilisateurs et ainsi proposer des moyens de raisonnement et d'explication des services web trouvés pour implémenter un besoin métier.

Par rapport aux approches dédiées à la recherche de services web basées sur des ontologies (Burnstein *et al.*, 2007), et plus précisément celles basées sur OWL-S (puisque nous nous appuyons sur OWL-S en ce qui concerne la description de services web), des algorithmes d'appariement (Li *et al.*, 2004) exploitant des descriptions de profils de services ont été proposés. D'autres algorithmes (Paolucci *et al.*, 2002) comparent des transformations d'état décrites dans une requête avec celles données dans les descriptions de services web. Tous ces algorithmes exploitent principalement la relation de subsumption. Des mécanismes de classement ont également été proposés (Benatallah *et al.*, 2003). Notre approche se distingue de ces travaux par le fait que nous nous concentrons sur l'élicitation des besoins des utilisateurs et leurs transformations en requêtes permettant la recherche de services web (plus que sur l'exécution des requêtes elles-mêmes). En d'autres termes, nous nous intéressons au processus amont qui consiste à dériver des requêtes des besoins des utilisateurs finaux. De plus, nous nous intéressons également à comment annoter ces requêtes de façon à permettre leur capitalisation et leur partage au sein d'une communauté d'utilisateurs.

Une autre nouveauté de notre approche est d'implémenter les démarches à l'aide de règles dans le but de favoriser la mutualisation des descriptions intentionnelles de haut niveau des besoins des utilisateurs et la fertilisation croisée en matière de connaissances sur la façon de rechercher des services web. Pour cela nous appliquons un algorithme de chaînage arrière sur l'ensemble ou une partie des données acquises.

Pour être plus précis, le résultat de la recherche finale dans le modèle SATIS est constitué par (i) un ou des services candidats qui répondent exactement et globalement aux objectifs de l'utilisateur final ou (ii) un ou des services candidats qui apportent une solution partielle (incomplète) ou proche (inexacte mais plus ou moins similaire) à ces objectifs. En d'autres termes, selon la précision de l'expression des objectifs de l'utilisateur final et en fonction de la disponibilité réelle de services web pertinents correspondant à ces objectifs, la *réponse* du système SATIS sera plus ou moins précise, structurée et complète.

L'article est organisé de la façon suivante. Tout d'abord nous donnons une vue d'ensemble de notre approche SATIS dans la section 2. Puis, dans la section 3, nous présentons des exemples puis expliquons plus précisément notamment le rôle de chaque utilisateur, le concept de directive et les règles permettant l'inférence. Enfin, nous concluons et donnons quelques perspectives de notre travail.

2. Vue d'ensemble de SATIS

Le but général de notre approche est de proposer à une communauté de neuroscientifiques, qui ne sont pas informaticiens, une solution complète pour facilement tirer parti d'un ensemble de services web. En fait, nous proposons d'aider ces utilis-

teurs à récupérer un ensemble organisé de descriptions de services web susceptibles d'implémenter le processus de traitement d'image qui les intéresse.

Comme nous nous intéressons aux besoins de haut niveau des utilisateurs finaux, nous nous appuyons sur une notation graphique simple et dédiée pour capturer et représenter ces besoins. Dans le contexte d'une communauté de neuroscientifiques, ces besoins portent plus précisément sur les processus de traitement d'images médicales. Différents formalismes ont été proposés pour représenter des processus métiers (Nurcan *et al.*, 2005). Les modèles orientés activités, comme BPMN (OMG, 2009) donnent une vue linéaire de la décomposition d'un processus en différentes activités. Dans ces modèles, un processus est décrit comme un ensemble prédéfini d'activités à réaliser et de relations entre ces activités pour exprimer un flot de données. Les modèles de processus orientés produits mettent en avant le résultat de l'exécution d'une activité. Ces approches permettent de suivre l'évolution du produit tout au long du processus. À chaque activité est associé un état du produit en cours d'évolution. Des approches, telles qu'EPC (Van der Aalst, 1998), permettant de représenter les dimensions temporelles et logiques ont également été proposées, mettant en avant le flot de contrôle dans le processus métier. Une famille récente rassemble les modèles de processus métier orientés décisions. Dans ces modèles, les transformations successives d'un produit sont considérées comme les conséquences de décisions prises dans un contexte particulier d'exécution. Les modèles orientés décisions permettent d'explicitier non seulement comment le processus se déroule mais également pourquoi. La mise en œuvre de tels processus guide la prise de décision inhérente au processus métier en question et aide à réfléchir sur la façon dont il est modélisé (Nurcan *et al.*, 2005).

Selon (Nurcan *et al.*, 2005), les processus métiers peuvent être considérés selon trois points de vue complémentaires : les objectifs du processus, l'organisation dans laquelle le processus existe et son infrastructure technologique. Les différentes familles de modèles de processus métiers se concentrent sur un ou plusieurs de ces points de vue.

Dans notre approche, l'infrastructure technologique est constituée d'un ensemble de services web sémantiquement annotés. Et nous nous intéressons ici aux objectifs des processus métiers afin de permettre le transfert de connaissances sur l'opérationnalisation d'un processus métier des utilisateurs finaux experts vers les utilisateurs finaux novices. Il est alors essentiel de modéliser le pourquoi de la décomposition du processus (c'est-à-dire la façon dont les intentions qui sous-tendent le processus à opérationnaliser sont décomposées en sous-intentions) aussi bien que le comment de la décomposition. De plus, de façon à permettre à des utilisateurs de profils et de niveaux d'expertise variés de tirer profit de ces démarches d'opérationnalisation de processus métier, il est essentiel de reposer sur un modèle de processus permettant de spécifier les processus métiers à plusieurs niveaux de détail. Le modèle de processus sur lequel nous nous appuyons doit permettre de montrer une même information à différents niveaux d'abstraction en fonction du profil de l'utilisateur et doit également proposer un haut niveau d'abstraction pour faciliter l'utilisation du système par des utilisateurs finaux novices. Pour toutes ces raisons, notre approche s'appuie sur le modèle de carte

(Rolland, 2007) que nous avons adapté à nos préoccupations. Ce formalisme de modélisation intentionnelle des processus permet aux utilisateurs finaux (dans notre cas aux neuroscientifiques) de définir leur processus de traitement d'image en décrivant des *intentions* intermédiaires (*i.e.* des buts et des sous-buts à satisfaire au travers de l'exécution du processus métier) et des *stratégies* (*i.e.* des moyens d'atteindre ces buts).

Nous ne souhaitons pas que ces utilisateurs spécifient explicitement les services web qui les intéressent (ce qui les forcerait à connaître et comprendre les spécifications de service web disponibles) mais plutôt les intentions qu'ils souhaitent satisfaire en utilisant des services web. De plus, nous ne souhaitons pas associer de façon explicite des spécifications de services web aux *besoins intentionnels de haut niveau des utilisateurs finaux*. Dans notre approche, les utilisateurs finaux définissent plutôt des requêtes qui sont associées à leurs besoins. En effet, les requêtes constituent un moyen pour eux de définir des *spécifications génériques de services web*. Par exemple, dans une communauté de neuroscientifiques, en cherchant un service web qui prend en entrée une image et fournit en sortie une image débiaisée, l'utilisateur final spécifie le genre de service web qu'il recherche sans explicitement faire référence à un service en particulier. Ainsi, le couplage entre les besoins de haut niveau des utilisateurs finaux d'une part et les annotations de services web d'autre part est relativement faible. En effet, si une nouvelle annotation de service web est ajoutée à la base d'annotations de services web, elle pourra être proposée pour opérationnaliser un besoin de haut niveau même si celui-ci a été spécifié avant l'ajout de l'annotation en question ; et si, au contraire, une annotation de service web est supprimée de la base d'annotation, les besoins que ce service web satisfaisait restent toujours valides et pourront être opérationnalisés par d'autres services web dont l'annotation est toujours disponible dans la base d'annotation. En effet, les services web sont sélectionnés de façon dynamique au moment où la recherche de services web est effectivement mise en œuvre (et non au moment de la spécification de la démarche d'opérationnalisation d'un besoin), c'est-à-dire au moment où les requêtes associées aux besoins intentionnels de haut niveau des utilisateurs finaux sont exécutées.

Dans notre approche nous avons également adopté les modèles et langages du web sémantique comme un cadre unifié pour représenter (i) les besoins intentionnels de haut niveau des utilisateurs finaux, (ii) les spécifications génériques de services web et (iii) les spécifications des services web. En ce qui concerne les *besoins intentionnels de haut niveau des utilisateurs finaux*, nous avons adapté le modèle de carte (Rolland, 2007) en rassemblant ses éléments dans une ontologie dédiée à la représentation des processus intentionnels (Corby *et al.*, 2009). Cette ontologie est spécifiée en RDFS (W3C, 2004). Les processus intentionnels sont annotés à l'aide des concepts et des relations de cette ontologie permettant ainsi le partage et le raisonnement sur ces représentations intentionnelles.

Pour la spécification de services web, de nombreuses approches ont porté sur la description sémantique de services. Parmi elles, on peut citer les plus connues : OWL-S (OWL-S Coalition, 2005), WSMO (WSMO WorkingGroup, 2004) et SAWSDL (SAWSDL, 2007). OWL-S et WSMO sont des ontologies permettant de décrire les

différents aspects des services web. SAWSDL est un mécanisme permettant d'insérer dans la description d'un service des annotations sémantiques, indépendamment de tout langage et de toute ontologie. Notre approche n'est pas dédiée à un type d'annotation en particulier. Nous souhaitons à terme permettre de trouver des services web indépendamment de l'ontologie ou de la technique utilisée pour les annoter. Ainsi, un processus donné pourrait par exemple être opérationnalisé à la fois par des services web annotés en WSMO et en OWL-S. Dans cet article, nous illustrons notre approche avec des services web annotés à l'aide de l'ontologie OWL-S.

Enfin, les *spécifications génériques de services web* sont données à l'aide du langage de requête pour RDF SPARQL (W3C, 2008). Elles sont modélisées par des patrons de graphes qui sont projetés sur les graphes des annotations de services web. L'utilisation de SPARQL pour représenter les spécifications génériques de services web explique également notre choix de OWL-S et WSMO (plutôt que de SAWSDL) pour annoter les services web.

Dans SATIS, comme nous le détaillons dans la suite de cet article, les descriptions génériques de services web et les besoins intentionnels de haut niveau des utilisateurs finaux sont rassemblés dans des fragments autonomes et réutilisables permettant la réutilisation de connaissance sur l'opérationnalisation d'un processus de traitement d'image d'une démarche à une autre. Ces fragments sont implémentés par des règles. Nous nous appuyons sur le moteur de chaînage arrière de CORESE (Edelweiss Team of INRIA Sophia-Antipolis, 2009) pour raisonner sur ces règles et ainsi trouver des descriptions de services web pertinentes. Nous reviendrons plus en détail dans la section 3 sur ce que nous entendons par fragment de démarche.

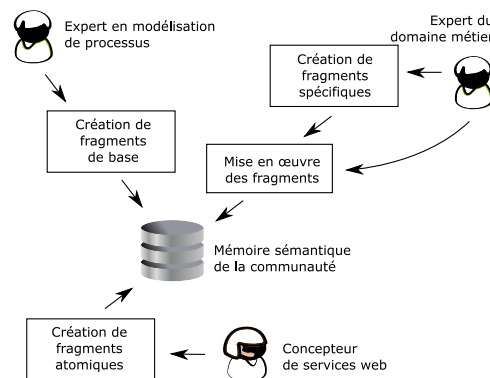


Figure 1. L'approche SATIS

Dans SATIS, nous distinguons trois acteurs principaux (figure 1) : le *concepteur de services web*, l'*expert en modélisation de processus* et l'*expert du domaine métier*.

Le concepteur de services web a pour mission d'écrire des descriptions génériques de services web et de leur associer des spécifications de besoins intentionnels de haut niveau dans le but de promouvoir les services web dont il est en charge, et cela du point

de vue de l'utilisateur final. Il contribue ainsi à la création de fragments de démarche atomiques et réutilisables.

L'expert en modélisation de processus a pour mission d'enrichir la mémoire sémantique de la communauté avec des fragments de base de démarches dédiés à l'implémentation de besoins intentionnels de haut niveau. Il propose des fragments réutilisables dans différentes chaînes de traitement d'image. La correction d'intensité, traitement commun à différentes chaînes de traitement d'image, est un exemple de fragment de base. Pour réaliser efficacement son travail au sein du système, l'expert en modélisation ne peut évidemment pas être étranger au métier visé. Il doit à la fois maîtriser celui-ci et la modélisation : il est le lien rare mais essentiel entre le concepteur de services web et l'expert du domaine métier.

Enfin, l'expert du domaine métier (utilisateur final des services web) cherche des descriptions de services pour opérationnaliser sa chaîne de traitement d'image. Il peut chercher parmi les besoins intentionnels de haut niveau déjà explicités dans la mémoire sémantique de la communauté ou décider de créer par lui-même une nouvelle spécification de besoin. Cette étape de création consiste à spécifier des besoins intentionnels de haut niveau à l'aide de buts et de stratégies et de les raffiner en buts et stratégies intermédiaires jusqu'à spécifier des buts suffisamment précis pour être associés à des spécifications génériques de services web. Ensuite, l'étape de mise en œuvre de la démarche de recherche de services web ainsi spécifiée consiste à opérationnaliser la chaîne de traitement d'image dont les besoins ont été élicités durant l'étape de création à l'aide de services web dont les annotations sont disponibles dans la mémoire collective de la communauté au moment de la recherche.

Dans une communauté de neuroscientifiques, les informaticiens peuvent jouer les rôles de *concepteurs de services web* et d'*experts en modélisation de processus* et les neuroscientifiques jouent les rôles d'*experts en modélisation de processus* et d'*experts du domaine métier*.

Durant l'*étape de création*, le modèle de carte (Rolland, 2007) aide à capturer les besoins intentionnels de haut niveau des utilisateurs finaux. L'ontologie pour les processus intentionnels, l'ontologie du domaine métier et l'ontologie OWL-S sont utilisées pour annoter les besoins intentionnels de haut niveau et pour spécifier les descriptions génériques de services web qui leur sont associées. Les annotations RDF qui représentent les besoins intentionnels de haut niveau et les requêtes en SPARQL qui représentent les descriptions génériques de services web sont ensuite rassemblées dans des règles qui sont considérées comme des fragments de démarches de recherche de services web réutilisables.

L'*étape de mise en œuvre* est supportée par un moteur de chaînage arrière qui exploite les règles et les annotations de services web. Pour cela nous nous appuyons sur le moteur sémantique CORESE² (Corby *et al.*, 2008; Corby *et al.*, 2006) à la fois pour le mécanisme de chaînage arrière sur la base de règles de SATIS, règles qui im-

2. <http://www-sop.inria.fr/edelweiss/software/corese/>

plémentent les fragments de démarche réutilisables (*i.e.* la mémoire sémantique de la communauté) et pour l'appariement des requêtes spécifiées dans les règles avec la base de connaissance des annotations de services web (base d'annotations des services de la communauté). Durant l'étape de mise en œuvre, des besoins intentionnels de haut niveau sont créés dynamiquement quand cela est nécessaire durant le processus de chaînage arrière, comme des sous-buts temporaires, jusqu'à ce que des descriptions de services web correspondant à chacun des sous-buts soient trouvées et ainsi le but principal (besoin intentionnel de plus haut niveau) satisfait. Ainsi, un membre de la communauté qui cherche des services web pour opérationnaliser une chaîne de traitement d'images particulière va tirer profit de toutes les règles et de toutes les annotations de services web présentes dans la mémoire sémantique de la communauté au moment de sa recherche. Cette mémoire évolue au cours du temps et les annotations de services web proposées comme résultat de la recherche peuvent donc également varier.

Nous attirons l'attention du lecteur sur le fait que le résultat de la mise en œuvre d'un ensemble de fragments de démarche est un ensemble de descriptions de services web candidats à l'opérationnalisation d'une chaîne de traitement et non un ensemble de services web. L'invocation des services web sélectionnés (parmi les services candidats) dépasse le cadre de notre travail. Quand le processus de recherche de descriptions de services web est mis en œuvre, un ensemble de services web candidats (alternatifs) est associé à chaque but ou sous-but élicité durant l'étape de création des fragments de démarche. Le résultat de la mise en œuvre du processus de recherche est donc une séquence d'ensemble de services web candidats. Et comme le formalisme que nous avons choisi pour modéliser les chaînes de traitement d'image, le modèle de carte (Rolland, 2007), permet de spécifier différentes façons d'atteindre une intention, le résultat de la mise en œuvre du processus de recherche peut être constitué de plusieurs séquences d'ensembles de services web candidats. En fonction du nombre et de l'adéquation des services web disponibles, les services candidats représenteront une solution plus ou moins complète et pertinente aux besoins exprimés. Mais, même si la solution est parcellaire ou approchée, elle peut servir à compléter et améliorer les services web disponibles.

En nous appuyant sur une représentation à l'aide de règles des besoins des utilisateurs finaux et des descriptions génériques des services web et en proposant d'utiliser des modèles distincts et dédiés aux experts du domaine d'une part et aux concepteurs de services web d'autre part ainsi que des mécanismes de correspondance entre les deux, notre objectif est d'améliorer la collaboration bilatérale entre les neuroscientifiques et les informaticiens au sein de la communauté.

Dans la section 3 de cet article, nous décrivons SATIS plus en détail. En premier lieu, nous présentons une architecture et un modèle simplifiés des fragments de démarche réutilisables. Nous expliquons ensuite plus précisément ces fragments. Puis nous détaillons comment ils sont exploités par les concepteurs de services web, les experts en modélisation de processus et les experts du domaine métier.

3. SATIS vu de l'intérieur

3.1. Architecture de SATIS

La figure 2 présente l'architecture de SATIS autour de la notion de *fragments de démarche réutilisables*. Nous y exprimons la liaison entre la base de connaissances rassemblant les fragments de démarche réutilisables et les trois principales ontologies du système SATIS : celle permettant d'annoter les processus intentionnels (de recherche de services web), celle du domaine métier (ici, les neurosciences) et celle décrivant les services web eux-mêmes (ici, OWL-S). Cette architecture est complétée dans la section suivante par un modèle présenté sous forme de diagramme de classes selon un formalisme UML simplifié.

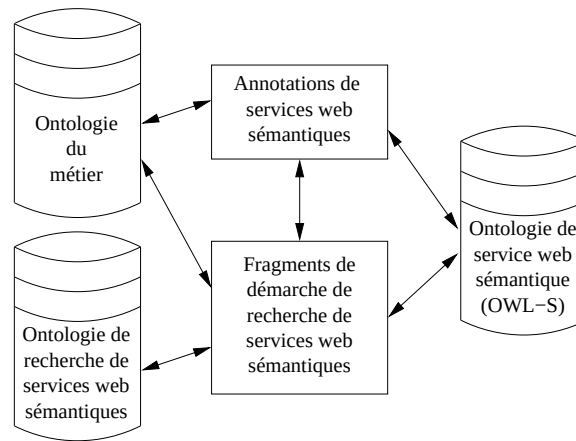


Figure 2. Architecture des fragments de démarche de SATIS

3.2. Modèle de fragments de démarche de SATIS

La figure 3 décrit les entités (classes) et relations entre entités du cœur de SATIS, autour des fragments de démarche. Nous pouvons y voir les entités classiques du modèle de carte (pour l'expression et la capture des buts) : *Carte*, *Section*, *Stratégie* et *Intention*. Les entités plus spécifiques à SATIS sont dévolues à la capitalisation de démarches (*fragments* et *directive intentionnelle*) et à la recherche de services web adéquats (*directive opérationnelle* et *requête*).

Dans SATIS, le processus de recherche d'annotations sémantiques de services web à partir de besoins intentionnels est vu comme un ensemble de fragments de processus (ou fragments de démarche) faiblement couplés et exprimés à différents niveaux de granularité. Cette vue modulaire a pour but de faciliter les adaptations et les extensions des démarches de recherche ainsi modélisées. De plus, cette façon de considérer

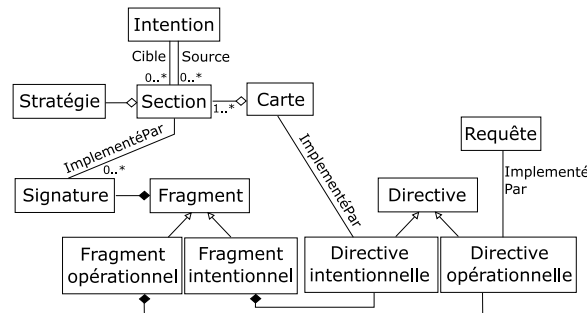


Figure 3. Modèle des fragments de démarche de SATIS

le processus de recherche permet de réutiliser des fragments construits dans l'optique d'opérationnaliser une chaîne de traitement d'images particulière lors de l'opérationnalisation d'une autre chaîne de traitement. Le corps d'un fragment de démarche est constitué de directives qui peuvent être considérées comme autonomes et réutilisables. La signature du fragment de démarche indique de son côté la situation dans laquelle il est pertinent de réutiliser le fragment.

Une directive est définie comme un fait, une indication ou une procédure pour déterminer la façon de réaliser une action. Pour nous, une directive capture un savoir-faire sur le moyen d'atteindre une intention dans une situation donnée. Nous distinguons deux types de directives : les *directives intentionnelles* et les *directives opérationnelles*. Les directives intentionnelles servent à spécifier les besoins intentionnels de haut niveau des utilisateurs finaux, besoins nécessitant d'être raffinés en des besoins plus précis. Les directives opérationnelles représentent des descriptions génériques de services web.

La figure 4 montre un exemple de besoin intentionnel de haut niveau. Cet exemple porte sur une chaîne de traitement d'images dédiée à la classification de tissus humains et de lésions. Elle a été spécifiée à l'aide du modèle de carte (Rolland, 2007). Une carte est un modèle de processus dans lequel une présentation non déterministe des *intentions* et des *stratégies* est capturée. C'est un graphe étiqueté et orienté avec des intentions comme nœuds et des stratégies comme arcs entre les intentions. Une intention de traitement d'images est un but qui doit être atteint en suivant une stratégie. Une intention exprime ce qui est voulu, un état ou un résultat attendu indépendamment d'où, quand et par qui (ou quoi) il est obtenu. Il existe deux intentions particulières indiquant le début et la fin d'un processus. Une stratégie caractérise le flux de l'intention source vers l'intention cible et la façon dont l'intention cible est atteinte. Il peut y avoir plusieurs stratégies entre deux intentions.

La figure 4 présente cinq intentions principales qui sont : conversion d'image, prétraitement d'image, préparation d'image, découpage du crâne et segmentation d'image. Dans une carte, chaque ensemble constitué d'une inten-

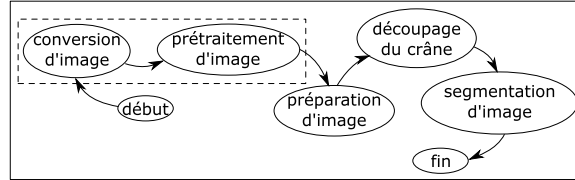


Figure 4. Exemple de besoin intentionnel de haut niveau des utilisateurs finaux

tion source, d'une intention cible et d'une stratégie est une *section*. Un exemple de section a été entouré par une ligne pointillée dans la figure 4. Précisons qu'une carte n'est pas du tout un diagramme d'état, car il n'y a ni structure de données, ni objet, ni valeur affectée. Une carte n'est pas non plus un diagramme d'activité car il y a toujours un contexte qui est pris en considération dans chaque section : l'intention source et la stratégie. Le modèle de carte permet de capturer d'autres aspects des besoins formulés par des utilisateurs finaux, mais ce n'est pas l'objet de cet article de le décrire complètement.

Dans l'exemple de la figure 4, des informations supplémentaires sont nécessaires pour comprendre quel genre de description générique serait adéquat pour rechercher des services web permettant d'implémenter la préparation et le prétraitement des images. Pour cela, chaque section de la carte peut être raffinée en une autre carte décrivant plus en détails la façon d'atteindre l'intention cible de la section considérée. La figure 5 montre un exemple de carte qui décrit de façon plus précise la section mise en évidence dans la figure 4. Dans cette carte, différentes façons d'atteindre l'intention cible de prétraitement des images sont proposées. Une première façon de faire consiste d'abord à réaliser l'intention *correction de biais* puis à réaliser directement le *débruitage d'image*. Une autre façon de faire consiste à réaliser l'intention *rotation d'image* avant d'atteindre le *débruitage d'image*.

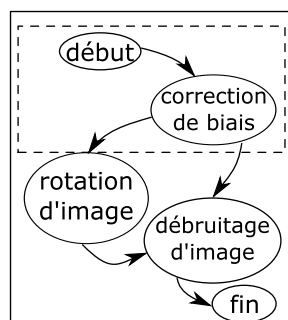


Figure 5. Exemple de raffinement de besoin

Au niveau de raffinement présenté dans la figure 5, il est maintenant possible d'associer des descriptions génériques de services web aux sections de la carte dans le but

d'expliciter comment trouver des services web susceptibles de les implémenter. Par exemple, une requête portant sur une annotation sémantique de service web indiquant que le service attend en entrée une image et produit en sortie une image débiaisée et un biais permettrait de rechercher des services web pour opérationnaliser la section.

Les cartes et les requêtes, qui modélisent respectivement des directives intentionnelles et des opérationnelles, sont conservées dans des fragments de démarche réutilisables par les membres de la communauté. Les directives constituent le corps des fragments de démarche et une signature leur est associée afin de permettre leur réutilisation. Une signature de fragment de démarche est représentée par une section de carte. L'intention cible de la section indique le but du fragment et l'intention source ainsi que la stratégie caractérisent la situation de réutilisation dans laquelle le fragment est pertinent. Nous distinguons les *fragments intentionnels*, dont le corps contient des directives intentionnelles (*i.e.* des cartes) des *fragments opérationnels*, dont le corps contient des directives opérationnelles (*i.e.* des requêtes). La section mise en évidence dans la figure 4 est un exemple de signature pour un fragment intentionnel dont le contenu est la carte présentée dans la figure 5. La section mise en évidence dans la figure 5 est un exemple de signature pour un fragment opérationnel dont le contenu est une requête recherchant des annotations sémantiques de services web qui ont en entrée une image et en sortie une image débiaisée et un biais.

Les fragments de démarche sont implémentés sous forme de règles dont la conclusion représente la signature du fragment et dont la prémisse représente le corps du fragment. Cette formalisation a été choisie car elle permet ensuite de raisonner sur les différents fragments de démarche présents dans la mémoire sémantique de la communauté, favorisant ainsi leur réutilisation.

Dans les sections qui suivent, nous allons tout d'abord détailler comment nous tirons parti des langages et modèles du web sémantique pour formaliser des directives intentionnelles (section 3.2.1) et opérationnelles (section 3.2.2) ainsi que pour représenter les signatures des fragments (section 3.2.3). Ensuite, dans la section 3.3, nous détaillerons l'implémentation de ces règles.

3.2.1. Directives intentionnelles

Une directive intentionnelle est une directive complexe qui s'appuie sur une structure de graphe pour relier entre elles ses sous-directives. Chaque sous-directive appartient à un des deux types : intentionnel ou opérationnel.

De façon à formaliser les intentions et les stratégies, nous nous appuyons sur la proposition de Prat (Prat, 1997; Prat, 1999) qui s'est déjà avérée utile pour formaliser des buts dans (Ralyté, 2001; Guzelian, 2007; Rolland, 2007). D'après ces références, une déclaration d'intention se caractérise par un verbe et plusieurs paramètres qui jouent des rôles spécifiques vis-à-vis du verbe. Parmi ces paramètres, nous pouvons citer, par exemple, l'objet sur lequel porte l'action décrite par le verbe.

Si nous considérons à nouveau le contenu de la carte de la figure 5, l'intention correction de biais est décrite par son verbe débiaiser et son objet image.

Grâce à l'ontologie pour les processus intentionnels que nous avons proposée (Corby *et al.*, 2009), ces derniers peuvent être représentés à l'aide d'annotations RDF.

Les classes principales de notre ontologie des processus intentionnels sont *Section*, *Intention*, *Ressource* (qui représentent les services web résultat du processus de recherche), *Verbe* et *Objet* (qui permettent de décrire une intention) et *Paramètre* (utilisé pour décrire une stratégie permettant d'atteindre une intention cible d'une section).

Actuellement nous utilisons différents verbes décrivant différents traitements d'images, tous instances de la classe *Verbe* ; et nous considérons également différents concepts du domaine métier comme des instances de la classe *Objet*. La classe *Paramètre* est instanciée en différentes classes permettant de modéliser le contexte du traitement d'images considéré.

Les correspondances entre les concepts de l'ontologie du domaine métier et ceux de l'ontologie des processus intentionnels (instanciations) sont créées automatiquement lorsque les concepts du domaine métier sont sélectionnés pour expliciter une intention ou une stratégie durant l'étape de création d'une démarche de recherche de services web.

3.2.2. Directives opérationnelles

Dans cet article, nous considérons que les annotations sémantiques des services web sont exprimées en OWL-S. Actuellement, nous n'utilisons que les parties *profile* et *grouding* de la description OWL-S ainsi que la description des entrées et des sorties dans la partie *process*. Nous enrichissons la description OWL-S d'un service en considérant les paramètres en entrée et en sortie des services web comme des instances des concepts de l'ontologie du domaine métier. Par exemple, dans la partie *process* de la description OWL-S qui suit, dans laquelle l'espace de nommage *process* fait référence à l'ontologie des processus OWL-S et l'espace de nommage *dom* fait référence à l'ontologie du domaine métier, le paramètre *Input_Img* est considéré comme une instance du concept *Image*, le paramètre *Output_Img* est considéré comme une instance du concept *DebiasedImage* et le paramètre *Output_BiasField* est considéré comme une instance du concept *BiasField*.

```
<process:AtomicProcess
  rdf:ID="BiasCorrection">
  <service:describes
    rdf:resource="#BiasCorrectionService"/>
  <process:hasInput>
    <process:Input rdf:ID="Input\_Img">
      <process:parameterType>&xsd:string
    </process:parameterType>
  </process:Input>
  </process:hasInput>

  <process:hasOutput>
```

```

    <process:Output rdf:ID="Output\_Img">
      <process:parameterType>&xsd;#string
    </process:parameterType>
    <rdf:type
      rdf:resource="&dom;#DebiasedImage"/>
    </process:Output>
  </process:hasOutput>
  <process:hasOutput>
    <process:Output
      rdf:ID="Output\_BiasField">
        <process:parameterType>&xsd;#string
      </process:parameterType>
      <rdf:type
        rdf:resource="&dom;#BiasField"/>
      </process:Output>
    </process:hasOutput>
  </process:AtomicProcess>

```

Les descriptions génériques de services web au cœur de l'approche SATIS exploitent ces instanciations pour retrouver des services web susceptibles d'opérationnaliser une chaîne de traitement d'images. Selon le moteur de recherche sémantique utilisé, une requête (*i.e.* une description générique de service web) peut porter sur des paramètres exactement de même nature (*i.e.* des instances de concepts comme BiasField ou DebiasedImage) mais aussi sur des paramètres instances des sous-classes des concepts considérés.

Une directive opérationnelle capture une spécification générique de service web. Les directives opérationnelles sont exprimées sous forme de requêtes SPARQL sur les annotations OWL-S de services web. La requête suivante est un exemple de description générique permettant de rechercher des services web pour débiaiser des images.

```

prefix dom: <http://.../dom-onto#>
prefix process: <http://.../Process.owl#>
select ?service
where
{
  ?service process:hasInput ?r1
  filter(?r1 =: dom:Image)
  ?service process:hasOutput ?r2
  filter (?r2 <=: dom:DebiasedImage)
  ?service process:hasOutput ?r3
  filter (?r3 <=: dom:BiasField)
}

```

Dans cet exemple, nous recherchons les services web dont la description OWL-S indique que le service web considéré attend comme paramètre d'entrée une instance de la classe Image et comme paramètres de sortie une instance de la

classe `DebiasedImage` (ou d'une de ses sous-classes) et une instance de la classe `BiasField` (ou d'une de ses sous-classes).

3.2.3. Signature de fragment de démarche

La signature d'un fragment de démarche sert à capturer le but du fragment et la situation de réutilisation dans laquelle la directive proposée dans le corps du fragment est pertinente. Cette situation de réutilisation est explicitée par une section de carte dans laquelle (i) l'intention source et la stratégie décrivent la situation dans laquelle le fragment peut être réutilisé et (ii) l'intention cible représente le but que le fragment de démarche permet d'atteindre.

Dans une signature de fragment, seule l'intention cible est obligatoire car elle décrit le but du fragment de démarche. Un fragment de démarche peut également nécessiter un contexte particulier pour être applicable. En d'autres termes, une intention source et/ou une stratégie peuvent être requises pour atteindre une intention cible. La spécification de l'intention source et/ou de la stratégie impacte l'opérationnalisation proposée dans le corps du fragment. Les annotations de services web sélectionnées comme résultat de l'exécution d'un fragment de démarche dont la signature n'inclut pas d'intention source correspondent à des services moins spécifiques que ceux dont les annotations sont sélectionnées comme résultat de l'exécution d'un fragment de démarche dont la signature inclut une intention source (restreignant ainsi le champ des services web considérés). De la même façon, le fait de préciser ou non une stratégie dans la signature d'un fragment permet respectivement de réduire ou d'élargir le spectre des services web considérés. Ces deux moyens (spécification d'une intention source et spécification d'une stratégie) peuvent être combinés pour obtenir une signature dont le niveau de généralité correspond effectivement aux directives proposées dans le corps du fragment.

La signature du fragment dont le corps est l'exemple de requête décrite précédemment (section 3.2.2) est constituée d'une intention cible dont le verbe est *débiaiser* et l'objet *image*. Aucune intention source et aucune stratégie ne sont proposées, ce qui signifie que les directives proposées dans le corps du fragment de démarche sont pertinentes quelle que soit la situation de réutilisation.

3.3. Des fragments de démarche aux règles

Dans SATIS, nous proposons de conserver les fragments de démarche dans la mémoire sémantique de la communauté sous forme de règles exploitées en chaînage arrière dont la conclusion représente la signature du fragment et la prémisse représente le corps du fragment. Nous parlons de règle *concrète* ou *abstraite* suivant que la prémisse représente des directives opérationnelles ou intentionnelles. Le langage SPARQL constitue un cadre unifié pour représenter ces règles concrètes et abstraites en reposant sur la forme de requête `CONSTRUCT WHERE`. La clause `CONSTRUCT` de la requête est interprétée comme la tête de la règle (*i.e.* ce qui doit être prouvé). La

clause WHERE de la requête est interprétée comme le corps de la règle, c'est-à-dire la condition qui permet de prouver sa tête. De façon récursive, un ensemble de requêtes SPARQL de la forme CONSTRUCT WHERE peut-être vu comme un ensemble de règles traitées en chaînage arrière.

3.3.1. Règles concrètes

Une règle concrète assure la réalisation d'une section de carte (la condition à prouver) en recherchant des annotations OWL-S de services web correspondant au patron de graphe spécifié dans la clause WHERE de la requête SPARQL. La règle suivante, dans laquelle l'espace de nommage map fait référence à l'ontologie des processus intentionnels, est un exemple de règle concrète qui implémente un fragment de démarche opérationnel visant à rechercher des services web pour implémenter un débiaisage d'image.

```
<rule rdf:ID="rule-c2">
<rule:value>
prefix dom: <http://.../dom-onto#>
prefix map: <http://.../map-onto#>
prefix process: <http://.../Process.owl#>
construct
{
    \_:s map:hasStrategy \_:g
    \_:g map:hasParameter map:AnyParameter
    \_:s map:hasSource \_:o
    \_:o map:hasObject map:AnyObject
    \_:o map:hasVerb map:AnyVerb
    \_:s map:hasTarget \_:i
    \_:i map:hasObject dom:Image
    \_:i map:hasVerb dom:Debiasing
    \_:s map:hasResource ?service
}
where
{
    ?service process:hasInput ?r1
    filter(?r1 =: dom:Image)
    ?service process:hasOutput ?r2
    filter (?r2 <=: dom:DebiasedImage)
    ?service process:hasOutput ?r3
    filter (?r3 <=: dom:BiasField)
}
pragma {cos:server cos:query true}
</rule:value>
</rule>
```

La requête spécifiée dans la clause WHERE de la règle correspond à une description générique de service web dont il a été question précédemment. Dans cet exemple,

nous recherchons les services web dont la description OWL-S indique que le service web considéré attend comme paramètre d'entrée une instance de la classe `Image` et comme paramètres de sortie une instance de la classe `DebiasedImage` (ou d'une de ses sous-classes) et une instance de la classe `BiasField` (ou d'une de ses sous-classes). Dans la clause `CONSTRUCT` de la règle, le patron de graphe spécifié correspond à la carte à construire si des annotations de services web sont trouvées comme résultat de la requête décrite dans la clause `WHERE`. Dans ce cas, une section de carte ayant pour intention cible une intention qualifiée par l'objet `image` et le verbe `debiasing` est considérée comme vraie. La section de carte en question a une intention source qualifiée par n'importe quel objet (`AnyObject`) et n'importe quel verbe (`AnyVerb`) et une stratégie qualifiée par n'importe quel paramètre (`AnyParameter`), ce qui signifie qu'il n'y a pas de précondition particulière associée à la réalisation de la section en question. Les annotations de services web (`?service`) sont associées à la section de carte à l'aide de la propriété `hasResource`.

3.3.2. Règles abstraites

Les règles abstraites assurent la réalisation des sections de carte en prouvant l'opérationnalisation de leurs sections. Par exemple, les règles abstraites qui assurent la réalisation de la section de carte mise en évidence dans la figure 4 le font en prouvant l'opérationnalisation de chacune des sections qui sont présentées dans la figure 5, c'est-à-dire en prouvant, pour chacune des sections, au moins une règle ayant la section considérée comme conclusion, et ainsi de suite de façon récursive. De façon plus précise, deux formes de règles abstraites sont nécessaires pour modéliser un fragment de démarche intentionnel. La première forme de règle abstraite explicite comment construire une directive complexe à partir de directives simples, c'est-à-dire comment construire une séquence à partir de sections à prouver (*i.e.* un chemin dans la carte). L'autre forme de règle abstraite explicite comment construire une section de carte qui qualifie le fragment intentionnel considéré (*i.e.* la signature du fragment intentionnel) à partir d'un chemin à prouver. Si la directive complexe considérée dans le corps du fragment intentionnel (*i.e.* la carte) contient plusieurs chemins, il faut produire un couple de règles abstraites (une de la première forme et une de la seconde) pour chacun de ces chemins.

Nous représentons la première forme de règle abstraite comme une requête SPARQL dont la clause `CONSTRUCT` contient un patron de graphe qui correspond à la séquence des sections à prouver pour réaliser l'intention spécifiée dans la signature du fragment intentionnel considéré (*i.e.* un chemin dans la carte correspondant à la directive complexe considérée). Et la clause `WHERE` de la requête SPARQL contient les patrons de graphe de chaque section à prouver pour pouvoir construire la séquence de la conclusion de la règle.

La deuxième forme de règle abstraite est représentée par une requête SPARQL dont la clause `WHERE` contient le patron de graphe de la séquence de sections à prouver (*i.e.* le chemin dans la carte) pour réaliser la section de carte spécifiée dans la signature du fragment de démarche, c'est-à-dire dans la clause `CONSTRUCT` de la règle abstraite.

La figure 6 montre un exemple de couple de règles abstraites. Ces règles représentent un fragment de démarche intentionnel correspondant à l'opérationnalisation du chemin qui n'inclut pas la réalisation d'une rotation dans la carte présentée dans la figure 5.

<pre> <!-- abstract rule --> <rule rdf:ID="rule-a2b"> <rule:value> prefix dom: <http://.../dom-onto#> prefix map: <http://.../map-onto#> construct { _:s1 map:hasSource map:Start _:s1 map:hasStrategy _:g1 _:g1 map:hasParameter map:AnyParameter _:s1 map:hasTarget _:i1 _:i1 map:hasObject dom:Image _:i1 map:hasVerb dom:Debiasing _:s1 map:hasResource ?servicex _:s2 map:hasSource _:i1 _:s2 map:hasStrategy _:g2 _:g2 map:hasParameter map:AnyParameter _:s2 map:hasTarget _:i2 _:i2 map:hasObject dom:Image _:i2 map:hasVerb dom:Denoising _:s2 map:hasResource ?servicey _:s3 map:hasSource _:i2 _:s3 map:hasStrategy _:g3 _:g3 map:hasParameter map:AnyParameter _:s3 map:hasTarget map:Stop } where { ?sx map:hasTarget ?ix ?ix map:hasObject dom:Image ?ix map:hasVerb dom:Debiasing ?sx map:hasResource ?servicex ?sy map:hasTarget ?iy ?iy map:hasObject dom:Image ?iy map:hasVerb dom:Denoising ?sy map:hasResource ?servicey } </rule:value> </rule> </pre>	<pre> <!-- abstract rule --> <rule rdf:ID="rule-a2a"> <rule:value> prefix dom: <http://.../dom-onto#> prefix map: <http://.../map-onto#> construct { _:s map:hasStrategy _:g _:g map:hasParameter map:AnyParameter _:s map:hasSource _:o _:o map:hasObject map:AnyObject _:o map:hasVerb map:AnyVerb _:s map:hasTarget _:i _:i map:hasObject dom:Image _:i map:hasVerb dom:Preprocessing _:s map:hasResource ?service1 _:s map:hasResource ?service2 } where { ?s1 map:hasSource map:Start ?s1 map:hasTarget ?i1 ?i1 map:hasObject dom:Image ?i1 map:hasVerb dom:Debiasing ?s1 map:hasResource ?service1 ?s2 map:hasSource ?i1 ?s2 map:hasTarget ?i2 ?i2 map:hasObject dom:Image ?i2 map:hasVerb dom:Denoising ?s2 map:hasResource ?service2 ?s3 map:hasSource ?i2 ?s3 map:hasTarget map:Stop } </rule:value> </rule> </pre>
--	--

Figure 6. Exemple de couple de règles abstraites

La clause WHERE de la règle abstraite de gauche contient les patrons de graphe correspondant aux sections permettant de réaliser les intentions débiaiser image et débruiter image. Le patron de graphe correspondant au chemin constitué de ces deux sections est spécifié dans la clause CONSTRUCT de la règle dans laquelle une première section d'intention source start et une dernière section d'intention source stop ont été introduites. On remarque également que l'intention cible de la première section (qualifiée par l'objet Image et le verbe Debiasing) apparaît comme intention source de la deuxième section, constituant ainsi une carte. La propriété hasResource permet de lier les annotations de services web résultats au chemin en construction. Le patron de graphe correspondant à ce chemin apparaît également dans la clause WHERE de la règle abstraite de droite. Ici, la stratégie n'a pas été reprise dans la description du patron, ce qui signifie que n'importe quelle carte incluant une intention qualifiée par Image et Debiasing suivie d'une intention qualifiée par Image et

Denoising convient. On aurait aussi bien pu inclure les paramètres décrivant la stratégie dans cet exemple. La règle de droite contient dans sa clause **CONSTRUCT** le patron de graphe correspondant à la signature du fragment intentionnel considéré : ce fragment permet de réaliser l'intention cible caractérisée par l'objet *Image* et par le verbe *Preprocessing*. L'intention source est qualifiée par l'objet *AnyObject* et le verbe *AnyVerb* ce qui signifie qu'aucun contexte d'exécution particulier n'est associé à ce fragment. La propriété *hasResource* permet d'associer les annotations de services web résultats (*?service1* provenant de la première section et *?service2* provenant de la seconde) à la section de carte considérée.

À un instant donné, dans la mémoire sémantique de la communauté, plusieurs opérationnalisations d'une section de carte (*i.e.* une signature de fragment) exploitant différentes directives opérationnelles et éventuellement intentionnelles et traduisant différentes façons de rechercher des descriptions de services web à différents niveaux d'abstraction peuvent coexister. Tous ces fragments de démarche partagent une même signature (conclusions de règle identiques) mais diffèrent par leurs directives (prémisses de règle différentes).

4. Mise en œuvre des fragments de démarche

4.1. Point de vue du concepteur de service web

Le concepteur de service web est en charge de promouvoir les services web disponibles au sein de la communauté de neuroscientifiques. Pour cela, quand il souhaite faire connaître un nouveau service dans la communauté, en plus d'ajouter une annotation sémantique du service dans la base d'annotations, il propose un fragment de démarche atomique dans la mémoire sémantique de la communauté. Il écrit alors une règle de la forme de celle présentée dans la section 3.3.1. Dans la clause **CONSTRUCT** de la règle, l'intention cible est obligatoire alors que l'intention source et la stratégie ne sont spécifiées que si le service web considéré n'est réutilisable que dans un contexte particulier. L'intention source permet de spécifier des préconditions sur le service web dont il est question. La stratégie permet d'indiquer la façon dont l'intention cible est atteinte : à l'aide d'un algorithme particulier ou en satisfaisant des critères de qualité de service par exemple.

4.2. Point de vue de l'expert en modélisation de processus

L'expert en modélisation de processus a pour mission de peupler la mémoire sémantique de la communauté de fragments de démarches afin d'aider les experts du domaine métier (i) à expliciter les chaînes de traitement d'images pour lesquelles ils sont à la recherche de service web et (ii) à trouver des services web leur permettant d'opérationnaliser les chaînes de traitement d'images qui les intéressent. Pour cela, l'expert en modélisation peut partir des fragments de démarche atomiques fournis par le concepteur de service web au sein de la mémoire sémantique de la communauté dans le but de

les agréger en des chaînes basiques de traitement d'images. Par exemple si, à un moment donné, des fragments de démarche atomiques qualifiés par `normaliser image` et `enregistrer image` (verbes des intentions cibles) sont proposés par le concepteur de service web dans la mémoire sémantique de la communauté, l'expert en modélisation de processus peut alors décider de les agréger en un fragment de démarche dont l'intention cible est qualifiée par le verbe `préparer image` et ainsi proposer une façon d'opérationnaliser une chaîne basique de traitement d'images. L'expert en modélisation de processus peut également observer dans le travail de création des experts du domaine un besoin récurrent et décider de le spécifier sous forme de fragments de démarche adéquats. Par exemple, il peut proposer l'ensemble des règles abstraites présentées à la figure 6 afin de mettre à la disposition des experts du domaine métier un ensemble de fragments de base pour opérationnaliser l'étape de prétraitement de leurs chaînes de traitement d'images (*i.e.* les sections ayant une intention cible qualifiée par le verbe `prétraiter image`). Dans ce cas, si des règles concrètes pour rechercher des descriptions de services web correspondant aux sous-intentions `correction de biais`, `rotation d'image` et `débruitage d'image` sont disponibles dans la mémoire sémantique de la communauté, alors l'expert en modélisation de processus peut n'écrire que les règles abstraites de la figure 6. Sinon, il écrit également les règles concrètes permettant d'opérationnaliser les sous-intentions. Cette description de l'expert en modélisation de processus démontre qu'il est important qu'il connaisse bien le métier ciblé : il est souvent impossible de travailler sur la modélisation d'un processus sans bien comprendre l'usage de celui-ci. Aussi, généralement, cet expert est aussi un expert du domaine (métier) dont l'expérience en matière de modélisation est devenue importante. C'est un profil rare mais il n'est pas nécessaire que les experts en modélisation soient nombreux.

4.3. *Le point de vue de l'expert du domaine*

Enfin, l'expert du domaine recherche des services web pour opérationnaliser une chaîne de traitement d'images. Pour cela, il peut commencer par regarder dans la mémoire sémantique de la communauté s'il existe déjà des règles permettant l'opérationnalisation d'intentions qui caractérisent la chaîne de traitement d'images qui l'intéresse. Si un autre membre de la communauté a déjà explicité une chaîne de traitement d'images ayant le même but, l'expert du domaine peut alors réutiliser ce qui a été conservé dans la mémoire sémantique de la communauté. Le but de la chaîne de traitement d'images considérée peut également correspondre à un sous-but d'une autre chaîne de traitement d'images plus conséquente et déjà explicitée sous forme de fragments de démarche conservés dans la mémoire sémantique de la communauté. Dans ce cas aussi, l'expert du domaine peut décider de réutiliser telles quelles les règles présentes dans la mémoire sémantique de la communauté. Si aucune des règles abstraites de la mémoire sémantique de la communauté ne traite de la chaîne de traitement d'images que l'expert du domaine souhaite opérationnaliser, alors, avec l'aide de l'expert en modélisation de processus, il spécifie la chaîne de traitement d'images. Pour cela, ils écrivent les règles abstraites correspondant à cette carte de plus haut niveau.

Ensuite, pour chaque section identifiée dans la carte, les experts peuvent rechercher dans la mémoire de la communauté si des règles opérationnalisant les sections de la carte principale ont été proposées. Si c'est le cas, ils peuvent décider de s'appuyer sur ces règles et terminer l'étape de création. Ils peuvent également décider de proposer une façon différente d'opérationnaliser ces sections. Ainsi, ils enrichissent la mémoire sémantique de la communauté avec une nouvelle façon d'opérationnaliser une intention déjà présente dans la mémoire. Cela a pour conséquence d'enrichir également les façons d'opérationnaliser les chaînes de traitement d'images dont les buts et les sous-buts ont déjà été explicités dans la mémoire sémantique de la communauté. Et, quand un autre expert met en œuvre une démarche de recherche de services web pour la chaîne de traitement d'images qui l'intéresse, si cette démarche inclut des intentions cibles pour lesquelles de nouveaux fragments ont été ajoutés dans la mémoire sémantique de la communauté, alors le moteur de chaînage arrière exploite les règles précédemment conservées dans la mémoire de la communauté ainsi que les nouvelles, augmentant ainsi la diversité des façons de trouver des services web pertinents pour opérationnaliser un sous-but d'une chaîne de traitement d'images.

À chaque fois que l'expert du domaine métier, avec l'aide de l'expert en modélisation de processus, explicite une nouvelle façon d'opérationnaliser une section de carte, il doit choisir le bon niveau de spécification de la signature du fragment afin que les autres membres de la communauté puissent éventuellement réutiliser son savoir-faire en dehors de la chaîne de traitement d'images pour laquelle il a été créé.

Dans une communauté de neuroscientifiques, un expert du domaine (*i.e.* un neuroscientifique) qui cherche des services web pour opérationnaliser une chaîne de traitement d'images dédiée à la classification de tissus et de lésions, par exemple, commence par expliciter les besoins intentionnels de haut niveau comme cela est illustré à la figure 4. Ensuite, il cherche dans la mémoire sémantique de la communauté s'il existe déjà des règles abstraites ou concrètes qui permettent d'opérationnaliser les sections de cette carte de haut niveau, à savoir les sections ayant pour verbes des intentions cibles : *conversion d'image*, *prétraitement d'image*, *préparation d'image*, *découpage du crâne* et *segmentation d'image*. Il trouve et décide de s'appuyer par exemple sur les règles abstraites proposées par l'expert en modélisation de processus pour opérationnaliser *prétraitement d'image*. Il explicite également le fait que *conversion d'image* est une tâche manuelle pour laquelle il n'est pas nécessaire de rechercher des descriptions de services web. L'expert du domaine, avec l'aide de l'expert en modélisation de processus, écrit ensuite des règles concrètes pour réaliser les intentions cibles *découpage du crâne* et *segmentation d'image* ainsi que *préparation d'image*. En proposant une règle concrète pour opérationnaliser la réalisation de l'intention cible *préparation d'image*, l'expert enrichit la mémoire sémantique de la communauté d'un nouveau savoir-faire pour opérationnaliser la *préparation d'image* (déjà explicitée par l'expert en modélisation de processus). Lorsqu'il décide de mettre en œuvre la démarche ainsi explicitée dans le but de rechercher des descriptions de services web susceptibles d'opérationnaliser la chaîne de traitement d'images qui l'intéresse, toutes les règles de la mémoire sémantique de la communauté sont exploitées par le moteur de chaînage arrière, à sa-

voir, en autres, la nouvelle règle concrète pour réaliser l'intention cible préparation d'image ainsi que les règles déjà présentes dans la mémoire sémantique de la communauté. Il peut ainsi découvrir d'autres façons de rechercher des services web.

5. Conclusion

Dans cet article, nous avons présenté SATIS, une approche pour rechercher des services web à partir des besoins des utilisateurs finaux dans le but d'opérationnaliser des processus métiers dans le contexte d'un domaine d'application donné. Notre approche repose sur une représentation modulaire du processus de recherche à l'aide de fragments de démarche réutilisables et combinables permettant de conserver et de partager le savoir-faire en matière de recherche de services web au sein de la mémoire sémantique d'une communauté de neuroscientifiques.

Notre approche s'appuie sur une représentation intentionnelle des processus d'une part et sur une description sémantique des services web d'autre part. Elle exploite les capacités de raisonnement des modèles et langages du web sémantique pour dériver les caractéristiques des services web recherchés des besoins des utilisateurs. Plus précisément, les fragments de démarche sont modélisés sous forme de règles implémentées par des requêtes SPARQL de la forme `CONSTRUCT WHERE`. Le mécanisme de recherche de service web est supporté par un moteur de chaînage arrière sur la base de règles et sur les annotations OWL-S des services web.

CORESE (Edelweiss Team of INRIA Sophia-Antipolis, 2009; Corby *et al.*, 2008) est un moteur de recherche sémantique développé par l'équipe EDELWEISS de l'INRIA. Il intègre un moteur de chaînage arrière exploitant des règles implémentées par des requêtes SPARQL de la forme `CONSTRUCT WHERE`. À court terme, nos travaux futurs vont porter sur l'adaptation de notre modèle de représentation des démarches de recherche de services web aux contraintes de ce moteur afin de pouvoir l'utiliser pour illustrer notre travail, notamment sur des exemples plus concrets et plus complexes dans le domaine des neurosciences. Nous prévoyons également de développer des outils pour automatiser les étapes fastidieuses de notre approche (comme la génération des correspondances entre les concepts du domaine et les concepts de l'ontologie des processus intentionnels par exemple ou la transformation des cartes annotées en requêtes SPARQL).

À moyen terme, nous comptons également proposer, à l'aide du métamodèle présenté dans (Hug, 2009), un modèle hybride couplant les concepts du modèle de carte à ceux d'autres modèles de processus afin d'étendre l'expressivité de notre modèle pour prendre en considération d'autres aspects des processus métiers tels que les ressources auxquelles les différentes activités sont affectées.

Nous envisageons également d'enrichir notre approche en intégrant le point de vue organisationnel sur les processus métiers afin notamment de mieux tenir compte du contexte dans lequel les objectifs des processus métiers sont spécifiés et de ceux dans lesquels ils sont réutilisés.

En ce qui concerne la spécification des services web, dans cet article, nous avons illustré notre travail sur des services annotés à l'aide de OWL-S. Nous comptons aussi étudier l'ontologie WSMO afin que notre approche reste relativement indépendante du modèle d'annotation utilisé. Ainsi, nous prévoyons de proposer la recherche de services dans une base d'annotations de services exprimées aussi bien en OWL-S qu'en WSMO. Nous comptons également étudier plus en détail le langage de requête proposé pour rechercher des services annotés avec SAWSDL afin d'étendre notre approche à la capitalisation de requêtes SPARQL et de requêtes SAWSDL dans les fragments opérationnels.

6. Bibliographie

- Benatallah B., Hacid M., Rey C., Toumani F., « Request rewriting-based Web service discovery », *Proceedings of the international Semantic Web conference (ISWC 2003)*, Springer, Sanibel Island, FL, p. 242-257, October, 2003.
- Burnstein D. M. M., McDermott D., McIlraith S., Paolucci M., Sycara K., McGuinness D., Sirin E., Srinivasan N., « Bringing Semantics to Web Services with OWL-S », *World Wide Web (2007)*, p. 243-277, 2007.
- Corby O., Dieng-Kuntz R., Faron-Zucker C., « Querying the semantic web with the CORESE search engine », *16th European Conference on Artificial Intelligence (ECAI/PAIS)*, Valencia, Spain, p. 705-709, 2008.
- Corby O., Dieng-Kuntz R., Faron-Zucker C., Gandon F., « Searching the Semantic Web : Approximate Query Processing based on Ontologies », *IEEE Intelligent Systems Journal*, 2006.
- Corby O., Faron-Zucker C., Mirbel I., « Implementation of intention-driven search processes by SPARQL queries », *11th International Conference on Enterprise Information Systems (ICEIS)*, Milan, Italy, May, 2009.
- da Silva Santos L. B., Pires L. F., van Sinderen M., « A Goal-Based Framework for Dynamic Service Discovery and Composition », *International Workshop on Architectures, Concepts and Technologies for Service Oriented Computing*, Porto, Portugal, p. 67-78, July, 2008.
- Dieng R., Corby O., Gandon F., Giboin A., Golebiowska J., Matta N., Ribière M., *Knowledge management : Méthodes et outils pour la gestion des connaissances*, Dunod, 2005.
- Edelweiss Team of INRIA Sophia-Antipolis, « CORESE : COncceptual REsource Search Engine », , <http://www-sop.inria.fr/edelweiss/software/corese/>, 2009.
- Gomez J., Rico M., Garcia-Sanchez F., « GODO : Goal Oriented Discovery for Semantic Web Services », *5th International Semantic Web Conference*, 2006.
- Guzelian G., *Modélisation et spécification de composants réutilisables pour la conception de systèmes d'information*, Thèse de doctorat, Université Aix Marseille, 2007.
- Hug C., *Méthode, modèles et outil pour la méta-modélisation des processus d'ingénierie de systèmes d'information*, Thèse de doctorat, Université Joseph Fourier, 2009.
- Kaabi R., *Une approche méthodologique pour la modélisation intentionnelle de services et leur opérationnalisation*, Thèse de doctorat, Université Paris I Sorbonne, 2007.
- Li L., Horrocks I., « A Software Framework for Matchmaking Based on Semantic Web Technology », *International Journal of Electronic Commerce*, vol. 8, n° 4, p. 39-60, 2004.

- Norton M. S. B., « A Refined Goal Model for Semantic Web Services », *Second International Conference on Internet and Web Applications and Services (ICIW'07)*, 2007.
- Nurcan S., Edme M., « Intention-driven modeling for flexible workflow applications », *Journal of Software Process : Improvement and Practice*, 2005.
- OMG, « Business Process Model and Notation », , [http ://www.omg.org/spec/BPMN/2.0/](http://www.omg.org/spec/BPMN/2.0/), 2009.
- OWL-S Coalition, « OWL-S Specification », , [http ://www.ai.sri.com/daml/services/owl-s/1.2/](http://www.ai.sri.com/daml/services/owl-s/1.2/), 2005.
- Paolucci M., Kawamura T., Payne T., Sycara K., « Semantic Matching of Web Services Capabilities », *First Int. Semantic Web Conference*, Sardinia, Italy, June, 2002.
- Prat N., « Goal formalisation and classification for requirements engineering », *Third International Workshop on Requirements Engineering : Foundations of Software Quality, (REFSQ)*, Barcelona, Spain, 1997.
- Prat N., Réutilisation de la trace par apprentissage dans un environnement pour l'ingénierie des processus, Thèse de doctorat, Université Paris I - Sorbonne, 1999.
- Ralyté J., Ingénierie des méthodes à base de composants, Thèse de doctorat, Université Paris I - Sorbonne, 2001.
- Rolland C., *Conceptual Modelling in Information Systems Engineering*, Springer-Verlag, 2007.
- SAWSDL, « Semantic Annotations for WSDL and XML Schema », , [http ://www.w3.org/TR/sawSDL/](http://www.w3.org/TR/sawSDL/), 2007.
- Van der Aalst W., Formalization and Verification of Event-driven Process Chains, Computing Science Reports n° 98/01, Eindhoven University of Technology, Eindhoven, 1998.
- Vukovic M., Robinson P., « GoalMorph : Partial Goal Satisfaction for Flexible Service Composition », *International Conference on Next Generation Web Services Practices*, 2005.
- W3C, « RDF Vocabulary Description Language 1.0 : RDF Schema », , [http ://www.w3.org/TR/rdf-schema/](http://www.w3.org/TR/rdf-schema/), 2004.
- W3C, « SPARQL Query Language for RDF », , W3C Recommendation, [http ://www.w3.org/TR/rdf-sparql-query/](http://www.w3.org/TR/rdf-sparql-query/), 2008.
- W3C, « RDF : Resource Description Framework », , [http ://www.w3.org/RDF/](http://www.w3.org/RDF/), 2009.
- WSMO WorkingGroup, « Web Service Modeling Ontology », , [http ://www.wsmo.org/](http://www.wsmo.org/), 2004.
- Zhang K., Li Q., Sui Q., « A Goal-driven Approach of Service Composition for Pervasive Computing », *1st International Symposium on Pervasive Computing and Applications*, p. 593-598, August, 2006.