Un assistant de mémoire pour les très petits projets d'ingénierie du logiciel

Vincent Ribaud, Philippe Saliou, Claude Laporte

▶ **To cite this version:**

Vincent Ribaud, Philippe Saliou, Claude Laporte. Un assistant de mémoire pour les très petits projets d'ingénierie du logiciel. Études de communication - Langages, information, médiations, 2011, 36, pp.67-86. hal-00630358

HAL Id: hal-00630358
https://hal.univ-brest.fr/hal-00630358

Submitted on 9 Oct 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.

VINCENT RIBAUD, PHILIPPE SALIOU
ET CLAUDE Y. LAPORTE

UN ASSISTANT DE MÉMOIRE POUR LES TRÈS PETITS PROJETS D'INGÉNIERIE DU LOGICIEL

Résumé

Nous proposons d'assister la mémoire des très petits projets d'ingénierie du logiciel avec une infrastructure la plus simple possible (un wiki sémantique) et des activités de gestion de connaissances intégrées dans deux processus issus de la norme ISO/IEC 29110, la gestion de projet (*Project Management*) et l'implémentation du logiciel (*Software Implementation*). L'enregistrement, la réutilisation, la recherche et le partage de connaissances sont facilités par l'emploi d'un noyau ontologique basé sur le modèle de référence CIDOC CRM, enrichi par la modélisation du domaine de la norme 29110.

Introduction

Cet article est une proposition d'assistant de mémoire pour les très petits projets d'ingénierie du logiciel, opérant dans le cadre de la norme ISO/IEC 29110 (ISO 29110, 2011). Un projet est réalisé selon une succession d'étapes, appelée cycle de vie, dont les activités-types sont décrites dans la norme. Durant le projet, de

nombreuses informations sont utilisées qui ne sont généralement pas conservées. L'enjeu d'un assistant de mémoire est de permettre l'enregistrement de connaissances à partir de ces informations afin de faciliter leur emploi ultérieur. Cette introduction passe en revue notre problématique et les standards d'appui.

Processus du cycle de vie du logiciel

En ingénierie du logiciel, la plupart des processus d'entreprise sont décomposés en activités et en tâches constituantes, utilisant un ensemble d'artefacts d'entrée et de sortie. La norme ISO/IEC 29110 est destinée à fournir aux très petits organismes (TPOs) qui développent des logiciels, c'est-à-dire des entreprises, des départements ou des projets jusqu'à vingt-cinq personnes, un modèle de référence des processus, activités et tâches ainsi que la nature des artefacts produits ou consommés (Laporte, 2008). Elle propose quatre profils d'utilisation de complexité croissante : Entrée, Basique, Intermédiaire et Avancé. Ainsi le profil d'entrée s'adresse aux TPOs en démarrage ou qui développent des projets d'un effort inférieur à six personnes-mois.

Gestion de connaissances pour les très petits projets

La force des très petites organisations repose sur leur personnel et peu d'effort peut-être consenti à la gestion et au transfert des connaissances. L'expertise d'un professionnel repose en partie sur la mémoire de ses expériences et il est admis qu'une collectivité dispose aussi d'une mémoire d'organisation (Walsh et Ungson, 1991) qui fournit un support aux activités de la collectivité. L'application d'une norme destinée aux TPOs du logiciel a pour but de favoriser la capitalisation du savoir-faire des individus et l'amélioration de leurs procédés de travail. Dans cette optique, un assistant de mémoire permet d'enregistrer les traces produites par les activités d'ingénierie préconisées par la norme.

Tout au long de son cycle de vie, l'ingénierie d'un produit consomme et crée des artefacts documentaires. Ce qui différencie l'ingénierie du logiciel des autres secteurs d'ingénierie est que les produits finaux (le logiciel et sa documentation) sont des documents. Par conséquent, le point de vue adopté dans cet article est que la

gestion de connaissances peut-être centrée sur les documents et les processus qui les consomment et les produisent. L'assistant de mémoire permet d'enregistrer des descriptions de ressources documentaires (les artefacts d'ingénierie) et des descriptions d'entités temporelles (les activités d'ingénierie et les événements relatifs au projet). Ce sont ces descriptions qu'on appelle connaissances dans la suite de cet article.

La description de ressources (documentaires ou non) est traditionnellement gérée à l'aide de métadonnées et les technologies employées actuellement sont celles du Web sémantique. Parmi tous les outils disponibles, nous avons choisi d'appuyer cet assistant sur un wiki défini par (Cunningham, 2002) comme « *the simplest online database that could possibly work* ». Nous avons retenu comme plate-forme pour cet assistant le logiciel *MediaWiki* doté de son extension sémantique SMW *Semantic MediaWiki*. Cette extension permet de typer sémantiquement les littéraux et les liens du wiki.

Cet assistant de mémoire opère à deux niveaux. Dans un petit projet, le chef de projet est responsable de l'approvisionnement et du contrôle des ressources et met à jour les informations de ce niveau, consultées et utilisées par l'équipe-projet. L'équipe développe les projets avec les ressources allouées et ce niveau traite de l'exécution du projet, de sa vie de tous les jours, de ses événements, des documents qui y entrent, y sont produits et en sortent. Les informations de ce niveau complètent celles de l'autre niveau et sont créées, mises à jour, cherchées, consultées et utilisées par l'équipe-projet.

Gestion d'expérience à l'aide du CIDOC CRM

Le cadre de recherche le plus général est celui de la gestion de l'expérience (*experience management*, EM), que (Ras *et al*., 2007) définissent ainsi : « *EM includes methods, techniques, and tools for identifying, collecting, documenting, packaging, storing, generalizing, reusing, adapting, and evaluating experience and for development, improvement, and execution of all knowledge-related processes* ». A l'échelle d'un TPO, les ambitions sont modestes et les méthodes, techniques et outils sont destinés à la constitution et à la gestion d'une base d'expériences plutôt qu'à la gestion de l'expérience elle-même. Il s'agit donc de pouvoir enregistrer, partager et rechercher des connaissances relatives aux projets et nous avons choisi comme

ontologie de base le CIDOC CRM (ISO 21127, 2006), modèle conceptuel de référence (*Conceptual Reference Model*) du Comité International pour la DOCumentation (CIDOC), conçu pour l'échange d'informations du patrimoine culturel. La modélisation spécifique du domaine de l'ingénierie du logiciel, lorsqu'elle est nécessaire, complète ce noyau ontologique.

A la différence de la majorité des standards documentaires qui se concentrent sur la description des artefacts et de leur environnement proche, le CIDOC CRM fait une part égale à la modélisation d'événements. Même si la description d'un objet de musée tel qu'il se présente aujourd'hui est primordiale, cette documentation est une interprétation de l'objet en relation avec un contexte historique qu'il s'agit de resituer. Le cœur du CIDOC CRM relate donc le fait qu'« il s'est passé quelque chose » comme une E7 Activité, un E63 Début d'existence, une E12 Modification, etc. C'est autour de la notion d'événement qu'on répond à des questions fondamentales : Quand (E52 Durée) est-ce arrivé ? Où (E53 Lieu) est-ce arrivé ? Qui (E39 Acteur) l'a fait ? A ou avec quoi (E77 Chose) est-ce arrivé ? L'expérience d'un projet traite autant des documents produits et consommés que de leur contexte de création et d'utilisation. Le CIDOC CRM nous permet donc de décrire chaque processus, activité ou tâche d'ingénierie du logiciel comme une instance de la classe E2 Entité temporelle (ou de ses sous-classes) en relation avec des occurrences de choses, de personnes, d'idées se rencontrant dans l'espace et dans le temps.

Positionnement du travail

Après une étude des initiatives conduites par 68 petites et moyennes entreprises (PME), (Chan et Chao, 2008) concluent que la gestion de connaissances est influencée par deux types de capacité, les infrastructures et les processus, qui doivent donc être déployés dans les PMEs.

Infrastructures de gestion documentaire

Dans un projet d'ingénierie, la gestion de la connaissance se matérialise d'abord par la gestion de la documentation. Une des

premières étapes de la gestion des documents est l'association de métadonnées à leurs contenus. L'association entre un ensemble de métadonnées et la ressource qu'il décrit peut prendre deux formes : soit les métadonnées sont contenues dans un enregistrement séparé de l'item – comme dans un catalogue de bibliothèque ; soit les métadonnées sont incluses dans la ressource (DCMI, 1999).

Par conséquent, il existe deux solutions principales pour la gestion des enregistrements de métadonnées, soit la construction d'un système de gestion indépendant, soit l'addition d'une extension au système de gestion des ressources. Devant la variété des outils d'édition et des formats de documents, il nous a paru plus simple de gérer les métadonnées indépendamment des ressources. Le terme d'annotation sémantique est synonyme de métadonnées, il fait apparaître l'activité et le rôle de l'annotateur, souvent une partie tierce qui produit de la connaissance sur les ressources annotées en créant les métadonnées associées aux ressources.

Du point de vue des infrastructures, un gestionnaire de contenu (*Content Management System*, CMS) est généralement utilisé comme support principal de la gestion des ressources documentaires. Cependant, il n'y a pas de consensus pour gérer les métadonnées.

Comme (García Alonso *et al*., 2008), nous nous situons dans le même contexte de gestion de documentation d'un projet logiciel et nous suivons le même point de vue d'utiliser un wiki plutôt qu'un CMS, car les wikis sont en général plus adaptés à une gestion collective ne nécessitant pas un contrôle éditorial. Comme (Maxwell, 2007), notre approche est d'utiliser un wiki comme système de gestion aussi simple que possible qui impose peu de règles initiales et permet d'ajouter une structure et des contraintes au fur et à mesure des besoins.

Processus de gestion de connaissances

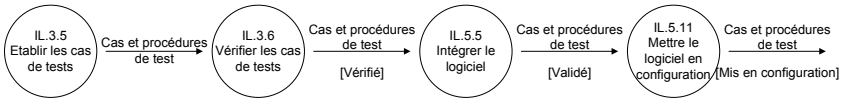
(Rech *et al*., 2007) ont identifié plusieurs défis relatifs aux processus de gestion et de transfert de connaissances dans les PMEs : enregistrement, réutilisation, recherche et partage. Notre approche est de baser la gestion de connaissances sur une ontologie appuyée sur le CIDOC CRM et d'intégrer les processus de gestion de connaissances dans les processus d'ingénierie du logiciel, notamment à l'aide d'un workflow très simple sur la documentation produite.

Processus d'ingénierie du logiciel

Les profils d'entrée et de base de la norme ISO/IEC 29110 contiennent deux processus : Gestion de projet (GP) et Implémentation du Logiciel (IL). Le processus de gestion de projet est décomposé en 4 activités : Planification du projet, Exécution du plan de projet, Évaluation et contrôle du projet, Clôture du projet. Le processus d'implémentation du logiciel est décomposé en six activités : Initiation de l'implémentation, Analyse des exigences, Identification des composants/Conception, Construction, Intégration et tests, Livraison du produit final.

La norme décrit aussi un ensemble d'artefacts d'ingénierie du logiciel associé aux activités et tâches qui les produisent et les consomment. A titre d'exemple, l'artefact « Cas et procédures de test » suit un circuit éditorial (workflow) représenté à la figure 1.

Figure 1 : Workflow de l'artefact Cas et procédures de test



Une idée essentielle de cet article est que tout projet d'un très petit organisme qui développe du logiciel soit organisé de la même manière et que les artefacts produits soient toujours rangés selon la même organisation documentaire.

Enregistrement des connaissances

(Rech *et al*., 2007) observent que les informations sur – et la connaissance relative à – des projets existent pendant l'exécution du projet mais sont souvent perdues à sa fin. Nous proposons que la gestion documentaire d'un très petit projet utilise une organisation physique (telle qu'on trouve dans un CMS) ou logique (à l'aide des catégories d'un wiki) basée sur la décomposition des processus et activités proposées dans la norme ISO/IEC 29110. La place de chaque artefact dans la structure documentaire est fixée à partir de la dernière activité qui met l'artefact à jour ; ainsi pour l'exemple de la figure 1,

l'artefact « Cas et procédures de test » est enregistré dans l'espace (logique ou physique) associé à l'activité IL 5. Intégration et tests. A cette occasion, les métadonnées de description de l'artefact – les connaissances sur cet artefact – doivent être renseignées.

Une difficulté de la gestion des métadonnées documentaires est d'être capable de mettre en relation les différentes incarnations d'un même document, telles que des versions successives ou des traductions. Nous utilisons la distinction entre Œuvre, Expression, Manifestation et Item, établie par la norme FRBR – *Functional Requirements for Bibliographic Record* (IFLA, 2009) de l'IFLA (*International Federation of Library Associations and Institutions*).

L'enregistrement des descriptions des événements historiques du projet (essentiellement qui a fait quoi, pourquoi, où et à quel moment) est opéré à deux niveaux : planification et exécution. Un TPO doit établir une planification élémentaire de chaque projet en lots de travaux, au minimum en utilisant la décomposition en deux processus et dix activités présentée au début de cette section, ou à l'aide de la décomposition de chaque activité en tâches constituantes. Quel que soit le niveau retenu, activités ou tâches, la planification doit créer une description d'un E5 Evénement (prévisionnel) pour chaque lot de travaux, avec les dates estimées de début et de fin et les ressources nécessaires. L'exécution du projet doit enregistrer son histoire en créant une description à chaque fois qu'un nouvel événement survient et en mettant à jour ces descriptions. C'est cette base historique qui permettra au TPO d'analyser les écarts entre planification et exécution et améliorer ainsi sa gestion de l'expérience.

Une pratique commune du Web sémantique est d'étendre l'usage des adresses URI (*Uniform Resource Identifier*) pour identifier non seulement des pages Web ou des documents sur le Web, mais aussi n'importe quoi. Qu'identifie alors cette adresse URI : la ressource elle-même (p. ex. une personne) ou la description de cette ressource ? Nous présentons au § 3.4 une interprétation de cette question. Dans cet article, on emploie le terme URI à la place d'adresse URI.

Réutilisation et recherche des connaissances

Les difficultés de réutilisabilité (en général) sont liées à la localisation des éléments réutilisables qui répondent à une exigence donnée, liées à l'évaluation de leur pertinence, et liées à l'adaptation

des éléments à la tâche en cours. (Ramadour et Cauvet, 2008) notent que « *the retrieval difficulty is related to the crucial problem of interaction between component providers and users* » et proposent l'utilisation d'ontologies pour guider, faciliter et adapter à l'utilisateur le processus de réutilisation. Les ontologies utilisées dans notre proposition sont présentées à la section 3. Par construction, l'emploi de la norme ISO/IEC 29110 nous fait hériter de son ontologie sous-jacente : processus, activités, tâches, rôles, produits, etc. Pour les événements historiques du projet, nous avons choisi le CIDOC CRM principalement pour sa pertinence et sa libre disponibilité.

L'implantation du CIDOC CRM dans le wiki sémantique SMW permet aux annotateurs d'indexer assez simplement leurs données à l'aide du vocabulaire du CRM. Ainsi préciser la signification du lien entre une tâche et un artefact se fait en annotant le lien avec la propriété sémantique P16 a utilisé l'objet spécifique. Le wiki convertira cette annotation en ajoutant un triple RDF <tâche, P16 a utilisé l'objet spécifique, artefact> au réseau sémantique correspondant au contenu du wiki. Rechercher les artefacts utilisés dans une tâche se fait dans une syntaxe très voisine de l'annotation textuelle et sera converti en une requête de type SPARQL sur le graphe RDF. Nous exploitons donc essentiellement un paradigme de recherche à l'aide de requêtes sémantiques (paramétrées ou paramétrables par l'utilisateur).

La stabilité d'une ressource identifiée par une URI est très relative. Ses différentes versions successives peuvent être référencées avec la même URI et la référence peut-être ambiguë. Nous précisons au § 3.4 notre interprétation de la version référencée par une URI.

Partage des connaissances

(Uren *et al*., 2006) établissent qu'un processus de gestion de connaissances centré sur les documents doit gérer trois catégories de données : ontologies, documents et annotations. Il s'agit donc de gérer le partage et le contrôle d'accès pour ces trois types. Le partage de documents se fait très naturellement par une organisation logique ou physique, associée à une gestion de droits utilisateurs. Le partage d'ontologie se fait aussi naturellement car les modifications apportées au noyau ontologique sont disponibles pour tous les utilisateurs du wiki sémantique – il se pose tout de même le problème de la propagation dans d'autres wikis utilisant la même ontologie. Après

plusieurs essais infructueux, nous avons renoncé à la gestion de portée d'annotations dans un wiki sémantique – toute annotation publiée est donc publique. Cela ne pose généralement pas de problèmes dans un cadre professionnel – surtout dans un petit groupe, mais à cause des risques de vandalisme, cela exclut sans doute l'usage d'un wiki sémantique dans un contexte plus ouvert.

Un assistant de mémoire pour supporter quoi ?

Ce travail vise essentiellement à assister les très petits projets logiciels dans la mise en œuvre des processus d'ingénierie et dans la gestion de leurs produits, essentiellement documentaires.

Processus d'ingénierie du logiciel

La norme ISO/IEC 29110 définit les processus de gestion de projet (*GP*) et d'implémentation du logiciel (*IL*) appropriées pour ces TPOs. Les processus sont décomposés en activités d'ingénierie du logiciel. Chaque activité est décomposée en tâches constituantes.

Les processus, les activités d'ingénierie, les tâches sont des activités réalisées par des personnes. En modélisation, on dirait que ce sont des sortes d'activités, ce qui indique que Processus, Activités d'ingénierie, Tâches en sont des sous-classes.

Dans le CIDOC CRM, le concept d'action intentionnelle est représentée par la classe E7 Activité (la propriété P14 réalisée par permettant d'indiquer l'agent de cette activité). E7 Activité est une sous-classe de E5 Evénement qui représente un changement d'état (les propriétés P11 a eu pour participant et P12 est arrivé en présence de permettant d'indiquer les entités persistantes affectés par ce changement d'état). E5 Evénement est une sous-classe d'E4 Période qui sert à représenter des phénomènes ou des manifestations qui sont située dans l'espace et le temps (à l'aide des propriétés P7 a eu lieu dans et P4 a pour durée).

En spécialisant Processus, Activité d'ingénierie, Tâche comme sous-classes d'E7 Activité, le noyau ontologique du CRM nous fournit les propriétés et les classes permettant d'exprimer la réponse aux questions posées dans l'introduction Quand, Où, Avec qui, A quoi ou avec quoi ? Les relations de composition entre processus et activités

d'ingénierie, entre activités d'ingénierie et tâches s'expriment à l'aide de la propriété méréologique P9 consiste en.

Bien entendu, le domaine d'application (l'ingénierie du logiciel) apporte ses propres concepts : un processus est destiné à satisfaire des objectifs (d'ingénierie) et il faut définir la classe Objectif d'ingénierie (une sous-classe d'E33 Objet linguistique) et la propriété reliant un processus à ses objectifs.

Figure 2 : Extrait de la hiérarchie des classes et de propriétés significatives

Classe	Propriétés significatives
E2 Entité temporelle	P4 a pour durée (est la durée de)
E4 Période	P7 a eu lieu dans (a été témoin de) P9 consiste en (fait partie de)
E5 Evénement	P11 a eu pour participant (a participé à) P12 est arrivé en présence de (était présent à)
E7 Activité	P14 réalisée par (a exécuté) P16 a utilisé l'objet spécifique (a été utilisée pour) P134 est la suite de (a été continuée par)
- Processus	A pour objectif
- GP	A créé le lot de travaux
- IL	A créé l'exigence
- Activité d'ingénierie	Implique le rôle
- Tâche	A pour produit d'entrée A pour produit de sortie

Un informaticien arrêterait sans doute là la modélisation conceptuelle en faisant de chaque processus d'ingénierie, tel que la gestion de projet, une instance de la classe Processus. Il faut aller plus loin car un processus de gestion de projet n'est pas un processus

d'implémentation : ce sont tous les deux des sortes de processus (donc deux sous-classes) et les spécificités conceptuelles de chacun doivent être exprimées. A titre d'exemple, une exigence logicielle décrit une capacité fonctionnelle ou technique que le logiciel doit fournir. Dans le processus d'implémentation du logiciel, il est fondamental que les exigences logicielles soient identifiées, analysées, allouées à différentes parties du logiciel, et testées. On doit donc définir la classe Exigence logicielle, sous-classe d'E89 Objet propositionnel qui représente des items immatériels qui sont documentés comme des unités individuelles ou qui sont utilisés comme sujets du discours. Il faut aussi définir les propriétés qui relient Processus d'implémentation et Exigences logicielles. De même le processus de gestion de projet manipule ses propres concepts tels que des Lots de travaux. Ainsi on spécialise une ontologie générale en une ontologie de domaine et on la rend pertinente et exploitable.

La Figure 2 présente la position des processus Gestion de Projet (GP) et Implémentation du Logiciel (IL) dans la hiérarchie des classes de l'ontologie avec des exemples de propriétés significatives (avec entre parenthèses la propriété inverse). Les classes sans code Ex et les propriétés sans code Py sont celles qui sont liées au domaine de l'ingénierie du logiciel et qui ont été ajoutées au noyau du CIDOC CRM.

Toute la difficulté de la norme ISO/IEC 29110 est d'exprimer les concepts et les bonnes pratiques pouvant apporter une aide à des TPOs vivant des réalités très différentes. A ces fins, le standard est structuré en quatre profils de difficulté croissante qui sont présentés à l'aide d'une décomposition hiérarchique en processus, activités, tâches, qu'on appelle un Modèle de Référence de Processus (*Process Reference Model, PRM*), censé fournir l'ossature de tout projet.

Une des idées principales de notre travail est que la structure du PRM est un élément essentiel de gestion de la connaissance car elle fournit aussi l'ossature de ce que Donald Schön (Schön, 1983, p. 138) appelle un répertoire – la collection d'idées, d'exemples, de situations et d'action d'un praticien réflexif. L'assistant de mémoire calque sa structure sur le PRM du profil d'entrée et fournit un premier répertoire aux employés des TPOs, un premier pas vers la pratique réflexive de l'amélioration des processus d'ingénierie.

Gestion de la documentation

Assister un très petit projet dans la gestion de sa mémoire commence par la gestion de la documentation. En tout premier lieu, cette gestion nécessite de localiser et d'identifier les documents (et plus généralement des ressources électroniques). Nous avons présenté au § 2.2 un cadre de localisation de chaque artefact d'ingénierie dans un espace associé à une activité d'ingénierie du PRM. Ce cadre se justifie par l'application de la norme. En revanche, toute liberté est laissée aux TPOs pour l'identification des documents.

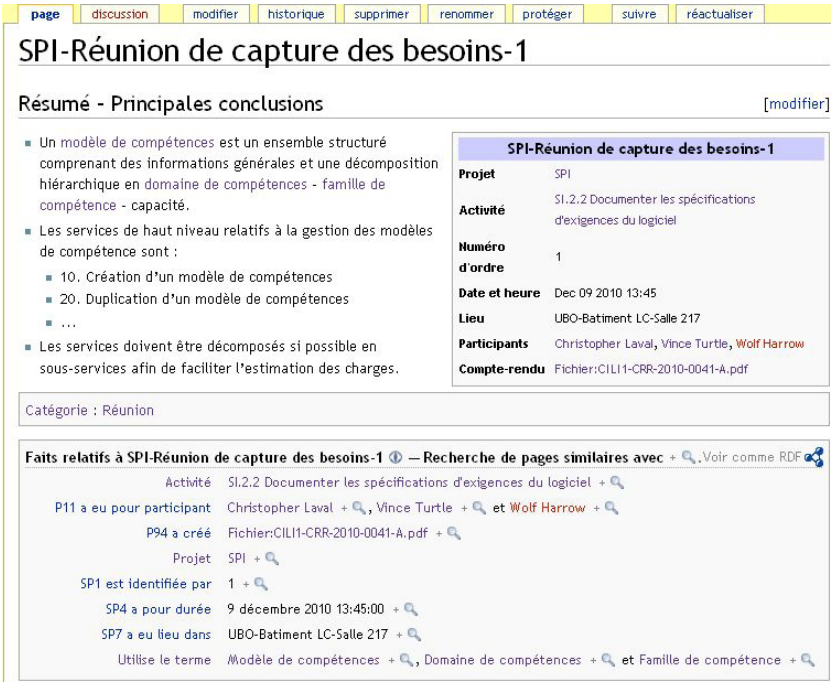
Une vraie difficulté de la gestion des métadonnées documentaires est d'être capable de mettre en relation les différentes versions d'un même document, ses différents formats, ses différentes traductions. Nous avons établi dans (Ribaud et Saliou, 2010) un classement basé sur la distinction établie dans les FRBR (IFLA, 2009). Notre proposition permet de relier les différentes incarnations d'un document à l'aide d'un schéma d'identification basé sur la distinction entre Œuvre (l'abstraction de document), Expression (ses versions), Manifestation (ses traductions et ses représentations dans différents formats), Item (ses exemplaires). Les relations entre Œuvre et Expression, Expression et Manifestation sont incluses dans le nom du document, évitant ainsi l'emploi et la mise à jour d'une partie des métadonnées.

Pour la gestion des métadonnées des documents, à la manière du Dublin Core (DCMI, 1999), telles que créateur, titre ou date de modification, le CIDOC CRM fournit la classe E31 Document, sous-classe d'E73 Objet d'information, elle-même sous-classe d'E89 Objet propositionnel et d'E90 Objet symbolique. Cependant, le CIDOC CRM n'est pas une ontologie documentaire et est assez pauvre en propriétés à ce sujet. Pour les TPO, il y a un investissement intellectuel à faire à ce sujet, pour étendre le noyau ontologique avec des propriétés pertinentes et maintenables par les employés. Le modèle des FRBR n'est pas utilisé dans cet assistant mais le travail de convergence entre FRBR et CIDOC CRM entrepris dans les FRBRoo (http://www.cidoc-crm.org/docs/frbr_oo/frbr_docs/FRBRoo_V1.0.1.doc) est une extension documentaire solide au CIDOC CRM qui peut servir de base à cet investissement.

Exploitation des métadonnées

Quels seraient les bénéfices pour les TPOs d'investir dans un assistant de mémoire ? Les TPOs vont d'abord examiner ce que leur apporte l'outillage – le wiki sémantique. Un wiki permet la mise en place d'un site partagé, que les employés peuvent faire évoluer en termes de contenu et de structure, avec un minimum d'effort d'apprentissage. Un wiki sémantique permet de formaliser la connaissance sous forme d'annotations sémantiques, modélisées sous forme de graphes RDF rattachés aux pages, formant un réseau sémantique global. L'objectif est de produire des connaissances exploitables par des agents logiciels.

Figure 3 : Un exemple de description d'un événement temporel



Prenons pour exemple la description d'une réunion d'un projet nommé SPI concernant la capture des exigences. Les métadonnées sont son numéro, sa date et son lieu, le(s) projet(s) concerné(s), la ou les activité(s) d'ingénierie du logiciel (référant à la norme 29110) impliquée(s), ses participants et le compte-rendu de réunion la relatant. On crée une page de description dans le wiki en choisissant soigneusement le nom qui devra servir d'identifiant unique parmi toutes les pages du wiki et on renseigne les valeurs de métadonnées à l'aide d'un formulaire ad hoc ou d'un patron textuel. On saisit le texte libre (ici le résumé) formaté avec la syntaxe MediaWiki et éventuellement annoté sémantiquement (ici l'utilisation du glossaire). Une fois publiée, la page apparaît comme la copie d'écran de la figure 3.

Les métadonnées sont présentées de deux manières : celles saisies via un formulaire ou un patron apparaissent dans une boîte d'informations (*InfoBox*) en début de page et l'ensemble des métadonnées dans une boîte de fait (*FactBox*) en bas de page. L'intérêt d'une *InfoBox* est qu'on peut présenter des étiquettes pour la saisie et l'affichage plus significatives pour l'utilisateur que le nom des propriétés sémantiques de l'ontologie. Ainsi l'étiquette Compte-rendu a une valeur cliquable qui mène au fichier pdf du compte-rendu et masque à l'utilisateur l'emploi complexe de la propriété P94 a créé dont le domaine est une E65 Création (une activité de création immatérielle) et l'image un E28 Objet conceptuel (le quelque chose d'immatériel qui y est créé). La boîte de faits présente les couples (propriétés, valeur) qui correspondent aux triples RDF dont la page est le sujet (au sens RDF). Les propriétés sont conformes à l'ontologie : nom, domaine et image. On y retrouve les annotations générées à partir de l'InfoBox et les annotations parsemées par l'utilisateur dans le texte libre.

Semantic MediaWiki (SMW) présente dans la *Factbox* les annotations sémantiques concernant le sujet décrit par la page. Cette boîte de fait est donc le support d'une présentation améliorée des métadonnées (comme un format de sortie de date en français) et d'une navigation améliorée (par exemple, en cliquant sur la petite loupe qui donne une liste des pages ayant la même valeur de propriété). L'outillage de SMW permet d'exécuter des requêtes sémantiques sur le réseau global. Associé à un outil complémentaire, on peut même envisager une forme limitée d'inférence de connaissances.

Cet outillage peut décider les TPOs à utiliser l'assistant de mémoire pour améliorer la gestion de connaissances grâce aux facilités d'édition, de navigation et de recherche. Les bénéfices réels, à notre avis, sont relatifs à l'amélioration des processus et à l'ingénierie de l'ontologie.

Bien que l'établissement de processus d'ingénierie du logiciel soit reconnu comme aidant les entreprises du logiciel à améliorer leur productivité et la qualité de leurs produits, les TPOs y rechignent en mettant en avant que cette approche est coûteuse en temps et en ressources. La facilité de création et d'édition de contenu, structurée par l'usage de l'ontologie, fournit un retour d'informations dont les employés n'ont pas forcément conscience. L'enregistrement de l'activité historique doit conduire les TPOs à observer leurs processus et à les documenter, ce qui est un premier pas sur la voie de l'amélioration.

L'usage d'un assistant appuyé sur une ontologie de domaine (ici la norme ISO/IEC 29110) permet d'imprégner les employés de la norme, mais surtout de les faire participer à l'ajustement de la norme à leurs besoins, en faisant évoluer l'ontologie de manière collaborative. La connaissance propre aux TPOs se formalise au travers d'un processus évolutionnaire d'ingénierie de l'ontologie, par le biais des adaptations qui y sont apportées.

Difficultés d'interprétation

Nous présentons dans cette section la réponse à deux problèmes (soulevés en 2.2) générés par l'usage courant du Web sémantique : l'interprétation des URIs pour les ressources non-documentaires et l'absence de stabilité des ressources désignées à l'aide d'URI.

Une pratique commune du Web sémantique est d'étendre l'usage des URI (*Uniform Resource Identifier*) pour identifier non seulement des pages Web ou des documents sur le Web, mais aussi n'importe quoi. Une recommandation du W3C (http://www.w3.org/TR/cooluris/) et sa traduction francophone (http://www.yoyodesign.org/doc/w3c/cooluris/#semweb) appelle les « choses sur le Web » des ressources d'information (*information resources*), et les « choses qui ne sont pas sur le Web » des ressources non-documentaires (*non-document resources*). Dans le deuxième cas, le problème est de savoir ce que cette URI identifie alors : la ressource non-documentaire elle-

même (p. ex. un concept comme la qualité ou encore un employé réel) ou la description de cette ressource non-documentaire. (Halpin, 2006) indique que l'usage d'URI pour dénoter des ressources non-documentaires pose des problèmes d'identité, de référence et de signification.

Un wiki peut héberger une (des) ressource(s) d'information et l'URI de cette ressource s'obtient en concaténant l'URI de base du wiki avec le nom de la ressource d'information. Les métadonnées relatives à la ressource d'information sont gérées dans une page de même nom, mais localisée dans un autre espace technique ; selon le contexte d'utilisation, le wiki fait la différence entre la ressource ou bien la page de description de la ressource. Lorsqu'on génère les triples RDF correspondant aux métadonnées, c'est l'URI de la ressource qui est utilisée comme sujet. Notre approche étend ce principe à chaque instance de ressource non-documentaire : à l'intérieur du wiki, son URI référence une page de description contenant ses métadonnées. Lors de la génération des triples RDF, cette URI est considérée comme référençant la ressource non-documentaire, mais si la ressource non-documentaire possède une « vraie » URI, alors les triples générés sont incorrects.

L'usage d'une URI pour identifier un document fait croire à une stabilité du document qui est très relative. Du fait que de nombreuses instances différentes (en contenu) de ce document peuvent exister au travers du temps (au fur et à mesure de ses modifications successives), nous devrions préciser quelle instance nous voulons référencer à l'aide de cette URI :

> **- l'instance courante,** qui fait référence à n'importe quelle instance de document qui peut-être obtenue dans un intervalle temporel proche de celui où on a utilisé cette URI ;
> **- une instance précise,** qu'il faudra donc identifier (parmi les autres) dans le temps.

Notre approche utilise la gestion de version de *MediaWiki* et est donc du premier type. L'URI de description d'une ressource permet d'accéder à l'instance courante de la ressource. Toutes les instances (enregistrements successifs du document) sont disponibles et identifiées par un tampon temporel (*timestamp*). Cependant, les instances ne peuvent être distinguées les unes des autres sans examiner leur contenu. Il faut donc associer à la gestion de version

de *MediaWiki* un mécanisme d'identification tel que des numéros de versions.

Perspectives et conclusion

Une des forces de l'assistant repose sur le fait que tout projet d'un très petit organisme qui développe du logiciel soit organisé de la même manière et que les artefacts produits soient toujours rangés selon la même organisation documentaire, ce qui fournit un premier modèle de répertoire d'expérience. Notre intuition – à vérifier – est que ce principe est transposable à d'autres secteurs capables de décrire leurs processus et le workflow des artefacts documentaires.

Notre proposition d'emploi d'un assistant de mémoire dans un projet sert deux objectifs : guider la mise en œuvre de la norme ISO/IEC 29110 et conserver des traces de son histoire. L'emploi de la norme prescrit l'histoire du projet en le planifiant selon le modèle de référence. Lors de l'exécution, on enregistre l'histoire effective concernant les événements et les activités du projet ainsi que les entités persistantes concernées.

Le noyau extrait du CIDOC CRM sert d'ontologie de plus haut niveau, de vocabulaire de base pour prévoir et enregistrer l'histoire. Une application de ce travail est l'emploi de ce noyau (disponible à http://saga3.referata.com) pour la rédaction de synopsis de cinéma ou de biographies.

Comme cet article s'intéresse à l'histoire d'un très petit projet d'ingénierie du logiciel guidé par la norme 29110, les entités temporelles et persistantes propres à ce domaine sont modélisées dans l'ontologie de domaine des processus *Gestion de projet* et *Implémentation du logiciel* qui s'accroche au noyau ontologique généraliste extrait du CIDOC CRM.

Un prototype de wiki sémantique « garni » avec ces deux ontologies est prêt. L'étape suivante est celle de la validation de ces ontologies pour laquelle deux options sont possibles. La première serait de déployer cet assistant dans un TPO à l'aide d'un projet-pilote. On pourrait évaluer ainsi la pertinence de cet assistant. Pour la seconde option, nous disposons de l'enregistrement d'un projet de 6 apprentis-ingénieurs sur 9 mois utilisant un dispositif plus compliqué basé sur un observatoire du cours d'action implémenté à l'aide de 3 wikis sémantiques liés entre eux et avec un CMS. Cet

observatoire a été utilisé pour reconstruire l'activité des apprentis et tenter d'évaluer si elle est conforme aux tâches prescrites (Ribaud et Saliou, 2009). La validation de cette proposition d'assistant de mémoire consisterait à réenregistrer les données historiques ainsi que les ressources produites dans l'assistant de mémoire, afin d'évaluer la pertinence de la modélisation.

VINCENT RIBAUD
LISyC – Université de Brest – UEB Brest
PHILIPPE SALIOU
LISyC – Université de Brest – UEB Brest
CLAUDE Y. LAPORTE
École de technologie supérieure – Montréal

BIBLIOGRAPHIE

Chan I. et Chao C., (2008), *Knowledge management in small and medium-sized enterprises*, in CACM, n° 51(4), pp. 83-88.

Cunningham W., (2002), *What is a Wiki* : http://www.wiki.org/wiki.cgi?WhatIsWiki, date de la dernière visite : 14 janvier 2011.

DCMI – Dublin Core Metadata Initative, (1999), *Dublin Core metadata element set, version 1.1* : http://dublincore.org/documents/1999/07/02/dces/, date de la dernière visite : 14 janvier 2011.

García Alonso J. M., Berrocal Olmeda J. J. et Murillo Rodríguez J. M., (2008), *Documentation Center : Simplifying the Documentation of Software Projects*, in Wiki4SE Workshop, Porto, 4th International Symposium on Wikis.

Halpin H., (2006), *Identity, Reference, and Meaning on the Web*, in Workshop on Identity, Reference, and Meaning on the Web, Edinburgh, WWW2006.

IFLA, (2009), *FUNCTIONAL REQUIREMENTS FOR BIBLIOGRAPHIC RECORDS* : http://www.ifla.org/VII/s13/frbr/, date de la dernière visite : 14 janvier 2011.

ISO – International Organization for Standardization, (2011), *ISO/IEC TR 29110-5-1-2 Software Engineering – Lifecycle Profiles for Very Small Entities (VSEs) – Part 5-1-2 : Management and Engineering Guide : Generic Profile Group : Basic Profile*, Geneva, ISO.

ISO – International Organization for Standardization, (2006), *ISO 21127 : 2006 Information and documentation – A reference ontology for the interchange of cultural heritage information*, Geneva, ISO.

Laporte C. Y., Alexandre S. et O'Connor R., (2008), *A Software Engineering Lifecycle Standard for Very Small Enterprises*, in EuroSPI'08, CCIS 16, Heidelberg, Springer, pp. 129-141.

Maxwell J. W., (2007), *Using wiki as a multi-mode publishing platform*, in SIGDOC '07, New York, ACM, pp. 196-200.

Ramadour P. et Cauvet C., (2008), *An Ontology-based Support for Asset Design and Reuse*, in ENC '08, Washington DC, IEEE, pp. 20-32.

Ras E., Carbon R., Decker D. et Rech J., (2007), *Experience Management Wikis for Reflective Practice in Software Capstone Projects*, in IEEE Transactions on Education, n° 50(4), pp. 312-320.

Rech J., Bogner C. et Haas V., (2007), *Using Wikis to Tackle Reuse in Software Projects*, in IEEE Software, n° 24(6), pp. 99-104.

Ribaud V. et Saliou P., (2009), *Revealing Software Engineering Theory-in-Use through the Observation of Software Engineering Apprentices' Course-of-action*, in ICCGI'09, Washington DC, IEEE Computer Society, pp. 202-210.

Ribaud V. et Saliou P., (2010), *Using a Semantic Wiki for Documentation Management in Very Small Projects*, in MTSR 2010, CCIS 108, Heidelberg, Springer, pp. 119-130.

Schön D., (1983), *The Reflective Practitioner*, New York, Basic Books, 374 p.

Uren V., Cimiano P., Iria J., Handschuh S., Vargas-Vera M., Motta E. et Ciravegna F., (2006), *Semantic annotation for knowledge management : Requirements and a survey of the state of the art*, in Web Semantics : Science, Services and Agents on the World Wide Web, n° 4(1), pp. 14-28.
Walsh J. P. et Ungson G. R., (1991), *Organizational Memory*, in Academy of Management Review, n° 16(1), pp. 57-91.

(suite de la page)

... un paradigme similaire sous tend la méthode proposée par Nathalie Pinède et David Reymond pour la réalisation d'un taxonomie de sites web universitaires. La sémantique des Unités Lexicales Hypertextes, les liens internes constituant le « menu » de navigation du site conçus librement par les webmasters, sert de base à la réalisation d'un langage de référence uqi sera ensuite géré et exploité collaborativement...

(suite page)