

N° d'ordre : 184/mag/2008

Série : 001/inf/2008

MÉMOIRE

Présenté en vue de l'obtention du diplôme de
Magister en informatique
Option : Génie logiciel & intelligence artificielle
Par : Boucetta Zouhel
Encadré par : Professeur ZIZETTE BOUFAÏDA

APPARIEMENT SÉMANTIQUE DES CVs/OFFRES D'EMPLOI DANS LE CADRE DU E-RECRUTEMENT

Soutenu publiquement le *devant le jury composé de :*

Président:	Mme Faiza Belala	Maitre de conférences	Univ. Mentouri de Constantine
Rapporteur:	Mme. Zizette Boufaida	Professeur	Univ. Mentouri de Constantine
Examineur:	Mr. Nacer eddine Zarour	Maitre de conférences	Univ. Mentouri de Constantine
Examineur:	Mr. Ramdane Mamri	Maitre de conférences	Univ. Mentouri de Constantine

Remerciement

*« Nous sommes des nains
juchés sur les épaules de géants »
Isaac Newton*

En tout premier lieu, je remercie mon dieu, tout puissant, qui m'a éclairé le bon chemin et qui m'a aidé à réaliser ce travail dans les meilleures conditions ainsi que mon encadreur Madame Zizette Boufaïda Professeur à l'université de Constantine qui m'a guidé avec patience et gentillesse et m'a fait profiter de sa grande expérience ainsi que de ses précieuses remarques qui ont grandement contribué à améliorer la qualité de ce mémoire. Qu'elle soit ici assurée de mon très grand respect.

Ensuite, j'adresse toute ma profonde gratitude à Melle Leïla YAHIAOUI qui m'a beaucoup aidé dans le développement de ce projet ainsi que dans la rédaction de mon article. Ses conseils, ses suggestions, ses commentaires et ses critiques ont contribué à l'aboutissement de ce travail.

Je tiens à remercier très sincèrement l'ensemble des membres du jury qui me font le grand honneur d'avoir accepté de juger mon travail.

Je remercie Madame Faïza Belala Maître de conférence à l'université de Constantine pour l'honneur qu'elle m'a fait en acceptant la présidence de ce jury. Qu'elle trouve donc ici l'assurance de ma profonde gratitude.

Je tiens à exprimer toute ma reconnaissance à Monsieur Nacer eddine Zarour Maître de Conférence à l'université de Constantine et Monsieur Ramdane Mamri Maître de Conférence à l'université de Constantine, pour avoir acceptées d'être examinateur de ce travail et pour le temps qu'ils ont investi à l'évaluer malgré leurs nombreuses obligations.

A Monsieur Mahmoud Boufaïda, Directeur de l'équipe SI & BC. Je le remercie de m'avoir accepté dans son équipe. Je tiens à lui témoigner toute ma reconnaissance pour le soutien permanent qu'il m'a manifesté. Ses commentaires, ses critiques et surtout son encouragement dans les moments difficiles, m'ont permis de mener ce travail à son terme.

A Monsieur Sofiane Allioua et Monsieur Tahar Madaci et Monsieur Djamil Madaci un grand merci pour votre aide.

Enfin, toute ma gratitude est adressée à mes parents qui m'ont toujours aidé et soutenu dans mes études. Je leur dédie, ainsi que mon marié et toute ma famille, mon mémoire.

Table des matières

Introduction générale	1
1. Contexte de l'étude et problématique.....	1
2. Plan du mémoire	4
Chapitre I : Le Web Sémantique et l'annotation	
1. Introduction	7
2. Le Web Sémantique.	7
3. Les Ontologies	8
3.1. Les éléments constituant une ontologie.	9
4. Méthodes à suivre pour élaborer une ontologie.....	10
4.1. ENTREPRISE.....	10
4.2. TOVE.....	10
4.3. METHONTOLOGY.....	10
4.3.1. Cycle de vie de l'ontologie selon METHONDOLOGY.....	10
5. Les langages de spécification d'ontologie	12
5.1. RDF.....	12
5.2. RDFS.	13
5.3. OWL.....	13
6. Méta données, annotation, annotation sémantique.....	14
6.1 Les méta-données	15
6.1.1 Définition	15
6.1.2 Dublin Core	16
6.2. L'annotation.	18
6.2.1 Définition.....	18
6.2.2 Élément caractérisant une annotation	18
6.2.3 Rôle des annotations	20

6.2.4 La différence entre l'annotation et une méta-donné.....	20
6.3 L'annotation sémantique.....	21
6.3.1. Définition	21
6.3.2. Le stockage des annotations.....	22
6.3.3. Représentation des annotations et la tâche d'annotation.....	23
6.3.4. Approche d'annotation.....	25
6.3.5. Rôles des annotations.	26
6.3.6. Framwork pour les annotations sémantiques.....	27
6.3.7. Outils pour l'annotation sémantique.....	28
6.3.7.1. C'est quoi un outil d'annotation.....	28
6.3.7.2. Quelques outils pour l'annotation.....	28
6.3.7.3. Synthèse.....	29
7. conclusion.....	33

Chapitre II : Gestion des documents dans le cadre du e-Recrutement

1. Introduction	35
2. Les sites d'emplois.....	35
3. CVs /offres d'emploi et outils de génération.....	38
3.1. Rubrique d'un CV	39
3.2. L'offre d'emploi	41
3.3. Outils de génération des CVs /offres d'emploi.....	41
4. Le e-Recrutement.....	43
4.1. Les phases du processus de recrutement	43
4.2. Les limite du e-Recrutement.....	44
5. La gestion des connaissances et des compétences.....	46
5.1. Gestion des connaissances (KM).....	46
5.1.1 Définition.....	46
5.1.2 Objectif de la gestion des connaissances.....	47
5.2. C'est quoi une Compétence.....	47
5.3. Gestion des compétences.....	47
6. l'appariement sémantique pour le e-Recrutement.....	48
6.1. L'appariement syntaxique des documents	49
6.2. L'appariement sémantique des documents.....	49

7. Les travaux existants.....	49
7.1. Une approche basée sur l'annotation pour modéliser le contenu sémantique.....	50
7.2. Un processus de recrutement basé ontologie.....	52
7.3. BRIDGE	54
7.4. Evaluation des travaux existants	55
8. Conclusion.....	57

Chapitre III : Appariement sémantique entre les CVs et les offres d'emploi

1. Introduction	59
2. L'objectif de notre travail.....	59
3. L'architecture du système proposé.....	60
4. Modélisation de la compétence proposée.....	62
4.1. Définition des éléments utilisée dans le modèle de la compétence proposée.....	63
4.2. Le modèle de la compétence proposée.....	63
5. La modélisation du contenu sémantique a base de l'ontologie- ER.....	65
5.1 Description de l'ontologie-ER.....	66
6. Le processus d'annotation des CVs /offre d'emploi.....	68
7. L'appariement sémantique.....	69
7.1. Convention d'écriture et formule utilisées.....	71
7.2. Principe d'appariement des qualifications personnelles.....	71
7.3. Principe d'appariement des exigences.....	72
7.4. L'appariement des compétences.....	75
7.4.1. Principe de l'appariement exact des compétences « CompExact ».....	75
7.4.2. Principe de l'appariement partiel des compétences « PartielComp ».....	76
7.4.3. Principe de l'appariement rapproché des compétences «RapprochéComp»....	79
7.4.4. Discussion sur les algorithmes d'appariements proposés.....	87
8. Conclusion	88

Chapitre IV : Construction de l'ontologie-ER

1. Introduction	90
2. Processus de construction d'une ontologie.....	90
2.1 Spécification.....	91

2.2	Conceptualisation.....	91
2.3	Formalisation.....	92
2.4	Implémentation.....	94
2.5	Evolution de l'ontologie et Test.....	94
3.	Construction d'une ontologie pour le e-Recrutement.....	95
3.1	Spécification.....	95
3.2	Conceptualisation.....	97
3.2.1	Construction du glossaire de termes.....	98
3.2.2	Construction des hiérarchies de concepts	100
3.2.3	Construction du diagramme de relations binaires.....	103
3.2.4	Dictionnaire de concepts.....	104
3.2.5	Tableau des relations binaires.....	105
3.2.6	Tableau des attributs.....	106
3.2.7	Tableau des axiomes logiques.....	107
3.2.8	Les instances des concepts et les assertions sur les instances	108
3.3	Formalisation.....	109
3.3.1	Construction de TBOX.....	109
3.3.2	Construction de ABOX.....	111
3.4	Implémentation.....	112
3.4.1	Création d'un nouveau projet OWL.....	112
3.4.2	Création des classes et la hiérarchie des classes.....	114
3.4.3	Création des propriétés.....	115
3.4.4	Création des instances.....	119
3.4.5	Génération de code.....	119
3.4.6	Test et évolution.....	120
4.	Conclusion.....	121

Chapitre V : Etude de cas et implémentation

1.	Introduction.....	123
2.	Application des services d'appariement.....	123
2.1	L'appariement Exact et partiel des compétences.....	123
2.1.1	L'appariement de l'offre d'emploi offre1 avec CV1.....	125
2.1.1	L'appariement de l'offre d'emploi offre1 avec CV2.....	127

2.1.3. L'appariement partiel des compétences entre l'offre1 avecCV1.....	129
2.1.4. L'appariement partiel des compétences l'offre 1 avec CV2.....	130
2.2 L'appariement rapproché des compétences.....	131
2.2.1 L'appariement de l'offre d'emploi offre2 avec CV3.....	135
2.2.2 L'appariement de l'offre d'emploi offre2 avec CV4.....	137
2.2.3 L'appariement de l'offre d'emploi offre2 avec CV5.....	140
3. Aspect d'implémentation.....	143
4. Conclusion.....	146
Conclusion et perspectives	147
Bibliographie	150
Glossaire	156
Annexe	157

LISTE DES FIGURES

Figure 1.1- Le cycle de vie de l'ontologie selon METHONTOLOGY.....	11
Figure 1.2- La pyramide des langages basés Web.....	12
Figure 1.3- Exemple d'un fichier RDF (graphique à gauche et XML à droite).....	13
Figure 1.4- Dublin Core dans un document XHTML.....	17
Figure 1.5- Exemple d'une annotation.....	19
Figure 1.6- Exemple d'une annotation sémantique accordée par une ontologie de référence.....	22
Figure 1.7- Une page web www.famillededris.htm	24
Figure 1.8- Exemple d'annotation sémantique en RDF.....	24
Figure 1.9- Exemple d'annotation sémantique en OWL.....	25
Figure 1.10- Exemple d'annotation avec CREAM.....	27
Figure 1.11 Annotation avec ONTOMAT.....	30
Figure 1.12 Annotation avec SMORE.....	31
Figure.1.13 Annotation avec FETCH.....	32
Figure 2.1- Architecture globale de l'ontologie-ER.....	51
Figure 2.2- Architecture globale de l'ontologie-HR	53
Figure 3.1- Architecture du système proposé.....	62
Figure 3.2 - Le Modèle de la compétence.....	65
Figure 3.3- Architecture de l'ontologie-ER.....	67
Figure 3.4- Processus d'appariement.....	69
Figure 3.5- Appariement des acquis /requis.....	70
Figure 3.6- Appariement des qualifications personnelles.....	73
Figure 3.7- Appariement des exigences	74
Figure 4.1- Syntaxe de la logique de description SHOIN.....	93
Figure.4.2- Le document de spécification.....	97
Figure 4.3- Les hiérarchies de concepts.....	101
Figure 4.4 - La hiérarchie globale des concepts de l'ontologie-ER.....	102
Figure 4.5- Le diagramme des relations binaires.....	103
Figure 4.6- Création d'un nouveau projet OWL.....	113
Figure 4.7- Interface de l'outil Protégé-OWL (version3.1.1).....	113
Figure 4.8- La hiérarchie des classes.....	114
Figure 4.9- Editeur d'expressions en logique de descriptions.....	115
Figure 4.10- Création des propriétés (datatypeProperty) des classes.....	116
Figure 4.11- Création des propriétés (objectProperty) des classes.....	117
Figure 4.12- Définition des restrictions sur les propriétés.....	118
Figure 4.13- Création d'une restriction.....	118
Figure 4.14- Définition des instances.....	119
Figure 4.15- Une partie du menu «OWL» de l'interface de Protégé-OWL.....	120
Figure 5.1. Processus d'annotation de l'offre1.....	124
Figure 5.2. Fichier des annotations de l'offre1.....	124
Figure 5.3. Fichier des annotations de CV1.....	125
Figure 5.4. Fichier des annotations de CV2.....	125
Figure 5.5. Processus d'annotation de l'offre2.....	132
Figure 5.6. Processus d'annotation de CV3.....	133

Figure 5.7. Fichier des annotations de l'offre 2	133
Figure 5.8. Fichier des annotations du CV3.....	134
Figure 5.9. Fichier des annotations du CV4.....	134
Figure 5.10. Fichier des annotations du CV5.....	134
Figure 5.11 Hiérarchie des thèmes.....	135
Figure 5.12 Un extraie du code.....	145
Figure 5.13 Résumé du code.....	145

LISTE DES TABLEAUX

Tableau 1.1- La liste des 15 éléments formant le Dublin Core.....	16
Tableau 2.1- synthèse des sites d'emploi.....	38
Tableau 2.2- Comparaison entre les différents type de CV.....	41
Tableau 2.3- Limites de e-recrutement.....	45
Tableau 4.1- Les constructeurs essentiels d'une logique de description.....	92
Tableau.4.2- Glossaire de termes.....	99
Tableau 4.3- Dictionnaire des concepts de l'ontologie-ER.....	104
Tableau 4.4- La table des relations binaires de l'ontologie-ER.....	105
Tableau 4.5- La table des attributs de concepts de l'ontologie-ER.....	106
Tableau 4.6- La table des axiomes de concepts de l'ontologie-ER	107
Tableau 4.7- La table des instances de concepts de l'ontologie-ER.....	108
Tableau 4.8- La table des attributs de concepts de l'ontologie-ER.....	109
Tableau 4.9- Définitions des concepts de l'ontologie-ER (dans TBOX).	110
Tableau 4.10- Définitions des rôles de l'ontologie-ER (dans TBOX).....	110
Tableau 4.11- Description assertionnelle de concepts.	111
Tableau 4.12- Description assertionnelle de relations.	112
Tableau 5.1. Les degrés d'appariement de l'offre1 avec CV1 et CV2 et CV3.....	131
Tableau 5.2. Les degrés d'appariement de l'offre2 avec CV4 et CV5.....	142

Introduction générale

« Les commencements ont des charmes inexprimables »

Molière

1. Contexte de l'étude et problématique

Le web a rigoureusement changé la disponibilité en ligne des données et la quantité de l'information électronique échangée. Il a révolutionné l'accès à l'information et la gestion des connaissances dans les grands organismes. Le recrutement électronique est l'une des applications typiques d'une telle approche de gestion de la connaissance à travers le web.

Internet a prouvé que les méthodes de recrutement classiques qui utilisent les annonces d'emploi dans les journaux et les magasins, qui sollicitent les agences de recrutement et qui passent par l'inscription dans les sociétés de recherche, sont trop lentes, chères et insuffisantes dans leurs capacités de fournir des candidats de haute qualité dans les plus brefs délais.

Le Web a permis une évolution importante du marché du recrutement en ligne, il s'est transformé rapidement en un outil de recrutement performant. 53% des recrutements en Allemagne sont le résultat d'une postulation en ligne [BEZI05]. Un chiffre qui est susceptible d'augmenter. Cependant cette révolution du marché du travail basée sur l'utilisation des technologies de l'information n'est pas accompagnée par une évolution des outils consacrés à la recherche et à la gestion des CVs et des offres du travail. En effet, les techniques dédiées à l'analyse des ressources de l'information liées au recrutement, restent très rudimentaires. Par exemple le moteur de recherche fourni par Monster ([http:// monster.com](http://monster.com)) a une technique de recherche basée simplement sur la combinaison des mots clés. Vu que cette approche est purement syntaxique, alors elle est limitée pour permettre aux recruteurs de trouver la meilleure personne pour leurs besoins et pour permettre aux chercheurs d'emploi de trouver l'offre qui correspond le plus à leurs profils.

Une solution peut être trouvée dans le Web sémantique qui vise à enrichir les structures syntaxiques du Web actuel avec leur contenu sémantique. Par conséquent, l'exploitation de ces technologies dans le cadre du recrutement électronique semble être bénéfique, surtout

pour assurer un rapprochement automatique entre les offres et les demandes d'emploi. Le principe consiste à expliciter le contenu de ces documents à l'aide d'éléments ontologiques formant un référentiel commun (ontologie), inspiré des parties communes entre ces documents. Ce référentiel peut être utilisé par un recruteur (ou un chercheur d'emploi) pour annoter son document avec des éléments pertinents.

Les annotations résultantes peuvent être exploitées pour automatiser le processus de rapprochement entre son document et les CVs (offres d'emploi) disponibles.

Plusieurs travaux concernant le e-recrutement se basent sur la technique d'annotation sémantique, du fait de sa simplicité et de sa capacité d'expliquer le contenu des documents.

Néanmoins, ces travaux sont en cours de réalisation, alors les solutions présentées ne sont pas complètes.

Le travail de magister de Leila Yahiaoui [YAHIO6a], préparé au département informatique de l'université de Constantine est le plus complet au niveau modélisation. Mais ce dernier souffre de quelques lacunes, comme la négligence du contexte de la compétence (organisationnel et économique) dans la modélisation de la compétence. Par ailleurs, des critères importants tels que certaines exigences (type de contrat, type emploi, rotation, déplacement, salaire), n'ont pas été considérés. Ceci a engendré quelques faiblesses dans le processus d'appariement entre les CVs et les offres d'emploi. De plus, nous considérons dans notre travail, des poids de pondération à appliquer aux exigences (type de contrat, salaire, etc.), aux compétences (sous-jacentes aux métiers/diplômes) et aux qualifications personnelles (âge, sexe, etc.) annotant le document de l'utilisateur (recruteur ou chercheur d'emploi), et qui seront explicitement spécifiés par ce dernier afin d'exprimer son choix.

2. Notre contribution

Le but de ce travail consiste à proposer des services d'appariement simple mais efficace entre les offres d'emplois et les CVs annotés. Ces documents sont annotés à l'aide d'éléments ontologiques formant un référentiel commun (ontologie), inspiré des parties communes entre ces documents (diplôme, expérience professionnelle, compétence, qualification personnelle, exigence).

Afin d'aboutir à un tel objectif, nous décrivons l'architecture globale du système proposé. Dans cette perspective, nous construisons une ontologie nommée « Ontologie-ER » dont le modèle de compétence représente le cœur de cette dernière. Les concepts et les relations entre eux sont inspirés des parties communes les plus significatives des CVs et des offres d'emploi

(qualification personnelle, exigence, diplôme, expérience professionnelle, compétence). Le domaine choisi est « l'Informatique et la Télécommunication ». Pour décrire les métiers, nous allons baser sur le référentiel des métiers CIGREF [CIGR05] et pour décrire les diplômes sur le référentiel algérien du LMD en informatique [LMD04], nous allons utiliser également les référentiels des compétences [CIGR05], [LMD04], [YAHIA06a] et l'ontologie IT-SKILL et les avis des experts pour décrire les compétences.

La méthodologie METHONTOLOGY [FERN97] est adoptée pour la construction de l'ontologie-ER et la logique de description SHIQ pour sa formalisation. Nous allons implémenter et tester l'ontologie via les outils PROTEGE OWL et le raisonneur RACER.

Le processus d'annotation que nous allons proposer aux recruteurs et aux chercheurs d'emplois consiste à instancier l'ontologie-ER à travers l'outil d'annotation proposé par Allioua Sofiane [ALLI08] qui est dédié à l'annotation des images, nous l'avons adapté pour la notation des documents HTML. Les annotations résultantes seront exploitées pour automatiser le rapprochement entre l'offre et les CVs disponibles (et vice versa). Dans ce mémoire, nous allons proposer des techniques d'appariement sémantique entre CVs et offres d'emploi, dans lesquelles l'appariement des compétences est prépondérant. L'appariement sémantique des requis d'un recruteur et des acquis d'un chercheur d'emploi revient à calculer leur degré de rapprochement selon les trois critères suivants: qualifications personnelles, exigences et compétences. Vu l'importance des compétences dans la sélection du chercheur d'emploi le plus qualifié ou de l'offre d'emploi la plus intéressante, nous allons proposer trois techniques d'appariement des compétences : l'appariement exact (CompExact) et l'appariement partiel (CompPartiel) et l'appariement rapproché (RapprochéComp). Le résultat de l'appariement entre une offre d'emploi et un CV est un ensemble de quintuplés (URI, DA_{Final} , $DA_{QPersonnel}$, $DA_{Exigence}$, $DA_{compétence}$) trié par le degré d'appariement final (DA_{final}) suivi par les trois autres degrés ($DA_{compétence}$, $DA_{QPersonnel}$, $DA_{Exigence}$) leur ordre de tri est en fonction du choix de l'utilisateur, l'URI est l'identifiant d'un CV ou d'une offre d'emploi, DA_{final} est le degré d'appariement de tous les acquis avec tous les requis, $DA_{QPersonnel}$ est le degré d'appariement selon les qualifications personnelles, $DA_{Exigence}$ est le degré d'appariement selon les exigences, $DA_{compétence}$ est le degré d'appariement selon les compétences. Afin de montrer l'applicabilité des services proposés nous allons présenter une étude de cas, nous allons prendre en considération dans cette étude les différents cas particuliers. Aussi nous allons détailler les aspects d'implémentation et les outils logiciels

utilisés pour le développement des services proposés. Nous allons terminer ce travail par une conclusion et des perspectives pour des travaux futurs.

3. Plan du mémoire

Pour présenter notre travail, nous suggérons un plan de lecture divisé en cinq (05) chapitres :

Chapitre 1 : Le Web Sémantique et l'annotation des documents (état de l'art)

Ce chapitre introduit le Web sémantique et spécialement les ontologies. L'annotation classique sera brièvement exposée, tandis qu'une grande partie du chapitre sera dédiée à l'annotation sémantique car c'est celle qui intéresse le plus ce travail.

Chapitre 2 : Gestion des documents dans le cadre du e-recrutement (état de l'art)

Ici nous insistons sur le processus de recrutement électronique et sur ses limites. Nous allons détailler quelques principes de gestion des connaissances et des compétences. Puis nous allons définir l'appariement sémantique et nous allons présenter, une étude bibliographique sur les travaux qui ont abordé la problématique de l'appariement des CVs et offres d'emploi par l'utilisation des technologies du web sémantique.

Chapitre 3 : Appariement sémantique des documents à base d'ontologie

Notre contribution est détaillée dans ce chapitre. En premier lieu, nous allons exposer l'architecture globale du système proposé, puis seront détaillés, le modèle de la compétence ainsi que l'architecture globale de l'ontologie et le processus de l'annotation sémantique. Nous allons terminer ce chapitre par la description des services d'appariement sémantique à appliquer entre les documents et qui cernent les trois aspects : i) les qualifications personnelles, ii) les exigences, iii) les compétences. Le résultat de l'appariement entre une offre d'emploi et un CV est un ensemble de quintuplés $(URI, DA_{final}, DA_{QPersonnel}, DA_{Exigence}, DA_{compétence})$ trié par le degré d'appariement final (DA_{final}) suivi par les trois autres degrés ($DA_{compétence}, DA_{QPersonnel}, DA_{Exigence}$) leur ordre de tri est en fonction du choix de l'utilisateur, l'URI est l'identifiant d'un CV ou d'une offre d'emploi, DA_{final} est le degré d'appariement de tous les acquis avec tous les requis, $DA_{QPersonnel}$ est le degré d'appariement selon les qualifications personnelles, $DA_{Exigence}$ est le degré d'appariement selon les exigences, $DA_{compétence}$ est le degré d'appariement selon les compétences.

Chapitre 4 : Processus de construction de l'ontologie-ER

Dans ce chapitre, nous allons modéliser l'ontologie-ER, en suivant les étapes de la méthodologie «Methodology ». La logique de description est adaptée pour la formalisation de cette dernière. Nous allons implémenter et tester l'ontologie en utilisant les outils PROTEGE OWL et le raisonneur RACER.

Chapitre 5 : Étude de cas

Pour valider notre approche, nous allons présenter, dans ce chapitre, une étude de cas. Des offres d'emploi et des CVs seront **choisies** pour appliquer les trois types des services d'appariement proposés i) les qualifications personnelles, ii) les exigences, iii) les compétences. Dans le cas des compétences, nous allons appliquer également les trois techniques proposées à savoir : l'appariement exact, l'appariement partiel, et l'appariement rapproché, afin de montrer la particularité de chaque technique et son usage.

Nous allons implémenter les algorithmes proposés par l'exploitation des APIs de protégé-OWL et du raisonneur RACER dans une application JAVA.

Enfin, nous allons terminer ce mémoire par une conclusion générale dans laquelle, nous allons présenter un bilan du travail effectué ainsi que les perspectives et les travaux futurs, méritant d'être effectués pour améliorer et approfondir cette étude.

Chapitre 1

Le Web Sémantique et l'annotation

Sommaire

1. Introduction.
 2. Le Web Sémantique.
 3. Les Ontologies.
 - 3.1. Les éléments constituant une ontologie.
 4. Méthodes à suivre pour élaborer une ontologie.
 - 4.1. ENTREPRISE.
 - 4.2. TOVE.
 - 4.3. METHONTOLOGY.
 - 4.3.1. Processus de construction.
 5. Les langages de spécification d'ontologie.
 - 5.1. RDF.
 - 5.2. RDFS.
 - 5.3. OWL.
 6. Méta-donnée, annotation, annotation sémantique
 - 6.1 Les méta-données.
 - 6.1.1 Définition.
 - 6.1.2 Dublin Core.
 - 6.2 L'annotation.
 - 6.2.1 Définition.
 - 6.2.2 Élément caractérisant une annotation.
 - 6.2.3 Rôle des annotations.
 - 6.2.4 La différence entre l'annotation et une méta-donnée.
 - 6.3 L'annotation sémantique.
 - 6.3.1. Définition.
 - 6.3.2. Le stockage des annotations.
 - 6.3.3. Représentation des annotations et la tâche d'annotation.
 - 6.3.4. Approche d'annotation.
 - 6.3.5. Rôles des annotations.
 - 6.3.6. Framwork pour les annotations sémantiques.
 - 6.3.7. Outils pour l'annotation sémantique.
 - 6.3.7.1. C'est quoi un outil d'annotation.
 - 6.3.7.2. Quelques outils pour l'annotation.
 - 6.3.7.3. Synthèse.
 7. conclusion
-

"The Semantic Web is an extension of the current web in which information is given well defined meaning, better enabling computers and people to work in cooperation."

Tim Berners-Lee

1. Introduction

L'un des objectifs du Web Sémantique, est de décrire le contenu sémantique des ressources Web en les annotant avec des informations non ambiguës [PRIE04]. Ces annotations sont représentées dans un formalisme qui permet aux machines d'exploiter automatiquement le contenu sémantique des ressources pour réaliser des tâches variées telles que la recherche améliorée (en termes de précision), la découverte de ressources, et le filtrage d'informations.

La représentation explicite des contenus des ressources documentaires du Web est rendue possible grâce notamment aux ontologies, lesquelles représentent une technologie clé pour la mise en oeuvre de ce Web Sémantique. Elles ont été développées en Intelligence Artificielle pour faciliter le partage de la connaissance et leur réutilisation.

La tâche d'annotation pour le Web Sémantique consiste donc à prendre en entrée une ressource Web et fournir en sortie la ressource enrichie par des annotations sémantiques basées sur des ontologies de référence.

Dans la section 2 de ce chapitre, nous allons introduire des généralités sur le web sémantique et les ontologies. Les notions sur les méta-données, les annotations et surtout les annotations sémantiques seront bien détaillées dans ce qui suit. Pour finir, une conclusion synthétisera notre compréhension des travaux liés au web sémantique, aux ontologies et à l'annotation.

2. Le Web Sémantique

Selon Tim Berners-Lee « Le Web sémantique est une extension du Web actuel, dans lequel l'information reçoit une signification bien définie, améliorant les possibilités de travail collaboratif entre les ordinateurs et les personnes »

Cela signifie que l'information n'a pas de signification bien définie dans le Web actuel. C'est effectivement vrai puisque la majeure partie de l'information sur le Web est sous forme textuelle, très peu structurée et donc inutilisable pour faire des traitements de calcul ou d'inférences. Il est pourtant bien évident que l'information disponible sur le Web actuel a une signification, mais qu'elle n'est accessible aujourd'hui qu'à des lecteurs humains.

Le Web sémantique vise à cacher le manque de sémantique des documents du Web traditionnel, pour permettre aux ordinateurs et aux personnes de mieux travailler en coopération. Pour la lisibilité par une machine des documents et leur traitement automatique, ceci nous mène au fait qu'une certaine sorte de données additionnelles, indépendamment du contenu réel des pages Web, est exigée pour que n'importe quel outil puisse accéder à ces données, les traiter selon ses conditions et présenter ses résultats à l'utilisateur avec plus de pertinence et d'intelligence plutôt qu'un texte simple qui résulte de la recherche par mot-clé. Ces données additionnelles sont les méta-données (annotation). Leur représentation s'appuie sur des extensions des fonctionnalités des langages de marquage que le World Wide Web Consortium (W3C) a déjà publié comme des standards (RDF, OWL.....).

3. Les ontologies

Le terme « ontologie » est emprunté du domaine de la philosophie où il signifie «explication systématique de l'existence».

Dans le contexte de l'intelligence artificielle, il n'existe pas une définition commune du terme ontologie [GUAR97], [GAND02]. Une des raisons évoquées est que les ontologies se retrouvent dans plusieurs champs d'étude : ingénierie des connaissances, conception de base de données, représentation des connaissances, système à base de connaissances et recherches d'informations. La popularité des ontologies provient de ce qu'elles permettent une compréhension commune et partageable d'un domaine qui peut être communiqué à des humains et des ordinateurs. La définition la plus souvent rencontrée est celle de Gruber [GRUB92] : «une ontologie est une spécification explicite de la conceptualisation», une conceptualisation est la vision simplifiée du monde que l'on veut représenter.

Guarino, dans [GUAR97], souligne l'ambiguïté du terme conceptualisation qui doit être pris dans son sens intuitif et propose la définition suivante pour tenir compte du caractère subjectif : «une ontologie est un compte rendu explicite et partiel de la conceptualisation».

Pour [GRÜN95] « une ontologie est une description formelle d'entités et leurs propriétés, relations, contraintes, comportements ». Elle est simplifiée dans [IKED99] où une ontologie est définie comme « un ensemble de définitions de concepts et leurs relations, à ne pas confondre avec un modèle qui est un ensemble d'instances de ces concepts ».

Cependant ce qui importe le plus à propos des ontologies, ce ne sont pas leurs définitions mais de savoir à quoi elles servent. Le but, lorsqu'on utilise des ontologies est de permettre aux connaissances d'être partagées et réutilisées.

3.1. Les éléments constituant une ontologie

Les connaissances traduites par une ontologie sont véhiculées à l'aide des Concepts, des Relations, des Fonctions, des Axiomes, des Instances [GOME99].

§ **Concepts** : ils sont appelés aussi termes ou classes de l'ontologie. Il existe diverses manières de définir un concept. Un concept est un constituant de la pensée (un principe, une idée, une notion abstraite) sémantiquement évaluable et communicable. L'ensemble des propriétés d'un concept constitue sa compréhension ou son intension et l'ensemble des êtres qu'il englobe, son extension.

Ces concepts peuvent être classifiés selon plusieurs dimensions [GOME99]:

- 1) niveau d'abstraction (concret ou abstrait).
- 2) atomicité (élémentaire ou composée).
- 3) niveau de réalité (réel ou fictif).

§ **Fonctions** : elles sont des cas particuliers de relations dans lesquelles le nième élément de la relation est défini de manière unique à partir des n-1 premier.

Elles se définissent formellement à partir d'un produit de n concepts :

$$F : C_1 \times C_2 \times \dots \times C_{n-1} \longrightarrow C_n.$$

Comme exemple de fonctions binaires, nous avons la fonction mère et le carré, et comme exemple de fonction ternaire, le prix d'une voiture usagée sur lequel on peut se baser pour calculer le prix d'une voiture d'occasion en fonction de son modèle, de sa date de construction et de son kilométrage. [LOCH99]

§ **Relations** : elles traduisent les associations existant entre les concepts présents dans le segment analysé de la réalité. Ces relations regroupent les associations suivantes :

- sous-classe-de (spécialisation, généralisation).
- partie-de (agrégation ou composition).
- associée-à ; instance-de etc.

Ces relations nous permettent d'apercevoir la structuration et l'interrelation des concepts, les uns par rapport aux autres.

§ **Axiomes** : une ontologie est en outre composée d'axiomes, qui forment des contraintes sémantiques pour le raisonnement et donnent un acompte d'une conceptualisation. Ils prennent la forme d'une théorie logique.

§ **Instances** : elles constituent la définition extensionnelle de l'ontologie, ces objets véhiculent les connaissances à propos du domaine du problème.

L'ensemble formé par les concepts, leurs instances, les relations entre les concepts et les axiomes constitue une base de connaissances.

4. Méthodes à suivre pour élaborer une ontologie

Plusieurs méthodologies ont été définies pour cadrer le développement d'ontologies, nous citons d'après [LOCH99]:

4.1. Entreprise [USCH95]

Proposée par Uschold et King la première méthode d'ingénierie "générale", résultat de leurs travaux de construction d'ontologies dans le domaine de la gestion des entreprises. Initialement, cette méthode reposait sur quatre étapes :

- Identifier le but et la portée de l'ontologie.
- Construire l'ontologie : capturer les connaissances, coder, réutiliser et intégrer des ontologies existantes.
- Evaluer l'ontologie.
- Documenter l'ontologie.

Les étapes et les sous-tâches de la méthode ENTERPRISE, sont décrites de façon abstraite. Les techniques utilisées pour les sous-tâches ne sont pas précisées (par exemple : Comment identifier les concepts fondamentaux ? Quel est le langage utilisé pour représenter l'ontologie ?)

4.2. Tove [GRUN95]

L'ontologie est construite à partir des scénarios d'entreprises pour lesquels elle sera utilisée. Cette méthodologie reste sommaire et aucune étape n'est décrite par rapport aux techniques qui peuvent y être employées. De plus, elle est spécialisée dans la spécification d'ontologies pour les entreprises.

4.3. Methontology [FERN97]

Cette méthode est développée au laboratoire d'intelligence artificielle. Elle vise la construction d'ontologie au niveau de connaissance. Ce projet a été motivé par le constat suivant : l'absence de méthodes ou de guides structurés est un obstacle à la construction d'ontologies partagées et consensuelles. Il est également un obstacle à l'extension d'une ontologie existante ou à sa réutilisation dans d'autres ontologies. L'approche METHONTOLOGY distingue plusieurs étapes dans son processus de construction. Elles sont détaillées dans la section suivante.

4.3.1. Processus de construction

Le processus de construction peut se résumer aux phases suivantes voir (Figure1.1):

- § **Spécification** : cette étape a pour but de fournir une description claire du problème étudié ainsi que la façon de le résoudre. Elle permet de préciser l'objectif, la portée et le degré de granularité de l'ontologie qui sera construite.
- § **Conceptualisation** : l'objectif est d'organiser et de structurer la connaissance acquise durant l'étape de spécification en utilisant des représentations externes qui sont indépendantes des paradigmes de représentation de connaissances et des langages d'implémentation dans lesquels l'ontologie va être formalisée et implémentée.
- L'idée est de combler graduellement le canal entre les moyens d'expressions des intéressés et les langages d'implantation des ontologies. Les représentations intermédiaires utilisées sont : les taxonomies de concepts, les diagrammes des relations binaires, le glossaire des termes, le dictionnaire des concepts, le tableau des relations binaires, spécifier des contraintes sur les attributs dans une table d'attributs, spécifier des axiomes sur les concepts dans une table d'axiomes logiques, décrire les instances des concepts dans une table d'instances.
- § **Formalisation** : cette étape consiste en la transcription du modèle conceptuel de l'ontologie dans un langage formel de représentation de connaissances.
- § **Implémentation** : elle consiste à la codification de l'ontologie formelle dans un langage opérationnel du Web Sémantique.
- § **Maintenance** : cela peut s'agir d'une maintenance corrective ou évolutive de l'ontologie (nouveaux besoins de l'utilisateur), ce qui permet la validation et l'évolution de celle-ci. Cette activité est généralement faite par le constructeur et des experts du domaine. La validation se base sur l'exploitation des services d'inférences associés aux LDs, et qui sont offerts par des raisonneurs.

Pour conclure, nous avons constaté que la démarche METHONTOLOGY présente un certain nombre de phases spécifiées de manière très détaillée, notamment la phase de conceptualisation. De ce fait, nous allons adopter cette méthodologie pour notre travail.

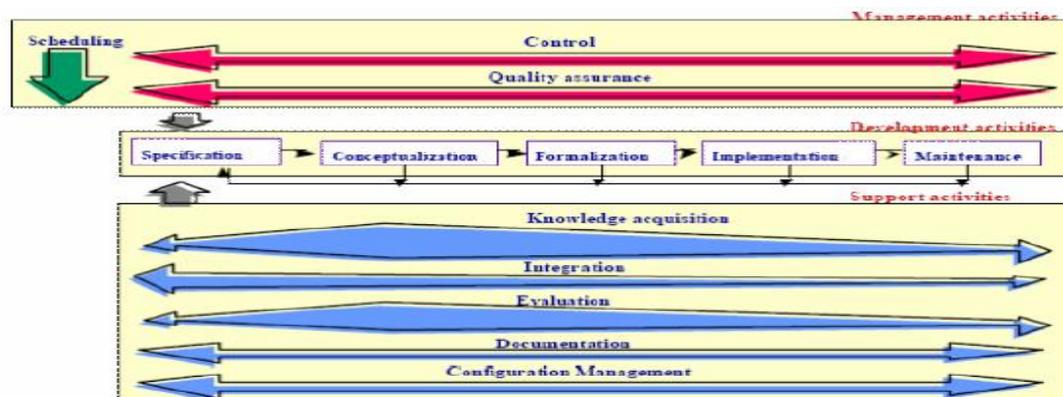


Figure1.1- Processus de construction de l'ontologie selon METHONTOLOGY.

5. Les Langages de spécification d'ontologie

Plusieurs langages de spécification d'ontologies (ou langages d'ontologies) ont été développés pendant les dernières années, et ils deviendront sûrement des langages d'ontologie dans le contexte du Web sémantique. Certains d'entre eux sont basés sur la syntaxe de XML, tels que XOL (Ontology Exchange Language), SHOE (Simple HTML Ontology Extension - qui a été précédemment basé sur le HTML), OML (Ontology Markup Language), RDF (Resource Description Framework), RDF Schéma. Les deux derniers sont des langages créés par des groupes de travail du World Wide Web Consortium (W3C).

En conclusion, trois langages additionnels sont établis sur RDF(S) pour améliorer ses caractéristiques: OIL (Ontology Inference Layer), DAML+OIL et OWL (Web Ontology Language). La figure 1.2 présente des langages de spécification d'ontologie, qui ont été récemment développés. La figure ci-dessous représente les rapports principaux entre tous ces langages sous la forme d'une pyramide des langages du Web sémantique.

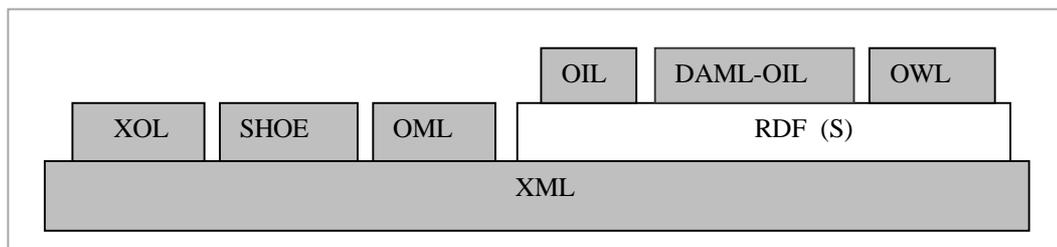


Figure 1.2- La pyramide des langages basés Web.

5.1. RDF (Resource Description Framework)

RDF est un langage de représentation des ressources du web, et en particulier les métadonnées telles que (le titre, l'auteur, la date de modification d'une page, etc.). Son objectif principal est le développement du « Web sémantique », afin de faciliter le traitement automatique des ressources Web. Différents domaines sont concernés, particulièrement la découverte de ressources pour améliorer l'efficacité des moteurs de recherche [TIXI01].

Un document RDF est un ensemble de triplets de la forme < sujet, prédicat, objet >.

1) *le sujet* représente la ressource décrite, i.e. tout document accessible sur le Web comme les pages HTML, les documents textuels (PDF, Ms Word) ou multimédias (images, vidéo), etc., mais aussi tout objet, abstrait ou non, du monde réel. Les ressources sont nommées en utilisant une URI.

2) *le prédicat* représente la propriété descriptive, i.e. une caractéristique spécifique, un attribut ou une relation, utilisée pour décrire une ressource.

3) *l'objet* représente la valeur de cette propriété, soit une valeur littérale, comme un nombre entier ou une chaîne de caractères, soit une autre ressource accessible par son URI. Par contre, une valeur littérale ne peut en aucun cas être le sujet d'un énoncé.

Cet ensemble de triplets peut être représenté de façon naturelle par un multi graphe orienté, étiqueté où les éléments apparaissant comme sujet ou objet sont des sommets, et chaque triplet est représenté par un arc dont l'origine est son sujet et la destination est son objet (voir figure 1.3).

Par contre, RDF ne fournit pas de mécanisme de contrainte de classes ou de types pour les différentes parties du triplet. Il n'est donc pas assez puissant pour représenter de vraies ontologies avec un système de raisonnement approprié.

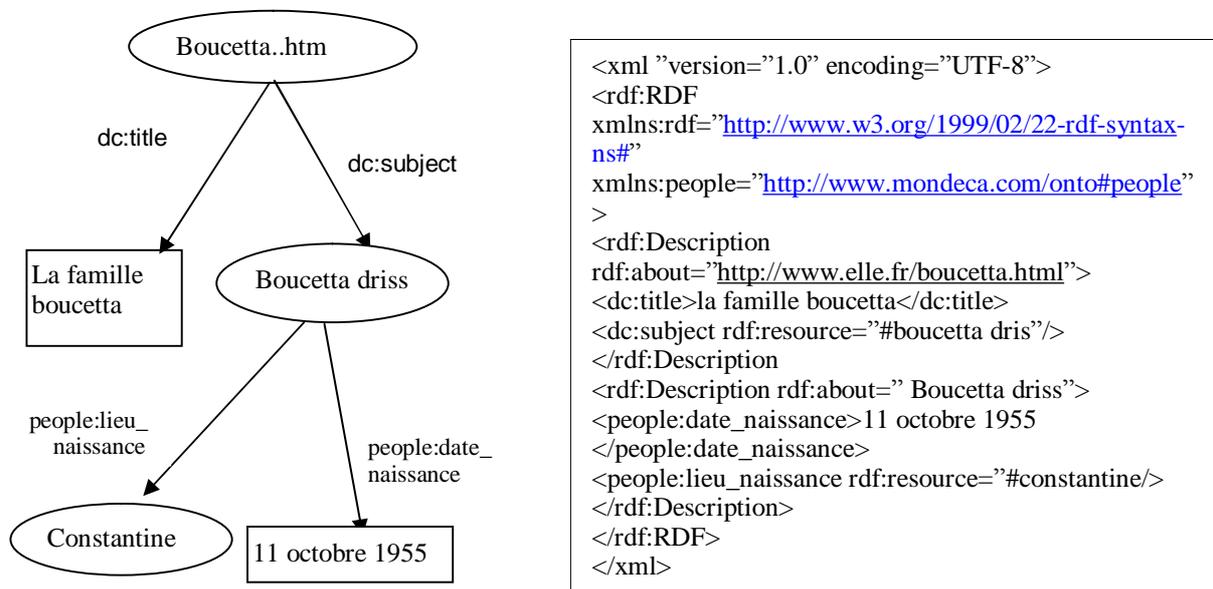


Figure 1.3- Exemple d'un fichier RDF (graphique à gauche et XML à droite).

5.2. RDFS (Resource Description Framework Schema)

Un schéma RDF est un ensemble de déclarations de classes et de propriétés. Il s'apparente en cela à la partie terminologique d'une logique de description où sont déclarés les rôles et les concepts d'une base de connaissances. Il permet en outre, comme dans les langages de Représentation de Connaissance par Objet, de définir les domaines et les co-domaines (domaines de valeurs) des propriétés RDF ainsi que des hiérarchies de classes. Il permet également de représenter des hiérarchies de propriétés, celles-ci étant considérées sur le même plan que les classes.

5.3. OWL: (Web Ontology Language) [W3C04]

Le problème de RDF est qu'il n'introduit aucune normalisation sur la sémantique

proprement dite, et que divers utilisateurs pourront donner des noms différents à des entités de même sens, ce qui empêche d'imaginer toute agrégation entre des données provenant de sources Web différentes.

La solution vient du développement d'ontologies, qui décrivent formellement les termes et les relations entre eux. OWL a été conçu pour satisfaire le besoin d'un langage d'ontologie du Web, il est basé sur la recherche effectuée dans le domaine de la logique de description, il repose sur la syntaxe des triplets RDF et réutilise certaines des constructions RDFS, il ajoute plus de vocabulaire pour décrire les propriétés et les classes.

On peut citer entre autre : les relations entre classes (par exemple la disjonction), les cardinalités (par exemple exactement un), l'égalité, typage plus riche des propriétés, caractéristiques des propriétés (par exemple la symétrie) et les classes énumérées.

OWL a été fractionné en trois sous langages ou chacun est une extension par rapport à son prédécesseur :

- OWL LITE : répond à des besoins de hiérarchie de classification et de fonctionnalité de contraintes simples de cardinalité 0 ou 1. Une cardinalité 0 ou 1 correspond à des relations fonctionnelles, par exemple, une personne a une adresse. Toutefois, cette personne peut avoir un ou plusieurs prénoms, OWL Lite ne suffit donc pas pour cette situation.
- OWL DL : concerne les utilisateurs qui souhaitent une expressivité maximum couplée à la complétude du calcul (cela signifie que toutes les inférences seront assurées d'être prises en compte) et la décidabilité du système de raisonnement (c'est-à-dire que tous les calculs seront terminés dans un intervalle de temps fini). Ce langage inclut toutes les structures OWL avec certaines restrictions, comme la séparation des types: une classe ne peut pas aussi être un individu ou une propriété. Il est nommé DL car il correspond à la logique descriptive.
- OWL FULL : se destine aux personnes souhaitant une expressivité maximale. Il a l'avantage de la compatibilité complète avec RDF/RDFS, mais l'inconvénient d'avoir un haut niveau de capacité de description, quitte à ne pas pouvoir garantir la complétude et la décidabilité des calculs liés à l'ontologie.

6. Méta-donnée, annotation, annotation sémantique

Une ressource doit être bien décrite, sinon elle peut demeurer pratiquement inexploitable et impossible à retrouver. L'annotation des documents est une solution à ce problème. Les annotations utilisées pour annoter un document peuvent être en langage libre ou formel

c'est-à-dire des éléments d'une ontologie. Par la suite, nous détaillerons les différents types d'annotation.

6. 1. Les méta-données Le terme "méta" vient du grec et dénote quelque chose de nature plus élevée ou plus fondamentale. Nous évoquons par la suite la définition des méta-données, puis nous présentons le modèles de méta-données Dublin Core.

6.1.1. Définition

On utilise généralement les méta-données pour parler d'informations descriptives à propos de ressources du Web. Plusieurs définitions sont présentées dans la littérature pour les méta-données. On va exposer quelques unes :

- «Une description normalisée attachée à une ressource identifiée (sur le Web notamment) » [PRIE04].

- «Une méta-donnée est une donnée sur une donnée » [HAND05]. Une méta-donnée est attachée à une ressource identifiée en tant que telle sur le Web et aura plutôt une pertinence à priori et sera plutôt saisie suivant un schéma. Par exemple, le titre d'un document, l'auteur d'un document, qui permettront de mettre en place des inférences.

Les méta-données c'est un ensemble structuré (ou pas) de données décrivant une ressource quelconque. Une méta-donnée permet de donner un sens au contenu des ressources de manière à ce que leur localisation et interrogation soient plus aisées et plus pertinentes. Ainsi, les méta-données constituent des points d'accès qui permettent à un document d'être repéré facilement. Selon Weibel et Lagoze « l'association de méta-données descriptives standardisées avec des objets en réseau offre un potentiel d'amélioration substantiel des possibilités de découverte de ressources en permettant des recherches basées sur des champs (e.g.,auteur, titre), en permettant l'indexation d'objets non-textuels et en permettant l'accès à un contenu de substitution, ce qui est différent de l'accès au contenu de la ressource elle-même » [WEIB97], d'où leur intérêt majeur dans la localisation rapide des ressources sur le web.

Les méta-données peuvent être évidentes telles que (l'auteur, la date de publication, l'éditeur d'un livre, etc.), ou plus complexes et moins facilement définies ainsi que la gestion des avis d'un collectif de lecture d'un article qui nécessite une structure de méta-données évoluées capables d'annoter des portions de cet article.

Les méta-données sont particulièrement importantes pour les ressources (images ou vidéos) sans lesquelles, elles peuvent demeurer pratiquement inexploitable et difficiles à retrouver. Plusieurs standards concernant les méta-données existent comme Dublin Core.

Nous allons les détailler dans ce qui suit :

6.1.2. Le Dublin Core :

Est un ensemble d'éléments simples mais efficaces permettant de décrire une grande variété de ressources. Dublin Core propose 15 éléments de description (Méta-données) relatifs au contenu d'une ressource. Ces champs peuvent être inclus dans les pages Web, dans la partie HEAD et dans la balise <méta>. Ils sont classifiés en trois catégories qui concernent [BENA05]:

- le contenu de la ressource : titre, sujet, description, date, source, langue relation, portée.
- la propriété intellectuelle : gestion des droits, collaborateur, éditeur, auteur / créateur.
- la matérialisation de la ressource : type de la ressource, format de la ressource, identificateur de la ressource.

Le Tableau 1.1 montre les noms de ces éléments, leurs identificateurs et leurs définitions.

Nom Titre	Identificateur Titre	Définition
Titre	Titre	Le nom donné à la ressource.
Auteur / Créateur	Créateur	L'entité principalement responsable de la création du contenu de la ressource.
Sujet et mot-clé	Sujet	Le sujet du contenu de la ressource.
Description	Description	Un compte rendu du contenu de la ressource.
Editeur	Editeur	Une entité ayant la responsabilité de rendre la ressource disponible.
Collaborateur	Collaborateur	Une entité ayant la responsabilité de collaborer au contenu de la ressource.
Date	Date	Une date associée à un événement dans le cycle de vie de la ressource.
Type de ressource	Type	La nature ou le type du contenu de la ressource.
Format	Format	Le caractère physique ou numérique de la ressource.
Identificateur de la ressource	Identificateur	Une référence non ambiguë de la ressource dans un contexte donné.
Source	Source	Une référence de la ressource présente. D'où est tirée la ressource
Langue	Langue	La langue du contenu intellectuel de la ressource.
Relation	Relation	Une référence à une ressource connexe.
Portée	Portée	L'étendue ou la portée du contenu de la ressource.
Gestion des droits	Droits	Des renseignements au sujet des droits détenus sur une partie de la ressource ou sur son ensemble.

Tableau 1.1- La liste des 15 éléments formant le Dublin Core [BENA05].

Voici dans la figure 1.4 ce que donnera l'inclusion de méta-données dans un document XHTML (c'est-à-dire un document HTML normalisé en XML) :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd
<html>
<head>
<title>Un document en HTML</title>
<meta http-equiv="Content-type" content="text/html; charset=iso-8859-1" />
<link rel="schema:DC" href=http://purl.org/dc/elements/1.1/ />
<meta name="DC.Title" lang="fr" content="Un document en HTML" />
<meta name="DC.Date.created" scheme="W3CDTF" content="2003-04-03" />
<meta name="DC.Date.modified" scheme="W3CDTF" content="2003-04-27" />
<meta name="DC.Subject" lang="fr" content="HTML, document, Dublin Core" />
<meta name="DC.Language" scheme="RFC3066" content="fr-FR" />
<meta name="DC.Description" lang="fr" content="Doc. html avec métadonnées" />
</head>
<body> ... </body>
</html>
```

Figure 1.4- Dublin Core dans un document XHTML [AYAR05].

Les balises *méta* permettent de spécifier le titre, la date de création, la ou les dates de modification, le sujet, la langue et la description d'un document XHTML. L'intérêt étant que les moteurs de recherche puissent avoir accès à ces méta-données, de façon à récupérer une sémantique des documents, et de ce fait permettre une meilleure indexation de ceux-ci.

Les méta-données permettent donc de faciliter le travail des moteurs d'indexation et de recherche en permettant de rendre accessibles des informations sur le document lui-même [AYAR05].

Cependant, en utilisant ces schémas, les auteurs des méta-données, s'ils sont à peu près d'accord sur la signification de l'élément « titre » ne le sont pas forcément concernant les éléments « sujet » ou « type » d'un document. Ce qui induit des ambiguïtés sur le sens des champs. En outre, la recherche en utilisation des méta-données même si elle est meilleure qu'une recherche classique (qui n'utilise pas des méta-données) n'est en fait suffisante que pour une utilisation humaine. L'exploitation de la structure du schéma par des logiciels est limitée puisqu'aucune sémantique interprétable par une machine n'est associée aux domaines de valeurs des différents champs et donc certaines ressources retrouvées pourraient ne pas convenir. Nous voyons bien donc que ces schémas des méta-données ne permettent de définir ni la sémantique opérationnelle des différents champs de description, ni celles de leurs domaines de valeurs. Ils peuvent être résumés en des standards descriptifs

exprimés sous forme de DTD ou des schémas XML. Pour évoluer donc vraiment vers un Web sémantique, la solution est d'utiliser des schémas des méta-données fondées sur des ontologies.

6.2. L'annotation

Plusieurs communautés de chercheurs (informatique, linguistique, rhétorique, etc.) ont étudié l'annotation, et en ont proposé plusieurs définitions.

Dans le cadre de notre travail on va s'intéresser aux études proposées par des informaticiens.

6.2.1. Définition

Selon la communauté des informaticiens plusieurs définitions sont accordées à l'annotation. Nous présentons ci-après les plus célèbres.

« Une annotation est une information graphique ou textuelle attachée à un document et le plus souvent placée dans ce document ». Cette place est donnée par une ancre [DESM02]. Prié dans [PRIE04] définit l'annotation comme « un commentaire libre situé à l'intérieur de la ressource documentaire ».

Selon le W3C « l'annotation est toute explication, scolie ou critique associée à une page web ». Alors une annotation seule ne fait pas sens, elle est toujours associée à l'objet qui a été annoté.

[BRIN04] (Cité dans [MKAD04]) définit l'annotation comme étant une « note particulière attachée à une cible. Qui peut être une collection de documents, un document, un segment de document (paragraphe, groupe de mots, mot, image ou partie d'image, etc.), une autre annotation. À une annotation correspond un contenu, matérialisé par une inscription, qui est une trace de la représentation mentale que l'annotateur se fait de la cible. Le contenu de l'annotation pourra être interprété à son tour par un autre lecteur. L'ancre est ce qui lie l'annotation à la cible (un trait, un passage entouré, etc.) ».

Cette définition, plus complète que les précédentes, innove en mettant en avant le lieu de l'annotation, appelé « cible ».

6.2.2. Eléments caractérisant une annotation : on peut citer :

§ Le document est la cible à laquelle l'annotation se réfère (voir figure 1.5). L'annotation peut porter sur un document dans sa globalité, sur une partie seulement, un paragraphe, ou sur un ensemble de documents. Elle peut être distincte physiquement ou non du document auquel elle se rapporte. Par exemple, une note de bas de page est intégrée au document alors qu'un post-it est distinct du document auquel il se réfère.

§ L'ancre est le point d'attache d'une annotation à l'objet annoté (voir figure 1.5).

L'annotation peut avoir un ou plusieurs auteurs, qui peuvent être ou non distincts du ou des auteurs du document. Par exemple, une note de bas de page peut être faite par l'auteur du document ou par un traducteur.

§ Le ou les consultants de l'annotation peuvent être visés : l'annotateur connaît au moment de la production de l'annotation le (ou les) consultant(s) à qui il veut transmettre un message. C'est le cas par exemple, d'un professeur qui annote le devoir de l'un de ses élèves. Mais les consultants de l'annotation peuvent aussi être inopinés. Dans ce cas l'annotateur ne connaît pas au moment de la rédaction de l'annotation les consultants potentiels de celle-ci. Par exemple, sur les sites internes de ventes de livres, le lecteur peut faire un commentaire à propos des ouvrages qu'il connaît. Ce commentaire n'est pas destiné à quelqu'un en particulier mais est consultable par l'ensemble des internautes qui visitent le site. Le consultant et l'auteur de l'annotation peuvent être la même personne, l'annotation n'a alors pas vocation à être partagée. C'est une annotation privée par opposition à une annotation publique.

Enfin, le consultant peut être un humain (on parle alors d'annotations cognitivement sémantiques), ou une machine. Dans ce dernier cas, les annotations s'adressent à des programmes informatiques et l'on parle d'annotations constitutionnellement sémantiques.

§ Le contenu de l'annotation correspond à l'information transmise (voir figure 1.5). Celui-ci peut être plus ou moins explicite (des notes de bas de page à certaines annotations personnelles) et plus ou moins formel (du langage codé, très formel, au langage naturel et sigles intuitifs informels).

§ Les formes des annotations (les aspects physiques que peuvent prendre les annotations sur le document) peuvent être variées : textuelle (icônes, symboles de liens, notes, mises en forme typographiques, redécoupages de texte), graphique, vocale, vidéo, etc.

§ La durée de vie des annotations peut être plus ou moins longue. Les annotations peuvent être provisoires comme dans le cas de celles sur un dossier pour le ranger, ou permanentes comme les notes de bas de page.

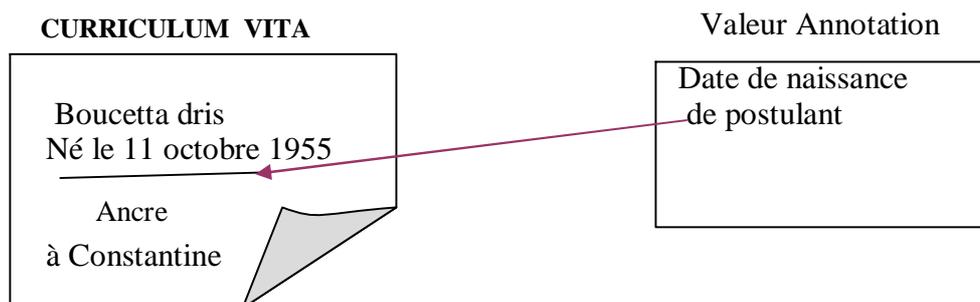


Figure 1.5- Exemple d'annotation.

6.2.3. Rôle des annotations

Les annotations ont des fonctions, au moment de leur production :

- aide à la lecture et à la compréhension du document au cours de la lecture d'un document : le lecteur annote celui-ci. Ces annotations complètent le document, l'illustrent ou correspondent à des réflexions fortuites du lecteur. Elles sont le reflet de l'attention du lecteur. Elles supportent la lecture du texte et permettent au lecteur de le comprendre et de s'en faire une représentation.
- mémoire externe pour retrouver des informations ou indiquer un emplacement : Le lecteur peut aussi marquer des emplacements dans le texte via des annotations de façon à retrouver certaines informations par la suite. Ces annotations sont une mémoire externe. Par exemple, les passages importants qui doivent être relus et mémorisés sont soulignés.
- opérationnalisation de l'information et restructuration du document : Certaines annotations permettent au lecteur de découper un document pour en réordonner les parties ou faire des corrélations entre celles-ci. Ainsi, un lecteur peut regrouper différents paragraphes en les encadrant, puis les renuméroter en fonction de la logique qui sous-tend sa tâche.
- forum et source de discussion : Les annotations peuvent aussi servir de support aux discussions, corrections et relectures d'un texte. En effet, les documents annotés constituent un contexte partagé qui facilite les échanges. Les annotations partagées qui ne nécessitent pas de droit d'accès en écriture peuvent supporter de riches communications à propos des pages Web [KOIV01]. Mais elles peuvent aussi permettre des échanges au sein de groupes de travail pour recommander la lecture d'un document [DENO00] ou exprimer un point de vue dans des tâches de correction.
- planification et organisation : Enfin, certaines annotations assurent la coordination des tâches au sein d'un groupe en planifiant et en organisant le travail de ses membres. Par exemple, les annotations peuvent être utilisées pour indiquer la répartition des tâches et leur durée.

6.2.4 La différence entre l'annotation et une méta-donnée

La distinction entre l'annotation et la méta-donnée du point de vue du processus de mise en place selon prie et al dans [PRIE04] est :

- Une méta-donnée sera plutôt attachée à une ressource identifiée en tant que telle sur le Web et aura plutôt une pertinence a priori et sera plutôt saisie suivant un schéma par exemple, la description normalisée d'un service Web, l'auteur d'un document.

- Une annotation sera plus située au sein de cette ressource et écrite au cours d'un processus d'annotation / lecture, par exemple : un commentaire libre associé à un fragment d'une page Web (quelques mots).

On voit clairement qu'il s'agit simplement de mettre l'accent sur un caractère plus situé au sein de la ressource (du fait de son exploitation par un utilisateur), l'annotation par rapport à une méta-donnée plus indépendante, voire ressource en tant que telle.

6.3. L'annotation sémantique :

Dans le cadre du Web, il existe un autre type particulier d'annotation appelée « annotations sémantiques » c'est à dire représenter le contenu du document par une description formelle. L'annotation sémantique est utilisée dans la recherche d'information et la classification des documents. Faisant référence à une connaissance (habituellement une ontologie) séparée du document, ces annotations sont destinées à être traitées par des machines.

6.3.1. Définition

Plusieurs significations ont été attribuées à la définition d'une annotation sémantique. Les plus complètes accordent les interprétations suivantes :

- « L'annotation sémantique est un processus d'instanciation des données à partir d'une ontologie du domaine » [MEST07].
- « L'annotation sémantique est définie comme une représentation formelle d'un contenu, exprimée à l'aide de concepts et relations et instances, décrits dans une ontologie, et reliée à la ressource documentaire source » [AMAR07].

En termes de documentation, les annotations sémantiques décrivent le lien entre les entités se trouvant dans le document et leurs descriptions sémantiques représentées dans l'ontologie. Elles permettent ainsi de désambiguïser le contenu du document pour un traitement automatique (ex. recherche documentaire, ...).

L'annotation sémantique à partir d'ontologie est devenue actuellement l'approche la plus prometteuse pour partager et exploiter l'information sur le Web.

La figure 1.6 montre des annotations sémantiques, où chaque contenu du texte est lié à des instances de concepts ou de propriétés décrites dans l'ontologie de référence. Le fait d'utiliser la sémantique de ces concepts et relations telles que modélisées dans l'ontologie pour créer l'annotation, permet alors d'assigner cette sémantique au contenu de la ressource et donc de le rendre exploitable par des agents logiciels. Dans notre travail on a utilisé ce type d'annotations. On a annoté les CV/offres d'emploi avec des instances de l'ontologie-ER puisque l'utilisation des annotations formelles va permettre l'automatisation du processus d'appariement entre les CVs et les offres d'emploi.

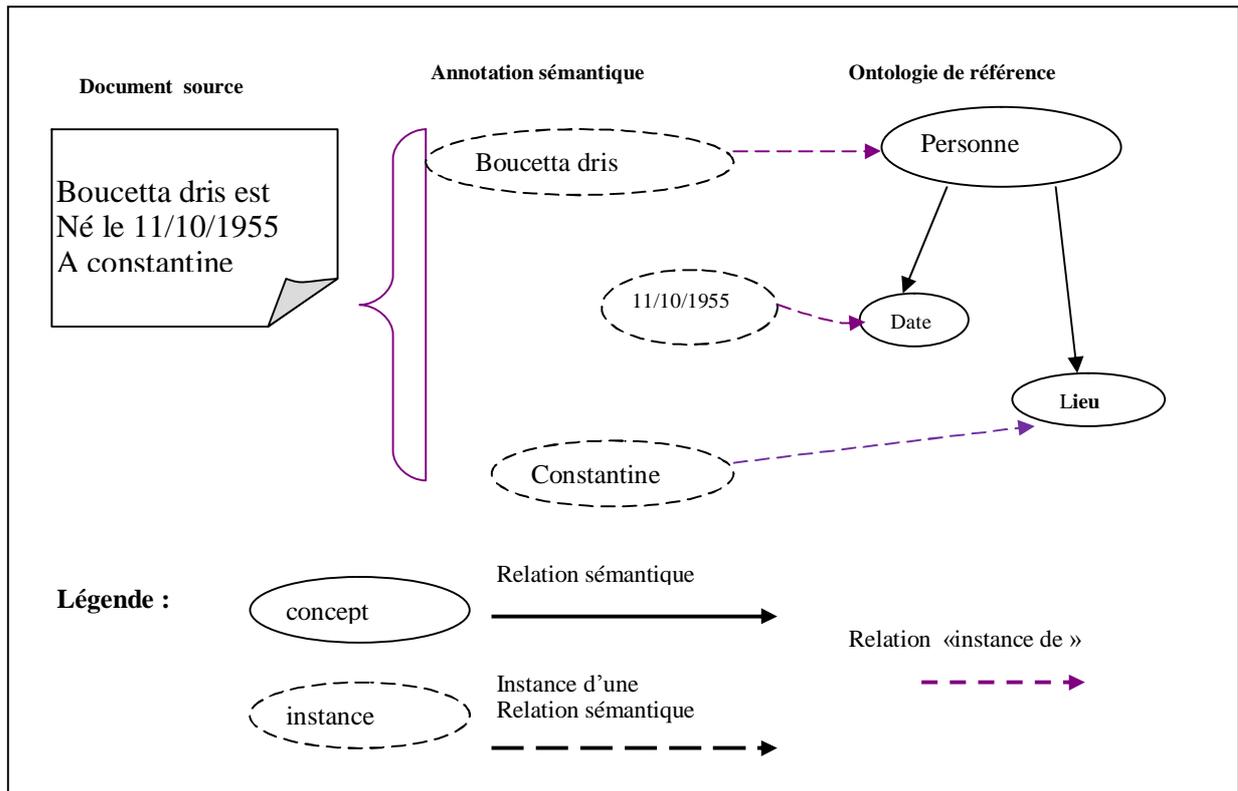


Figure 1.6- Exemple d'une annotation sémantique accordée par une ontologie de référence.

6.3.2. Le stockage des annotations

Les annotations peuvent être soit « embarquées » soit « débarquées » vis-à-vis de la ressource documentaire source [HABE05]:

§ **Une annotation est dite « embarquée »** : lorsqu'elle est ajoutée directement au contenu du document d'origine. Ce dernier pourra être mis à jour sur le site Web d'origine, dans le système de gestion de contenu documentaire de l'entreprise, ou simplement enregistré localement sur un ordinateur pour une réutilisation ultérieure [AMAR07].

§ **Une annotation est dite « débarquée »** : lorsqu'elle est stockée à l'extérieur du document source. Non seulement l'annotation elle-même doit être stockée, mais le lien avec la ressource annotée doit aussi être préservé. C'est ce qui permettra ensuite de retrouver toutes les annotations correspondantes à une ressource. Généralement, ces annotations sont stockées sur des serveurs d'annotations qui peuvent être interrogés afin de retrouver les annotations d'une ressource donnée.

L'inconvénient de l'annotation débarquée réside dans le fait que si le document source est modifié ou supprimé, les annotations deviennent obsolètes voire orphelines [AMAR07].

6.3.3. Représentation des annotations et la tâche d'annotation

Les premiers outils d'annotation comme SHOE KA, OntoAnnotate utilisent des langages non standardisés, respectivement SHOE, HTML-A. Puis avec l'arrivée de RDF et de DAML+OIL, ces précédents formats sont plus ou moins tombés en désuétude : les anciens outils ont migré leurs applications vers la prise en compte de RDF et/ou DAML+OIL alors que les nouveaux outils ont bien sûr directement implémenté l'utilisation de ces langages pour leurs annotations. Enfin, on constate récemment l'arrivée d'annotations générées en OWL. Mais RDF reste pourtant un standard en matière d'annotations documentaires.

Une annotation regroupe essentiellement trois éléments principaux [SIDH05], à savoir :

- L'annotateur, la personne qui réalise l'annotation.
- Le document source concerné par l'annotation (le document de la figure 1.7).
- Les objets d'annotation introduits sur le document (la figure 1.8 si on utilise comme langage d'annotation RDF et la figure 1.9 si on utilise OWL DL pour annoter).

La formalisation de pages web se définit comme la tâche d'annotation.

Une tâche d'annotation selon [TOUS07] consiste à annoter avec des méta-données une page Web ou son contenu, une tâche d'annotation sémantique c'est lorsque les méta-données utilisées pour l'annotation sont des instances de concepts et de relations entre les concepts décrits formellement et structurés dans une ontologie. Amardeilh dans [AMAR07] conserve la même définition et ajoute la notion de relier la notation à la page web annotée.

De ce fait la tâche d'annotation consiste à :

- décrire le contenu de la page en utilisant les ontologies.
- relier ce contenu à la page web.

Pour bien expliquer l'activité d'annotation, on a préféré utiliser un exemple. On utilise l'ontologie « famille » pour annoter l'extrait ci-dessous (figure 1.7). Cette ontologie contiendrait le concept « Personne » des attributs comme « lieu de naissance », « date de naissance » etc. Des relations telles que « MariéAvec (Personne, Personne) ». Le document serait alors annoté en fonction de ces éléments, à savoir :

- Par les instances de concepts. Par exemple, une annotation sera créée avec « Boucetta_Dris » qui représente une instance du concept « Personne ».
- Par les valeurs d'attributs. Par exemple, « 1 octobre 1955 » peut être la valeur de l'attribut « date de naissance » qui sera attachée à l'instance « Boucetta_Dris » de notre annotation.
- Par les instances de relations. Par exemple, l'instance « Boucetta_Dris » du concept Personne et l'instance « Zitouni_Halima » du concept Personne peuvent être connectées par la relation « MariéAvec ». Le document sera annoté avec cette instance de relation « MariéAvec (Boucetta_Dris, Zitouni_Halima) ».

La figure 1.8 présente le fichier d'annotation simulé en langage RDF du texte ci-dessous en utilisant « l'ontologie famille ». Ces annotations correspondent aux mots écrits en gris de la figure 1.7.

La famille de dris

Boucetta dris né le **11 octobre1955** à **Constantine**, en Algérie. Il est père de cinq enfants, zouhel, azhar , soumia , mohtaz, rawnak. **Marié avec Zitouni halima**, le 5 juillet 1978. Zitouni Halima est née le **20 janvier 1956** à **Constantine**.

Figure 1.7- Une page web www.famillededris.html.

```
<xml "version="1.0" encoding="UTF-8">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  Xmlns:famille="http://www.laboratoirelire.com/onto#famille">
  < rdf:Description rdf:about="http:// www.famillededris.htm">
  <dc:subject rdf:resource="#boucetta_ dris"/>
  </rdf:Description>
  <rdf:Description rdf:about="Boucetta_Dris">
  < Famille:date_naissance>11 octobre1955 </people:date_naissance>
  < Famille:lieu_naissance rdf:resource="#Constantine/>
  < Famille : MariéAvec>
  <rdf:Description rdf:about="Zitouni_Halima">
  < Famille:date_naissance>20 janvier 1956 </people:date_naissance>
  < Famille:lieu_naissance rdf:resource="#Constantine/>
  </rdf:Description>
  </ Famille : MariéAvec>
  </rdf:Description>
  </rdf:RDF>
</xml>
```

Figure 1.8- Exemple d'annotation sémantique en RDF.

```

<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#">
  <owl:Ontology rdf:about=" http://www.laboratoirelire.com/onto#famille "/>
  <owl:Class rdf:ID="personne"/>
  <owl:ObjectProperty rdf:ID="MariéAvec">
    <rdfs:range rdf:resource="#personne"/>
    <rdfs:domain rdf:resource="#personne"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="lieudenaissance">
    <rdfs:domain rdf:resource="#personne"/>
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="datenaissance">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#personne"/>
  </owl:DatatypeProperty>
  <personne rdf:ID="Boucetta_Dris">
    <datenaissance xml:lang="fr">11 octobre 1955</datenaissance>
    <lieudenaissance xml:lang="fr">Constantine</lieudenaissance>
    <MariéAvec>
      <personne rdf:ID="Zitouni_Halima">
        <lieudenaissance xml:lang="fr">Constantine</lieudenaissance>
        <datenaissance xml:lang="fr">20 janvier 1956</datenaissance>
      </Personne>
    </MariéAvec>
  </personne>
  < rdf:Description rdf:about="http:// www.famillededris.htm">
  <dc:subject rdf:resource="#boucetta_dris"/>
  </rdf:Description>
</rdf:RDF>

```

Figure 1.9- Exemple d'annotation sémantique en OWL.

6.3.4. Approche d'annotations

Nous pouvons citer plusieurs processus d'annotation des ressources web, selon le degré de leur automatisation. Les trois approches principales sont les suivantes : Manuelle, semi automatique et automatique [MESTI07].

Ø **Approche manuelle** [KAHA01], [HAND01] l'annotation manuelle est entièrement effectuée par un annotateur humain qui place les annotations de son choix dans un document existant en phase de rédaction.

Alors l'annotation manuelle consiste simplement à mettre en place une interface utilisateur dans laquelle l'utilisateur humain peut sélectionner la ressource à annoter, choisir le modèle formel servant à la création des annotations sémantiques, tout en respectant les contraintes imposées par le modèle formel, créer les annotations voulues sur la ressource sélectionnée.

Ø **Approche semi automatique** l'annotation semi-automatique s'appuie généralement sur un moteur d'extraction d'informations qui est intégré dans un outil d'annotation. Il suggère des annotations à l'utilisateur qui doit ensuite les valider manuellement.

Ø **Approche automatique** l'annotation automatisée est entièrement réalisée par un outil d'extraction d'informations qui est intégré dans l'outil d'annotation. Le traitement automatique consiste à [TENI07] :

1. Identifier automatiquement, dans une page web, les éléments qui sont pertinents.
2. Déterminer quels sont les concepts de l'ontologie les plus spécifiques possibles, pour annoter chacun de ces éléments.

L'automatisation repose sur un apprentissage à partir d'un corpus constitué d'éléments marqués par un expert. Le marquage associe à chaque concept de l'ontologie des éléments de la page en rapport avec ce concept. L'apprentissage génère un WRAPPER (système d'extraction d'informations) capable d'annoter des éléments du document sous la forme d'instances de concepts et de rôles de l'ontologie fournie. Des mécanismes de raisonnement exploitant l'ontologie sont utilisés pour déterminer le concept le plus spécifique avec lequel un élément doit être annoté.

Pour conclure, l'approche d'annotation adoptée dans notre travail est l'annotation manuelle vu l'outil utilisé pour l'annotation. Nous avons deux types de ressources à annoter : les CVs, les offres d'emplois.

6.3.5. Rôle des annotations sémantiques :

Les annotations sémantiques sont utilisées pour deux grandes tâches. La première est la recherche d'informations, puisque toute annotation informatique liée à une ressource représente de ce fait un index pour cette ressource. La seconde est la composition de documents, de fragments de documents, de services en vue de construire de nouveaux documents (cours, catalogues, réponses à une question, etc.) [PRIE04].

Les annotations sémantiques sont applicables à n'importe quel type de contenu : pages web, documents textuels non structurés, champs d'une base de données, documents audio ou

vidéo, etc. Enfin, plus le modèle de l'annotation est formalisé, plus les services proposés à partir de cette annotation peuvent devenir « intelligents ».

En effet, les agents logiciels pourront inférer de nouvelles connaissances, raisonner sur ces connaissances et ainsi améliorer les résultats de la recherche d'information ou bien dégager un sens implicite contenu dans le document d'origine [AMAR07].

6.3.6. Framework pour les annotations sémantiques

Annotea et CREAM sont deux cadres de travail (frameworks) pour l'annotation sémantique. Ils sont implémentés d'une manière différente dans plusieurs outils d'annotation sémantique [LUON07].

§ **Annotea** [KAHA01] [KOIV2005] est un projet du W3C qui spécifie l'infrastructure pour l'annotation des documents du Web. Le format principal utilisé dans l'Annotea est RDF et les types de documents qui peuvent être annotés sont les documents en HTML ou basés sur XML. Annotea fournit dans XPointer une méthode permettant de localiser des annotations dans un document. XPointer est une recommandation du W3C pour identifier des fragments des ressources avec leur URI. L'approche d'Annotea se concentre sur un modèle semi-formel d'annotation dans lequel les annotations contiennent des déclarations qui ont besoin d'être ajoutées à l'aide de méta-données (i.e. l'auteur, la date...).

§ **Le framework CREAM** [HAND03] (voir figure 1.10), comme Annotea, souscrit également aux formats standards du W3C avec des annotations représentées en RDF ou OWL et au XPointer. Ces annotations sont utilisées pour annoter des ressources et des documents dont les formats sont en XML et HTML. Il spécifie les composants exigés pour un système d'annotation qui doit supporter la création semi-automatique des annotations basées sur l'ontologie.

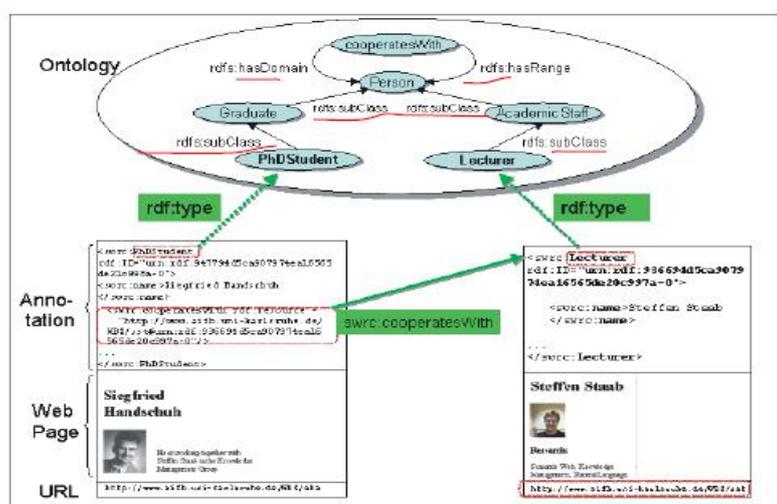


Figure 1.10- Exemple d'annotation avec CREAM.

6.3.7. Outils pour l'annotation sémantique

Basés sur Annotea et CREAM plusieurs outils ont été proposés afin de créer et manipuler les annotations. Ces outils peuvent supporter l'annotation automatique, semi-automatique et manuelle.

6.3.7.1 Qu'est ce qu'un outil d'annotation ?

Un outil d'annotation sémantique est un outil logiciel qui permet d'insérer et de gérer des annotations sémantiques liées à au moins une ressource documentaire donnée. Dans le cadre du Web Sémantique, les outils d'annotation sémantique utilisent une ontologie, ou tout au moins un modèle formel, qui formalise et structure les annotations produites en fonction des concepts et des contraintes définis dans cette ontologie [AMAR07].

Ces outils d'annotation sémantique ont pour objectif d'alléger le fardeau de l'annotation manuelle dans les pages Web. La plupart d'entre eux ont évolué vers des environnements de plus en plus automatisés grâce aux méthodes issues des domaines de l'Extraction d'Information et des Systèmes d'Apprentissage [CORC06]. En plus de l'interface traditionnelle d'annotation manuelle, ces méthodes sont capables de suggérer à l'utilisateur un ensemble d'annotations sémantiques relatives à la ressource documentaire analysée [AMAR07].

6.3.7.2 Quels sont les outils d'annotation les plus connus ?

Nous présentons dans cette section quelques outils, parmi les plus connus, qui nous permettent d'élaborer des annotations pour des ressources et des documents texte en se référant à une ontologie.

§ **SMORE** [KALY03] Le groupe de recherche Mindswap¹ de l'Université de Maryland, a fait évoluer SHOE KA² vers une implémentation des ontologies et des annotations générées en OWL et RDF respectivement donnant naissance à l'outil SMORE [KALY03]. Celui-ci conserve néanmoins les caractéristiques de SHOE KA, si ce n'est que les interfaces utilisateurs sont grandement améliorées. En plus d'annoter sémantiquement des pages Web existantes, SMORE permet également de combiner l'annotation et la création de nouvelles pages HTML grâce à son éditeur HTML incorporé. Il intègre aussi l'éditeur d'ontologie SWOOP³, également conçu par Mindswap, qui permet de naviguer dans une ontologie et de sélectionner les classes, relations ou attributs afin de créer les triplets qui seront ajoutés aux pages HTML traitées. Enfin, SMORE⁴ permet de vérifier les propriétés de domaine et de portée pour détecter les triplets invalides et en alerter l'utilisateur humain pour correction.

§ **OntoAnnotate**⁵ [STAA01] [STAA01a] est un outil d'annotation semi-automatique qui permet de rassembler les connaissances des documents et des pages Web aux formats

HTML et XML, de créer des annotations sémantiques de document et d'enrichir la base d'annotations avec des méta-données. Cet outil basé sur l'approche CREAM [HAND01] utilise la source de données fondamentales et le moteur d'inférence de l'outil OntoBroke⁶ pour son processus d'annotation. Par défaut, Annotea utilise les propriétés du DublinCore, telles que l'auteur, la date, le titre, l'éditeur, etc., pour créer des méta-données sur les documents traités. L'utilisateur peut également fournir une ontologie de domaine modélisée en RDFS. Les annotations créées sont conservées de façon externe au document annoté. Ces annotations sont alors disponibles soit localement sur l'ordinateur de l'utilisateur, soit sur un serveur d'annotation RDF public.

§ **OntoMat-Annotizer**⁷ est un outil d'annotation de page Web qui est intuitif et interactif pour l'utilisateur. Cet outil est aussi basé sur l'approche CREAM. Il aide aux tâches de création et de maintenance des méta-données des pages Web. Au lieu d'annoter manuellement la page avec un éditeur de texte, OntoMat-Annotizer permet à l'annotateur de souligner les parties appropriées de la page Web et de créer de nouvelles instances par les opérations *drag-and-drop*. Cet outil est une application Java, il a une architecture modulaire avec des *plugins* qui sont dédiés au navigateur d'ontologie, au navigateur de web, etc.

6.3.7.3 Synthèse

Beaucoup d'outils d'annotation sémantique ont été conçus ces dernières années. Mais malheureusement certains sont payants et le reste sont des outils de laboratoire pleins de bug et sont généralement dans la phase de test telles que ONTOMAT et SMORE. ONTOMAT est un outil facile à utiliser (voir figure 1.11). Son problème est la non génération du fichier résultat de l'annotation après le processus d'annotation, vu un bug dans la fonctionnalité qui génère ce dernier.

¹<http://www.mindswap.org/>

²<http://www.cs.umd.edu/projects/plus/SHOE/KnowledgeAnnotator.html> ³ <http://www.mindswap.org/2004/SWOOP/>

⁴ <http://www.mindswap.org/2005/SMORE/> ⁵ http://www.ontoprise.de/content/index_eng.html

⁶ <http://ontobroker.semanticweb.org>

⁷ <http://annotation.semanticweb.org/ontomat.html>

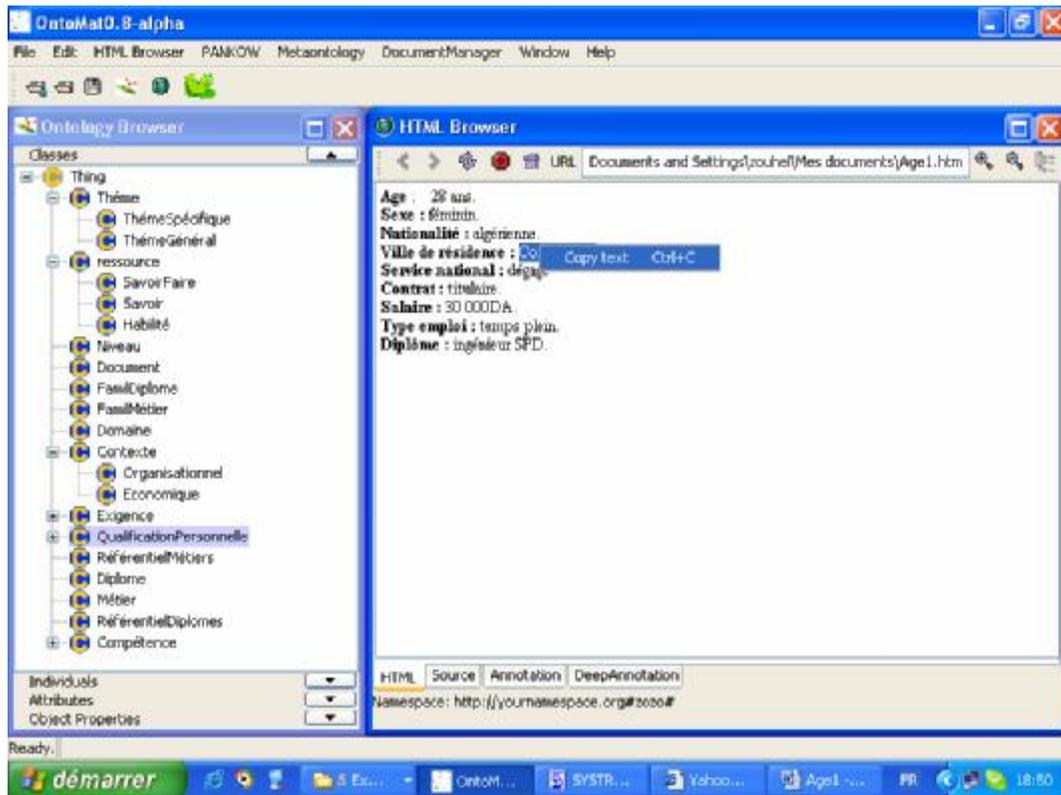


Figure 1.11 Annotation avec ONTOMAT.

SMORE (voir figure1.12) est un outil dédié aux ingénieurs ontologiques. Pour l'utiliser, il faut avoir des notions préalables sur les ontologies RDF (prédicat, sujet, etc.) pour créer des triplets RDF qui forment une annotation, ainsi sur OWL (concept, rôle, etc.) pour naviguer dans l'ontologie. Ce dernier après le processus d'annotation génère un fichier illisible et inexploitable.

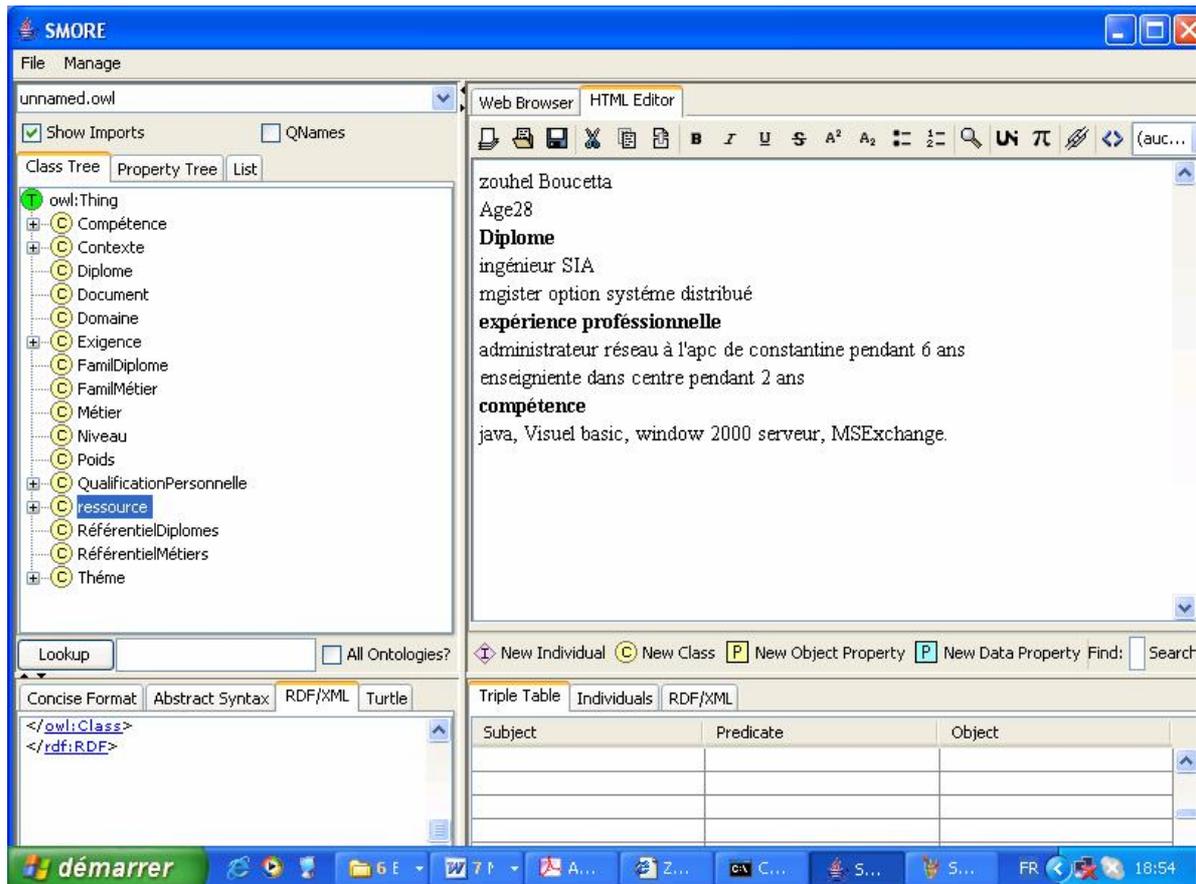


Figure 1.12 Annotation avec SMORE.

Vu ces problèmes, un outil d'annotation « FETCH » est développé dans notre équipe par Allioua soufiane dans [ALLI08], mais ce dernier est dédié aux images. Nous l'avons adapté pour l'annotation des fichiers HTML. Le résultat de l'annotation de ce dernier est un fichier OWL, à l'avenir nous pourrions ajouter une fonctionnalité pour avoir comme résultat de l'annotation un fichier RDF. Le type du fichier d'annotation (RDF ou OWL) n'est pas le problème, l'essentiel est qu'il ne soit pas ambigu et qu'il soit utilisable par un programme.

Exemple d'utilisation de l'outil « FETCH »: cet exemple est basé sur l'utilisation de l'ontologie-ER sous OWL et un CV. Cette ontologie permet aux chercheurs d'emplois et aux recruteurs, l'accès sémantique à leurs CVs ou offres d'emploi respectivement. Dans l'interface du système, la partie gauche s'affiche le document à annoter par l'annotateur et dans la partie droite les structures ontologiques (classes, attributs et relations) à utiliser. L'annotation consiste à choisir un segment du document (le sélectionner) puis voir ce qui lui correspond dans l'ontologie (instance/propriété/classe). A chaque instance créée les attributs littéraux ainsi que les relations (rôles) peuvent être renseignés. Comme on peut utiliser les instances de l'ontologie (instance de classe ou de relation) pour enrichir un CV

ou offre d'emploi avec des annotations. La figure 1.13 illustre cet exemple d'annotation sémantique.

Pour la sauvegarde des annotations l'outil d'annotation génère un document OWL qui renferme les annotations effectuées par l'utilisateur. Ce document est conservé dans l'entrepôt des annotations.

The screenshot shows the 'FETCH annotation' application window. The main content area displays a resume titled 'CURRICULUM VITAE' for Halim Zitouni. The resume includes personal information (Nom, Prénom, Adresse, Téléphone, E-mail), age (30 ans), sex (masculin), nationality (algérien), marital status (marié), and residence (Skikda). It also lists employment details (Déplacé: non, Contrat: Titulaire, Salaire: 50000 DA, Rotation: non, Type emploi: Temps plein) and a diploma (Licence Professionnelle SI).

On the right side, there is a hierarchical tree view of the semantic ontology. The root is 'owl:Thing', which branches into several categories: 'Compétence', 'Contexte', 'Diplome', 'Document', 'Domaine', 'Exigence', 'QualificationPersonnelle', 'ressource', and 'Thème'. Each category contains specific sub-entities, such as 'Contrat', 'Déplacement', 'Rotation', 'Salaire', 'TypeEmploi', 'Famildiplome', 'Familmétier', 'Métier', 'Niveau', 'Age', 'Nationalité', 'PermisConduire', 'ServiceMilitaire', 'Sexe', 'SituationFamill', 'VileRésidence', 'Habilité', 'Savoir', 'SavoirFaire', 'RéférentielDiplomes', and 'RéférentielMétiers'. The 'Thème' category is further divided into 'ThèmeGénéral' and 'ThèmeSpécifique'.

Figure.1.13 Annotation avec FETCH.

7. Conclusion

Nous avons détaillé dans ce chapitre la notion d'ontologie et d'annotation. Nous nous sommes intéressées principalement à l'annotation sémantique et nous avons vu que dans le cadre du Web Sémantique, elle est fondamentalement liée à la modélisation d'une ontologie. En effet, cette ontologie va représenter les concepts, attributs et relations d'un domaine à l'aide d'un langage de représentation des connaissances orientées Web comme OWL. Les annotations sémantiques sont structurées à l'aide de cette ontologie et leurs valeurs pointent vers les instances de concepts ou de relations de l'ontologie de référence. Les annotations sémantiques sont éditées dans un langage formel de représentation de connaissance, pouvant être stockées dans la ressource annotée ou/et dans un serveur des annotations selon l'outil d'annotation utilisé.

Nous avons vu qu'il existe une gamme assez importante d'outils d'annotations sémantiques ayant chacun des caractéristiques propres mais dont le but est de plus en plus orienté vers l'assistance des annotateurs humains à la création des annotations.

Nous avons vu que l'avantage majeur de l'annotation sémantique est qu'elle permet un traitement automatique du contenu des ressources web par des agents logiciels, ce qui permet de fournir à l'utilisateur des résultats pertinents et intelligents.

De ce fait, nous allons utiliser l'annotation sémantique pour annoter les CVs/offres d'emplois avec des éléments d'une ontologie-ER son domaine est « L'informatique et la Télécommunication » pour automatiser le rapprochement de ces documents.

Chapitre 2

Gestion des documents dans le cadre du e-recrutement

Sommaire

1. Introduction
 2. Les sites d'emplois
 3. CVs /offres d'emploi et outils de génération
 - 3.1. Rubrique d'un CV
 - 3.2. L'offre d'emploi
 - 3.3. Outils de génération des CVs /offres d'emploi
 4. Le e-recrutement
 - 4.1. Les phases du processus de recrutement
 - 4.2. Les limite du e-recrutement
 5. La gestion des connaissances et des compétences
 - 5.1. Gestion des connaissances (KM)
 - 5.1.1 Définition
 - 5.1.2 Objectif de la gestion des connaissances
 - 5.2. C'est quoi une Compétence.
 - 5.3. Gestion des compétences
 6. L'appariement sémantique pour le e-recrutement
 - 6.1. L'appariement syntaxique des documents
 - 6.2. L'appariement sémantique des documents
 7. Les travaux existants
 - 7.1. Une approche basée sur l'annotation pour modéliser le contenu sémantique
 - 7.2. Un processus de recrutement basé ontologie
 - 7.3. BRIDGE
 - 7.4. Evaluation des travaux existants
 8. Conclusion
-

«*J'écoute et j'oublie, je vois et je me souviens, je fais et je comprends*»

Proverbe chinois

1. Introduction

L'Internet s'est transformé rapidement en un outil de recrutement performant. 53% des recrutements en Allemagne sont le résultat d'une postulation en ligne [BIZE05]. Un chiffre qui est susceptible d'augmenter. Cependant, vu les limites des outils dédiés à la gestion et à la recherche des Curriculum Vitae (CVs)/offres d'emplois, l'écoulement de l'information sur ce marché de travail est loin d'être optimal. Dans ce qui suit, nous allons décrire les caractéristiques des CVs et des offres d'emplois, leurs outils de génération, puis nous allons décrire le processus de recrutement dans le web actuel et ses limites. La solution, pour la majorité de ces problèmes, réside dans la prise en compte du contenu sémantique de ces documents afin d'automatiser leur rapprochement. Pour, cela l'approche annotation du Web sémantique est utilisée par des travaux qui visent l'automatisation du e-recrutement, il seront détaillés dans la section 7.

2. Les sites d'emplois

Les portails de travail diffèrent selon la gamme des services qu'ils fournissent. Il existe par exemple, des sites Internet qui se spécialisent dans l'emploi provisoire, des agences de recrutement qui offrent des emplois dans des secteurs spécifiques comme « l'informatique et la télécommunication, la finance... », des sites d'emplois spécifiques pour les jeunes diplômés AFIJ¹, d'autres généraux comme *Jobpilot*².

Le tableau 2.1 [MIFE06] présente une synthèse des sites d'emplois les plus connus. Ces sites sont classés selon les services qu'ils présentent. Par la suite nous détaillons quelques uns.

- ▶ **Jobpilot** : Fondée en 1995, Jobpilot est une entreprise leader du recrutement en ligne, présente dans 15 pays européens. Jobpilot fournit à ses clients des solutions de recrutement rapides, économiques et efficaces, tout en assurant un service haut de gamme. L'offre de service Jobpilot comprend de nombreux produits novateurs, en complément de la publication d'offres et de demandes d'emplois. La publication des CVs est gratuite pour les demandeurs d'emplois tandis que les employeurs payent les espaces, d'annonce.

1 : (<http://www.afij.org/>, d'autre),

2 : (http://http://www.jobpilot.com/content/service/pr_ir/news/Adecco-jobpilot.html).

Ce portail www.Jobpilot.com fournit :

- § Un marché d'emploi consacré à la recherche et à l'enseignement.
- § Une recherche à texte intégral de toutes les offres d'emploi.
- § Une vue d'ensemble des sociétés et de leurs profils.
- § Une aide gratuite pour écrire un CV.
- § Une rubrique d'emploi est consacrée aux étudiants et aux diplômés.
- § Les offres sont rangées selon le niveau, l'expérience professionnelle.
- § Les mises à jour hebdomadaires d'information du travail, sont fournies aux adhérents par des messages E-mail.

La recherche en ligne dans ce portail implique deux étapes : une recherche générale suivie d'une recherche détaillée. Dans la recherche générale l'utilisateur définit: type d'emploi, région globale, la langue. Cette information sera employée comme base pour une recherche détaillée dans laquelle les particularités du travail désiré peuvent être définies telles que (la précision détaillée de l'emploi cherché, le secteur industriel, la précision de la région...). Une liste de résultats appropriés est donnée à la fin de la recherche.

► **StepStone** (<http://www.lesinfos.com/news61730.html>)

StepStone est un site de jobs où se rencontrent les recruteurs et les travailleurs, attirant 1,9 million de visiteurs par semaine. Les candidats peuvent littéralement y construire leur carrière : chercher un job, postuler et mettre leur CV en ligne dans la base des CVs . Les patrons peuvent quant à eux y recruter du personnel : en publiant leurs offres d'emploi, et rechercher des candidats idéaux dans la base de CV de StepStone.

StepStone travaille avec les petites et les grandes entreprises, les bureaux de sélection.

► **Jobgate** [JOBG07]: (<http://www.indicerh.net/actu/article.php?sid=318>)

Jobgate.fr est créé en 2002. Il se distingue en offrant une ergonomie révolutionnaire qui permet une accessibilité aisée aux informations.

Il est présent sur tous les secteurs d'activité avec plus de 70 000 CV en ligne, 320 000 visiteurs par mois et 2 millions de pages vues par mois, avec une augmentation régulière du trafic. Jobgate.fr est composé d'un site généraliste Ejob.fr (www.ejob.fr), de sites d'emplois spécialisés par domaines ou catégories socioprofessionnelles (Seaad.com, Batijob.com, Medicajob.com, ...). Il offre ainsi une visibilité optimale des annonces

d'emplois auprès des candidats en multi-diffusant ces annonces sur les sites partenaires du réseau auxquels les candidats s'identifient. D'autre part, Jobgate.fr a une politique référencement qui permet de promouvoir également les annonces sur les moteurs de recherches francophones (Google, Yahoo, Msn, Voila ...).

Sites Institutionnels	<p>§ AFIJ Le site de l'insertion professionnelle de jeunes diplômés. http://www.afij.org/</p> <p>§ ANPE Le site de L'ANPE : plus de 100 000 offres d'emplois. http://www.anpe.fr/</p> <p>§ APEC Efficace et très riche en études, le très célèbre site fonctionne très bien avec plus de 9 000 offres ! http://www.apec.asso.fr/</p>
Sites en Régions	<p>§ Regionsjob Site portail de l'emploi dans les régions avec 6 sites sur le territoire Français Ouestjob, Nordjob, Estjob, Rhonealpesjob, Pacajob, Sudouestjob. http://www.regionsjob.com</p> <p>§ MECABOURG Les Industriels de la Métallurgie Mécanique et ... 1er pôle français de carrosserie industrielle, construction de matériel de transport et savoir-faire associés. Plus de 30 entreprises – 3 000 salariés. http://www.mecabourg.com</p> <p>§ Rhonalpemploi annonces d'emploi Lyon Rhone Alpes, offre d'emploi CDD Lyon ...Le site emploi du groupe Progrès / Dauphiné Libéré ... De nombreuses offres d'emploi en CDD à Lyon avec Le Progrès et Le Dauphiné Libéré. ... www.rhonalpemploi.com</p>
Sites Spécialisés	<p>§ Emploi informatique - lesjeudis.com Lesjeudis.com : emploi informatique - Retrouvez nos offres d'emploi informatique, déposez votre CV, alerte email pour recevoir les offres, assistez aux ... http://www.lesjeudis.com</p> <p>§ FraGGo.com Transport, Transports, logistique : offres d'emploi ... FraGGo.com moteur de recherche Transport Logistique, Site emploi transport et logistique, guide export. http://www.fraggo.com</p> <p>§ BTP Site de consultation d'offres d'emploi dans le BTP et les carrières publiques.</p> <p>§ http://www.emploi.lemoniteur-expert.com</p>
Sites Travail Temporaire	<p>§ Interim Adecco Adecco, n°1 du travail temporaire et premier réseau d'agences d'interim spécialisées, vous propose également des prestations de recrutement pour des postes. www.adecco.fr/</p> <p>§ SYNERGIE travail temporaire Réseau international d'agences de travail temporaire. Dépôt de CV. www.synergie.fr</p>

<p>Sites Généralistes</p>	<p>§ Cadremploi Le site d'emploi du groupe Le Figaro. Un nouvel espace des jeunes diplômés vient d'ouvrir ses pages! http://www.cadremploi.fr</p> <p>§ Cadresonline Toutes les offres de la presse - Le Monde, l'Express, LSA, l'Usine Nouvelle... – http://www.cadresonline.com/</p> <p>§ EmailJob La réputation d'EmailJob n'est plus à faire, Il fait partie des sites emploi les plus performants. http://www.emailjob.com/</p> <p>§ Jobpilot Jobpilot est un des leaders européens de l'emploi sur Internet.. http://www.jobpilot.fr/</p> <p>§ Monster Offres d'emplois, dépôt de CV, conseils, le petit frère français d'un des plus gros sites US. http://www.monster.fr/</p>
<p>Site Emploi Fonction Publique Territoriale</p>	<p>§ Centres de Gestion de la FONCTION PUBLIQUE TERRITORIALE Site National des Centres de Gestion de la FONCTION PUBLIQUE TERRITORIALE réalisée par la Fédération Nationale de Centres de Gestion (FNCDG) www.fncdg.com</p> <p>§ Centre de gestion de l'ain Introduction Présentation, organes, concours, base documentaire, foire aux questions. www.cdg01.fr</p>
<p>Sites Emploi Social</p>	<p>§ Actualités sociales hebdomadaires (ASH) Flashes, interviews et informations pratiques, offres d'emplois de l'hebdomadaire de l'action sociale en ligne. http://www.ash.tm.fr</p> <p>§ CREAMI- Rhones alpes, carrefour des acteurs publics et privés CREAI Rhône-Alpes : carrefour des acteurs publics et privés / conseil / étude / formation / dans le champ de l'action sociale. www.creami-ra.org</p>

Tableau 2.1- Synthèse des sites d'emploi.

3. CVs /offres d'emploi et outils de génération

Les CVs/offres d'emplois sont des documents structurés, qui constituent la base du processus de recrutement. Les portails d'emplois disposent d'outils pour les générer sous format numérique.

3.1. Rubrique d'un CV :

Le CV est un document structuré. Les informations qu'il comporte sont groupées de façon thématique au sein de rubriques qui seront matérialisées par un titre.

Selon Europass [EURO03] les rubriques les plus importantes sont :

- § **Informations personnelles** : il s'agit de la première rubrique et de la seule dont la présentation différera de celle des autres. En effet, cette partie, qui prend en général place en haut à gauche, ne comporte pas de titre. Le recruteur comprend sans peine qu'il s'agit de l'état civil d'un demandeur d'emploi. Elle comprend les informations suivantes : prénom et nom, adresse postale, numéros de téléphone (fixe et/ou mobile), télécopie, situation familiale (marié, célibataire, enfants ...), date de naissance, nationalité, le sexe.
- § **Emploi recherché /Domaine de compétence** : cette rubrique donne un aperçu immédiat du profil du chercheur d'emploi et elle doit être remplie avec la plus grande attention.
- § **Education et formation** : il s'agit de détailler et de préciser toutes les études menées, les diplômes préparés et/ou obtenus, les formations suivies, allant généralement de la formation la plus récente à la plus ancienne. Chaque formation est décrite séparément par les informations : l'année de la formation, l'intitulé de la formation, le nom et le type de l'établissement fréquenté, le diplôme obtenu, niveau dans la classification nationale ou internationale.
- § **Expérience professionnelle** : cette rubrique est celle à laquelle un recruteur attache le plus d'importance, car elle met en avant tous les points forts d'un demandeur d'emploi. Organisée généralement sous la forme d'une énumération chronologique, chaque expérience professionnelle est décrite séparément, avec les informations suivantes : l'intitulé du stage ou la fonction occupée, la date d'exercice du stage ou de la fonction, un résumé des tâches effectuées et des responsabilités assumées, le nom de l'entreprise, sa localisation et son secteur d'activité.
- § **Aptitudes et compétences personnelles** : sous cette rubrique un chercheur d'emploi fait un inventaire de toutes ses aptitudes et ses compétences organisationnelles, techniques, informatiques, langue et autres acquis au cours de son parcours professionnel, éducatif, formateur.
- § **Information complémentaire** : un chercheur d'emploi indique ici toute autre information qu'il juge utile, par exemple: publications ou travaux de recherche.

§ **Annexes** : dans cette rubrique un postulant énumère ses pièces jointes.

Selon l'expérience de travail et les compétences, on peut structurer les CVs de trois façons :

§ **Le CV anti-chronologie** : c'est le CV à la "mode". Il consiste à présenter de manière systématique votre dernière expérience professionnelle ou votre dernière formation. Ce type de CV permet de mettre en avant les compétences acquises le plus récemment qui sont la plupart du temps vos expériences les plus significatives. Il est particulièrement conseillé si vous avez connu une progression de carrière régulière et cohérente.

§ **Le CV thématique** : ce type de CV est peu répandu, car difficile à faire, mais il présente, dans certains cas, de nombreux avantages. Il est particulièrement indiqué pour les personnes ayant exercé plusieurs métiers différents sans continuité ou cohérence. Mais il est aussi indiqué si on souhaite minimiser une erreur de parcours ou un trou dans notre expérience car il permet de mettre l'accent sur telle ou telle de nos compétences.

§ **Le CV chronologique** : ce type de CV se fait rare et clairement dépassé de la mode. En revanche il peut être utilisé en annexe dans certains cas. Par exemple, pour les prestataires et les consultants qui verront dans ce type de présentation un bon moyen de mettre en avant l'ensemble des missions qu'ils ont effectués au cours de leur carrière ou dans le cadre de leur dernier contrat de travail qui comptera le plus souvent plus d'une mission.

Le tableau 2.2 récapitule les avantages et les inconvénients de chaque type de CV.

Type de CV	Avantages	Inconvénients	Notre avis
Chronologique	met en valeur les parcours linéaires, logiques et cohérents entre l'expérience professionnelle et la formation	les expériences les plus récentes sont reléguées en fin de partie !	très classique.
Anti-chronologique	met en évidence l'expérience ou la formation la plus récente, donc la plus actuelle.	Lorsque la dernière expérience ne correspond pas au poste visé.	moderne et de plus en plus répandu, dynamique, il est généralement bien accepté.
Thématique	insiste sur les compétences plus que les parcours. Valorise les diverses activités. Estompe l'effet "à trous" du CV chronologique avec ruptures d'activités.	le recruteur est toujours tenté de se livrer à une gymnastique intellectuelle pour retracer la chronologie de nos parcours.	dans le cadre d'une candidature spontanée ou d'une expérience très riche, cela permet rapidement au recruteur de cerner les capacités d'un chercheur d'emploi.

Table 2.2- Comparaison entre les différents types de CV.

3.2. L'offre d'emploi : est un document moins structuré que le CV. Les informations de l'offre, d'après les différents sites d'emplois, peuvent être regroupées dans les catégories suivantes :

- § Entreprise : dans cette catégorie, le recruteur décrit son organisme, ainsi que le secteur d'activité de son organisme.
- § Poste : dans cette catégorie le recruteur décrit les responsabilités et les tâches liées au poste, ainsi que les avantages dont peut bénéficier une recrue.
- § Profil : le recruteur définit dans cette catégorie les diplômes, les compétences, l'expérience professionnelle requise.
- § Information de contact : le recruteur cite son adresse complète, le nom et l'email de la personne à contacter (responsable de recrutement).

3.3. Outils de génération des CVs /offres d'emploi

Il existe plusieurs outils pour la génération des CVs ou des offres d'emploi sous format électronique, à partir d'un ensemble de formulaires à remplir, avec la disponibilité de

listes de choix pour certaines rubriques. Parmi les outils qui nous semblent les plus importants, nous citons :

- § **Le site Europass pour la rédaction d'un CV** (<http://europass.cedefop.europa.eu/>). Le CV Europass est devenu pratiquement la plate-forme d'écriture des CVs dans toute l'Europe pour les demandeurs d'emploi qui veulent se présenter d'une manière claire et concise. Le Curriculum Vitae Europass, est disponible en 26 langues. Europass conseille fortement d'insérer le contenu dans l'ordre suivant : informations personnelles, expérience professionnelle, éducation et formation, aptitudes et compétences personnelles. Le site Europass de rédaction des CVs a rencontré un vif succès : actuellement près de quinze milles visites sont comptabilisées chaque jour ; plus d'un million et demi de CVs ont été créés en ligne, et plus de 8 millions de documents ont été téléchargés (modèles, exemples, instructions...). Par ailleurs, le CV Europass peut être sauvegardé sous différents formats électroniques (Word, OpenOffice, PDF, XML, HTML), ce qui permet sa diffusion par voie électronique, le format PDF a l'avantage de bloquer le contenu et la forme du CV telle que définie par le titulaire, sans possibilité de vandalisme. La mise à jour des données du CV peut se faire également par le biais du site Europass (pour les documents sauvegardés au format PDF ou XML), ce qui évite évidemment de tout retaper à chaque modification.
- § **Le site Annonces emploi** : (www.annoncesemploi.org) : il donne des exemples concrets des CVs et des conseils et services pratiques (Corriger, rediger, traduire) de construction de ceux-ci. Il propose également 110 modèles pour rédiger un CV, il donne la possibilité de télécharger la totalité de ces modèles. Après l'écriture de CV selon les modèles proposés, le candidat suit certaines instructions pour stocker son CV dans la base des CVs du site. Les chercheurs d'emploi peuvent également enregistrer leurs offres d'emplois électroniques dans la base des offres d'emplois.
- § **Le site CVGen** : (<http://cvgen.free.fr>) : il offre un générateur en ligne de CVs. La génération du CV n'est qu'une étape finale du processus de rédaction. Il faut tout d'abord préciser les objectifs professionnels (type de travail souhaité et centres d'intérêt) afin de connaître le candidat et de mieux cibler sa recherche. La création automatique de CV se déroule en plusieurs étapes : l'inscription du candidat, l'introduction des informations personnelles, la description des formations avec des

listes de choix disponibles, la description des expériences professionnelles et une rubrique personnalisée pour des informations spéciales. La mise en page du CV peut être classique ou sous forme d'un arbre XML.

4. Le e-recrutement

Internet a totalement modifié le processus de recrutement des entreprises : de nombreux sites (Jobpilot, Monster, etc.) disposent d'outils pour la génération des offres d'emploi (ou des CVs), permettant de les publier. Le résultat de la sélection est automatiquement envoyé par mail au postulant. Internet a permis d'optimiser le processus de recrutement, et d'augmenter considérablement le nombre des candidatures (plus de 70 000 CVs reçus par Jobgate).

Nous détaillons, ci-dessous le processus de recrutement ainsi que les problèmes rencontrés par le e-recrutement.

4.1. Les phases du processus de recrutement

Du point de vue organisation, un processus habituel de recrutement peut être divisé en quatre phases :

- § Le département de fonctionnement définit les requis pour un poste de travail et écrit l'offre d'emploi.
- § Le département du personnel publie l'offre d'emploi.
- § Le département du personnel reçoit les demandes d'emploi, sélectionne un ensemble et le passe au département de fonctionnement.
- § Après l'interview avec les candidats, la décision de recrutement au poste vacant est prise par le département de fonctionnement et du personnel.

Nous détaillons par la suite la totalité du processus de recrutement [BIZE04] :

- ▶ **Écriture d'une signalisation du travail** : de nos jours, les offres d'emploi sont écrites sous forme de texte libre en utilisant un vocabulaire non contrôlé. Elles sont habituellement structurées dans les rubriques suivantes: la description d'organisation, la description des fonctions, les compétences et les qualifications exigées et les informations de contact.
- ▶ **La publication des CVs/offres d'emploi** : deux manières de publication des

CVs/offres d'emploi sont possibles à travers les portails d'emploi ou les portails des entreprises :

- § Les portails d'emploi : il existe un nombre important de portails d'emploi tels que Monter1, jobpilot2... ils sont financés par les frais de publication des offres d'emploi (La publication des CVs est généralement gratuite). L'avantage de la publication dans un portail d'emploi réside dans le nombre de personnes qui peuvent consulter ces documents.
- § Les portails des entreprises : la publication dans ce cas n'est pas coûteuse, car le recruteur va publier son offre dans son propre site. Mais elle est limitée, parce que les offres sont consultées par un nombre limité de chercheurs d'emploi.
- ▶ **Réception et présélection des demandes des candidats** : il y a deux moyens pour écrire et envoyer un CV :
 - § Les chercheurs d'emplois envoient leur CVs par mail sous format texte (word, pdf, ...) aux organismes. Cette technique ne permet pas l'automatisation de la présélection, le recruteur doit analyser les CVs manuellement pour sélectionner le profil adéquat.
 - § Les candidats remplissent des formulaires Web prédéfinis qui existent dans les sites des organismes ou dans les portails d'emploi. L'utilisation des formulaires permet une certaine automatisation de la présélection. Ce qui réduit son coût élevé.
- ▶ **Décision de recrutement final** : la sélection finale consiste à faire un entretien avec chaque postulant qui possède un CV sélectionné à l'étape précédente puis sélectionner les meilleurs selon le classement des tests.

4.2. Les limite du e-recrutement

L'automatisation de la présélection des CVs (cas d'un recruteur) ou offre d'emploi (cas d'un chercheur d'emploi) est l'objectif visé par le e-recrutement. Malheureusement, il est confronté à plusieurs problèmes résumés dans le tableau suivant (Tableau 2.3).

Problème	description
L'hétérogénéité des formats des CVs/offre d'emploi.	(format Word, HTML, XML, etc) cela peut nécessiter un traitement particulier pour chaque type.
L'échec des formulaires prédéfinis.	l'originalité d'un CV est difficile à mettre en évidence dans un formulaire prédéfini.
Le coût de publication.	les frais de publication élevés sur les portails ont conduit les recruteurs à publier leurs offres seulement sur un nombre limité de portails.
La division du marché de travail.	les portails de travail disponibles sont nombreux mais indépendants. Ils ont divisé le marché du travail en lignes en îles d'informations isolées [BIZE05], ce qui a rendu impossible, pour un chercheur d'emploi, d'avoir une idée globale de tous les postes de travail offerts.
L'inefficacité des moteurs de recherche.	La recherche dans les portails de travail comme monster (http://www.monster.de), d'une offre d'emploi (cas d'un chercheur d'emploi), ou d'un CV (cas d'un recruteur), se fait via des moteurs de recherche propres au site. Cependant, la recherche est purement syntaxique et basée sur des mots clés. [BIZE04]
La formalisation des acquis et des requis.	La difficulté rencontrée pour exprimer les compétences acquises, cas d'un chercheur d'emploi (CV), ou requises cas d'un recruteur (offre d'emploi).

Tableau 2.3- Les Limites de e-recrutement

Les techniques disponibles pour la gestion des CVs/offre d'emploi sont purement syntaxiques et limitées face à l'afflux des informations à traiter [MORI02]. Ce qui a rendu ces grandes masses de données, souvent mal exploitées [TRIC06]. La gestion des compétences et des connaissances sous-jacentes aux documents, qui est primordiale dans le cadre du recrutement [TRIC06], doit être supportée par les technologies du Web sémantique, pour permettre l'automatisation de l'appariement d'une manière efficace.

5. La gestion des connaissances et des compétences

A l'heure actuelle, les sources d'informations et de connaissances dans les entreprises deviennent de plus en plus importantes et elles jouent aussi un rôle crucial pour le développement de l'entreprise. Le partage du travail nécessite un partage de connaissances au sein des entreprises et ce besoin peut devenir encore plus crucial à l'avenir. Les compétences personnelles des employés sont les connaissances les plus importantes dans les entreprises. Afin de maximiser la performance humaine, les compagnies doivent efficacement dépister et manipuler les compétences possédées ou exigées par des personnes dans toute l'entreprise avec un système de gestion de compétences, qui permet d'affecter la personne appropriée à un poste, détecter le manque en compétences des employés pour assurer la formation nécessaire. Alors pour réussir il faut que l'entreprise assure une bonne gestion des connaissances et des compétences de ses employés.

5.1. Gestion des connaissances (KM)

La gestion des connaissances est basée sur la capitalisation des connaissances permettant d'aborder la dialectique entre des métiers traditionnels et les apports extérieurs, les innovations. Elle constitue l'ensemble des méthodes et outils visant à rechercher, conserver et à gérer des connaissances, en utilisant une architecture informatique comme support structurel. Dans ce qui suit, nous présentons sa définition et ses objectifs.

5.1.1 Définition

Tisseyre [TISS99] définit le Knowledge Management (KM) comme "la gestion consciente, coordonnée et opérationnelle de l'ensemble des informations, connaissances des membres d'une organisation au service de cette organisation".

Ce qui introduit trois points essentiels :

- § **La prise de conscience** : il faut que les membres de l'organisation ou du groupe aient une conscience mutuelle à travailler ensemble et à partager des connaissances. Ceci introduit la notion d' "awareness" ou "conscience" qui facilite la coordination implicite [SIMO00].
- § **La coordination** : le KM vise à ce que l'ensemble des membres d'une organisation travaille de la même manière, tournés vers les mêmes objectifs en partageant le maximum d'informations et de connaissances.

§ **Résultats opérationnels attendus** : l'objectif du KM n'est pas de faire plaisir intellectuellement, c'est d'avoir des résultats concrets sur lesquels on puisse compter pour développer sa propre activité, ce qui impose de faire des calculs de retour sur investissement afin de démontrer la fiabilité de la solution.

5.1.2 Objectif de la gestion des connaissances [BENA05]

Les objectifs principaux de la gestion des connaissances sont :

- formaliser les connaissances tacites afin de les rendre mobilisables et opérationnelles au niveau de l'organisation entière [BOUG01].
- créer les conditions convenables à la création de nouvelles connaissances.

5.2. Qu'est ce que une Compétence³ ?

C'est la capacité reconnue d'un acteur à mobiliser des ressources (connaissances, savoir professionnel, qualités comportementales) afin de réaliser une action, d'atteindre un résultat et de répondre à l'attente d'une entreprise, d'une organisation ou de tout système social.

Nous pouvons distinguer plusieurs types de compétences :

- § **Compétences collectives** : différent de la somme des compétences individuelles. La réponse compétente à un problème dépend de la qualité de la collaboration entre les acteurs (réponse collective). La compétence collective serait alors l'organisation ou la structuration des diverses compétences individuelles. Une équipe est un acteur collectif.
- § **Compétences requises** : définissent ce qu'on attend du professionnel en situation de travail. De quels savoir-faire opérationnels validés avons-nous besoin (en réponse aux objectifs de soins).
- § **Compétences réelles** : potentiel du professionnel à agir dans l'action, ce qu'il fait, ce qu'on peut observer, donc évaluer. Elles sont définies par les résultats obtenus lors des évaluations.

L'écart entre les compétences acquises et celles requises définit le champ dans lequel devra opérer la formation proposée aux équipes.

5.3. Gestion des compétences : la gestion des compétences est issue d'un réel besoin des entreprises de maîtriser leur capital immatériel. Elle est complexe vu le nombre important de compétences et d'individus à gérer en entreprise [BLAN04]. Les gens, les connaissances, les

3 : http://www.interef.com/ateliers/grh_demain/fiches/gestion_compétences.htm

capacités et les habiletés réunies dans le milieu du travail constituent un levier fondamental pour la réussite des entreprises.

Suite à ce constat, de nombreuses recherches ont porté sur la compréhension et la définition de la notion « gestion des compétences ». Nous allons citer quelques unes par la suite :

- « Méthode, outil de gestion des ressources humaines qui vise à optimiser la compétitivité d'une entreprise en recherchant la meilleure adéquation possible entre les compétences nécessaires à l'entreprise et les compétences des ressources humaines disponibles³ ».

- « La gestion des compétences a pour fonction l'orientation du personnel et la gestion de leur potentiel de manière optimale. Elle contribue à l'épanouissement professionnel du personnel [BEUM07] ».

Elle comprend plusieurs processus complexes dont nous pouvons citer : l'identification des compétences requises d'un domaine, l'évaluation des compétences acquises par les individus, l'étude de l'adéquation des compétences acquises et requises, la mise à jour des compétences acquises et requises [BLAN04].

Des outils informatiques semblent nécessaires pour la réalisation de ces processus pour [DUJA02]:

- Mieux décrire les métiers dans leur entreprise.
- Décrire de manière systématique les compétences de leur personnel.
- Mieux connaître les compétences de leur personnel pour prendre des décisions adéquates en matière de recrutement.
- Cerner les besoins de formation et vérifier l'adéquation des programmes de formation par rapport à ceux-ci.

L'utilisation de la notion de compétence par les gestionnaires et les spécialistes des ressources humaines dans les organisations, a permis aux organisations de bien gérer leurs ressources humaines. Dans la section 7 nous allons détailler quelques travaux qui visent à gérer les connaissances et les compétences des CVs/offres d'emploi dans un portail de travail.

6. l'appariement sémantique pour le e-recrutement

Le mot appariement possède plusieurs synonymes selon le dictionnaire de la langue française qui sont : accorder, appareiller, assortir, coupler. Il signifie, dans le domaine de

reconnaissance des formes, la mise en correspondance de formes selon un ensemble prédéfini de règles ou de critères. Dans notre contexte de travail qui est le marché d'emploi, il signifie la compatibilité entre les demandes et les offres de travail selon les qualifications. Il vise d'automatiser le processus de recrutement électronique et en particulier la phase de présélection, où les recruteurs (chercheurs d'emploi) se trouvent devant un nombre énorme de CVs à analyser (offres d'emploi).

6.1. L'appariement syntaxique des documents

Consiste à chercher les mots d'un document avec exactitude dans un autre document, comme exemple d'un moteur de recherche syntaxique Google. Dans le cadre du e-recrutement, tous les portails d'emploi disposent de moteur de recherche syntaxique pour la recherche des CVs ou offres d'emploi.

6.2. L'appariement sémantique des documents

Nécessite l'utilisation d'une approche du Web Sémantique telle que l'annotation. L'annotation sémantique est utilisée pour enrichir les documents avec leurs contenus sémantiques, en termes de méta-données associées à des ontologies de domaine.

L'appariement sémantique est une technique qui calcule le degré de rapprochement de deux documents en se basant sur leur similarité sémantique et non seulement sur des mots clés, et cela par la comparaison de leurs méta-données en exploitant des ontologies de domaine pour cette tâche.

Exemple : si on a une offre d'emploi annotée seulement avec la compétence JAVA et deux CVs, le premier annoté seulement avec la compétence SQL et le deuxième avec la compétence C++.

Pour l'appariement sémantique le deuxième CV est plus similaire à l'offre d'emploi que le premier.

7. Les travaux existants

Vu les limites des méthodes de recrutement du web actuel, résultante de sa structure syntaxique qui ne permet pas à un recruteur (respectivement chercheur d'emploi) de trouver le CV (respectivement l'offre) qui correspond à son besoin, et vu la pertinence du web sémantique qui vise de combler le manque de la sémantique dans les documents du Web

traditionnel, pour qu'ils soient lisibles par une machine et donc permettre leurs traitements automatiques, cela est réalisé par l'ajout de certaines sortes de données additionnelles, pour que n'importe quel outil puisse accéder à ces données, les traiter selon ses conditions et présenter ses résultats à l'utilisateur avec plus de pertinence et d'intelligence plutôt qu'une simple recherche par mot-clé. Ces données additionnelles sont les méta-données. Leur représentation s'appuie sur des extensions des fonctionnalités des langages de marquage que le World Wide Web Consortium (W3C) a déjà publié comme des standards (RDF, OWL.....). Leurs ajouts et effectués par l'approche annotation du Web Sémantique qui permet d'assigner à un document des éléments d'une ontologie de domaine.

Parmi les travaux les plus importants qui s'intéressent à l'application de l'annotation pour l'automatisation du e-recrutement, nous citons :

7.1. Une approche basée sur l'annotation pour modéliser le contenu sémantique [YAHIA06]:

Ce travail présente une approche basée sur l'annotation sémantique des CVs et des offres d'emploi pour automatiser le processus de recrutement électronique. Le principe de base consiste à modéliser formellement le contenu de ces documents en termes de leurs acquis (cas des CVs) ou requis (cas des offres d'emploi), à l'aide d'un référentiel commun ontologique entre les recruteurs et les chercheurs d'emploi. L'ontologie de domaine construite est inspirée des parties les plus significatives de ces documents (données signalétiques, formation et expérience professionnelle) et permet la gestion des compétences sous-jacentes. Le modèle proposé de la compétence définit la compétence comme un savoir agir, elle peut être scientifique et technique ou comportementale (savoir être). Elle est caractérisée par un sujet et un niveau d'expertise qui peut avoir l'une des valeurs suivantes : notion (20%), application (50%), maîtrise (70%) ou expert (90%). Le domaine de l'ontologie est « l'informatique et télécommunication » elle est composée de cinq sous-ontologie inter liées sémantiquement (Figure 2.1) :

- § La sous-ontologie « personne » : décrit les caractéristiques personnelles les plus importantes demandées par un employeur ou possédées par un chercheur d'emploi.
- § La sous-ontologie « annotation » : associe au CV (offre d'emploi) respectivement des acquis (requis) correspondants sous forme d'annotation.
- § La sous-ontologie « Métier » : décrit les concepts relatifs aux métiers du domaine.

- § La sous-ontologie «diplôme» : décrit les concepts relatifs aux diplômes du domaine.
- § La sous-ontologie «compétence» : décrit les compétences relatives aux domaines «informatique et télécommunication ». Le cœur de cette ontologie est le modèle de compétence, qui définit la compétence comme suit : une compétence peut être une aptitude ou une compétence scientifique/technique qui peut être générale ou spécifique. Elle possède alors un sujet qui peut être un aspect logiciel ou un thème. Un thème peut être général, mathématique ou informatique et il appartient à une hiérarchie des thèmes. Chaque thème est caractérisé par un attribut «poids» qui représente le pourcentage de sa contribution dans son thème parent.

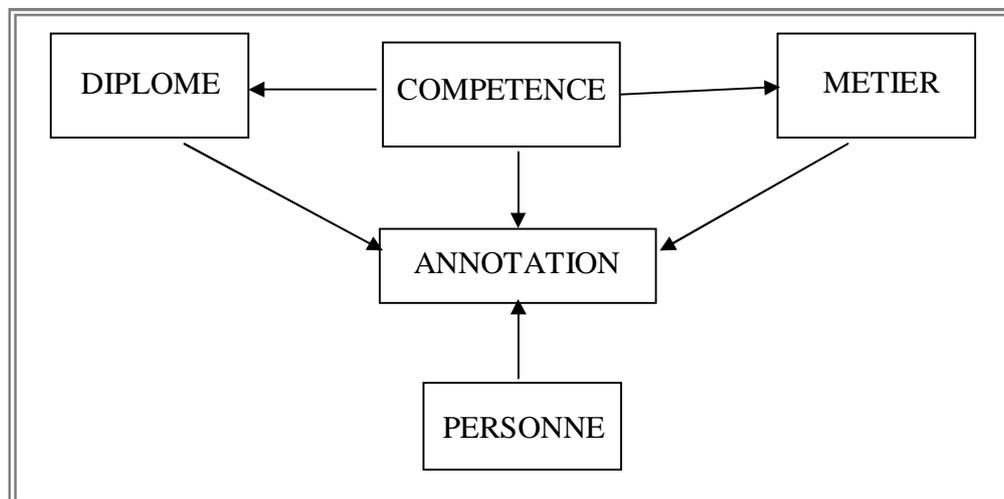


Figure 2.1 - Architecture globale de l'ontologie-ER

Cette ontologie permet à l'utilisateur d'enrichir explicitement son document avec des méta-données qu'il sélectionne depuis les différentes ontologies (Personne, Compétence, Métier, Diplôme) et il les enregistre dans l'ontologie « Annotation » manuellement.

L'appariement sémantique entre les CVs et les offres d'emploi, basé sur le calcul d'un coefficient de rapprochement, peut se faire selon deux techniques complémentaires.

D'une part, l'appariement superficiel s'intéresse au requis/acquis à un niveau superficiel et d'autre part, l'appariement basé compétence exploite toutes les compétences explicites et implicites, sous jacentes à ces documents.

Appariement superficiel : les acquis ou les requis, avec lesquels un document (CV/offre d'emploi) peut être annoté explicitement, sont de quatre types différents : compétence, diplôme, expérience professionnelle (métier et nombre d'années d'expertise) et qualifications personnelles (encapsulées dans une instance de « Person »). Dans ce type d'appariement, la recherche d'un requis du poste de travail dans l'ensemble des acquis du candidat (CV) se fait avec exactitude (existe ou non). Un coefficient de pondération est associé à chacun des types de requis/acquis afin de valoriser son importance dans le calcul du degré d'appariement en question; le coefficient 8 au type « Person » (1 pour chacune qualification), 10 au type «Diplôme», 20 au type « Métier_exp » et 5 au type « Compétence ».

Appariement basé compétence l'appariement basé compétence s'intéresse aux compétences sous-jacentes aux documents annotés. Le principe consiste à chercher chacune des compétences requises (de l'offre d'emploi) dans l'ensemble des compétences acquises (CV). Si cette compétence existe, un poids sera cumulé, sinon la hiérarchie du thème de cette compétence, si elle existe, sera exploitée pour valoriser le niveau du candidat dans ce thème. Un coefficient de pondération est associé à chacun des types de compétences. 2 au type « CompétenceGénérale » et «Aptitude» et un coefficient de 6 au type «CompétenceSpécifique». Le niveau d'expertise de la compétence scientifique et technique s'évalue à (N \cong 20%) si niveau<25%, à (A \cong 50%) si 25% \leq niveau <60%, à (M \cong 70%) si 60% \leq niveau \leq 75% et à (E \cong 90%) si niveau>75%.

Dans notre travail, nous adoptons la définition de la compétence présentée par ce travail : « une compétence peut être scientifique ou technique ou une aptitude ». Nous faisons introduire le contexte d'acquisition de la compétence et la notion de ressource pour la compétence scientifique ou technique et catégorie pour les aptitudes.

7.2. Un processus de recrutement basé ontologie [BIZE05] :

Comme première étape dans ce projet, une ontologie de gestion des ressources humaines est développée (ontologie-RH voir Figure 2.2). Elle est composée de plusieurs sous ontologies [MALG04], chacune d'elles est créée par l'utilisation des référentiels existants.

Education (description de la formation), Organisation (description du recruteur), Person (description du candidat), skills (description des compétences) et industry (inspirée des référentiels métiers existants).

La conception de chaque sous ontologie ainsi que les relations entre les sous ontologies ne sont pas détaillées. Le modèle de la compétence n'est pas défini.

Concernant l'annotation des CVs /offres d'emploi, un outil d'annotation est utilisé. L'utilisateur (recruteur ou chercheur d'emploi) utilise cet outil pour annoter son document (offres d'emploi ou CV respectivement) avec le vocabulaire de l'ontologie HR. Le résultat de l'annotation est un fichier de type RDF de méta-données qui représentent les acquis d'un chercheur d'emploi ou les requis d'un recruteur, décrits avec le vocabulaire contrôlé de l'ontologie-HR.

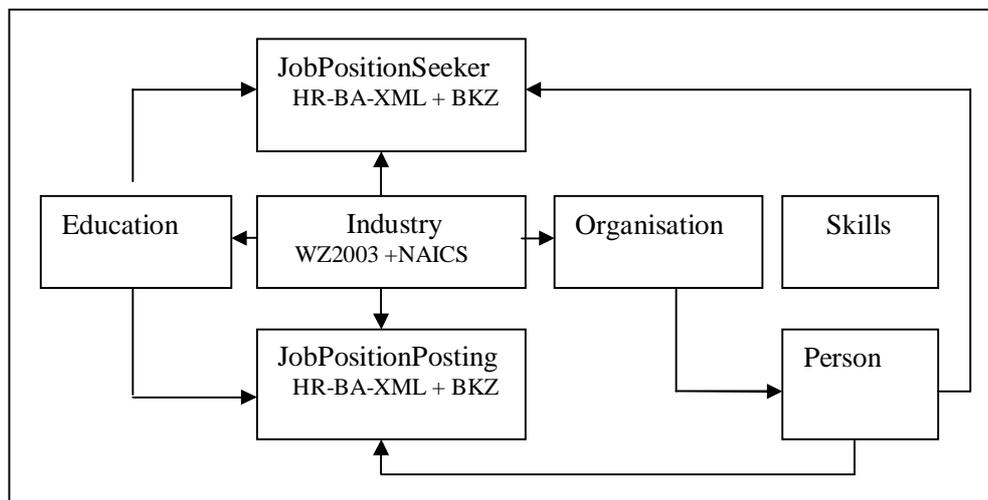


Figure 2.2 - Architecture globale de l'ontologie-HR

Le processus d'appariement consiste à comparer les descriptions des postes de travail offerts et les profils des candidats, en se basant sur leur similarité sémantique et non pas seulement sur des mots clés. Les connaissances spécifiques du domaine ont été représentées sous forme de différentes hiérarchies de concepts (secteurs d'activités, classification de métiers, compétences, etc.). Celles-ci peuvent être utilisées pour déterminer la similarité sémantique entre les concepts, en exploitant les résultats de certains travaux existants, pour le calcul de la similarité dans les graphes ([ZHON02] et [BILL02]). La similarité entre deux concepts C1 et

C_2 de la hiérarchie, est déterminée par la distance $d_c(C_1, C_2)$, qui reflète leurs positions respectives dans la hiérarchie des concepts : $sim_c(C_1, C_2) = 1 - d_c(C_1, C_2)$.

Une valeur Milestone(n) a été associée au préalable à chaque nœud (n) de la hiérarchie utilisée, avec : $Milestone(n) = \frac{1/2}{K^{l(n)}}$ tel que $l(n)$ est la profondeur du nœud et K un facteur supérieur à 1.

La distance entre deux nœuds est celle qui passe par le parent commun le plus proche (CCP) :

$d_c(C_1, C_2) = d_c(C_1, CCP) + d_c(CCP, C_2)$ et $d_c(C, CCP) = Milestone(CCP) - Milestone(C)$.

Le niveau de la compétence (CL) a été aussi pris en charge dans le calcul de la similarité tel que :

$$Sim_p(CL_1, CL_2) = \begin{cases} 1 - \alpha(CL_1 - CL_2) & \text{Si } CL_1 - CL_2 \geq 0 \\ 1 & \text{Si } CL_1 - CL_2 < 0 \end{cases} \quad \text{avec } 0 \leq \alpha \leq 0,25$$

Le degré de similarité entre une offre d'emploi (JPP) et une demande d'emploi (JPS) est calculé comme suit :

$$Sim(JPP, JPS) = \sum_{i \in I} w(C_{jpp}^i) \cdot \max[sim_c(C_{jpp}^i, C_{jps}^j) \cdot sim_p(p(C_{jpp}^i), p(C_{jps}^j)) / j \in J]$$

$$\text{Avec : } \sum_{i \in I} C_{jpp}^i = 1 \text{ et}$$

Chaque compétence requise par le poste de travail offert (C_{jpp}^i) est comparée avec chacune des compétences acquises par le candidat (C_{jps}^j). Cela inclut le calcul de la similarité aussi bien des compétences que des niveaux de compétence. Le meilleur degré de similarité entre le concept requis (compétence requise) et un des concepts acquis (compétence acquise) sera multiplié par le poids correspondant au requis, puis cumulé dans le degré de similarité final. L'algorithme présenté dans ce travail nous a servi d'avantage dans l'écriture des algorithmes d'appariement basé compétence que nous allons présenter dans le chapitre 3.

7.3. BRIDGE [TRIC06]:

BRIDGE est un projet qui vise le traitement des problèmes du e-recrutement en considérant une nouvelle approche basée sur la gestion des compétences. L'idée consiste à employer des ontologies de domaine comme un système de références pour qu'un demandeur d'emploi

(respectivement un recruteur) puisse identifier les compétences sous-jacentes à son curriculum vitae (respectivement son offre de travail). Ces compétences sont les connaissances, les qualifications comportementales, les capacités, acquises par une personne (respectivement requises pour un poste de travail). D'un point de vue technique, ces compétences correspondent aux annotations. Les annotations sont utilisées pour automatiser le processus d'appariement basé compétence entre « CV et offre d'emploi ».

Le modèle de la compétence adoptée, est basé sur la définition suivante : « une compétence correspond à un ensemble de ressources : savoir, comportement ou savoir-faire de base, qui sont mobilisés dans un contexte particulier pour atteindre un objectif ». Ce modèle de compétence nous a servi dans l'élaboration de notre modèle de compétence.

Dans BRIDGE le noyau ontologique se compose de deux types d'ontologies :

- Les ontologies des secteurs : pour chaque secteur d'activité une ontologie est développée. Elle fournit une description détaillée des postes de travail (tâches, savoir, savoir-faire, compétences comportementales) du secteur considéré.
- Les ontologies d'entreprise : elles incluent la description des aspects les plus singuliers des sociétés telles que les tâches spécifiques qui sont exclusivement exécutées par une entreprise ou la description technologique et économique de cette entreprise.

Ces ontologies sont en cours de développement, le langage du web sémantique adopté pour leurs implémentations est OWL. Les référentiels utilisés dans le cadre de ce travail pour la conception des différentes ontologies sont : O*Net⁴, Cigref⁵, CareerSpace⁶.

Jusqu'à ce jour, le processus d'appariement n'est pas traité dans ce projet.

7.4. Evaluation des travaux existants :

Le manque majeur dans ces travaux est l'incomplétude [TRIC06] ou un niveau de détail insuffisant [BIZE05], sauf le travail de yahiaoui et Al dans [YAHIO6]. La modélisation de l'ontologie ainsi que, le modèle de compétence, le processus d'annotation, les services d'appariement sont bien détaillés, ce qui nous a servi dans notre travail.

Néanmoins dans ce travail [YAHIO6], la simplicité du modèle et la négligence du contexte dans ce dernier a engendré quelques faiblesses dans le processus d'appariement. Pour mieux rapprocher ces documents d'un point de vue de la compétence, nous introduisons dans la modélisation de cette dernière, des notions concernant le contexte organisationnel et économique dans lequel cette compétence est acquise ou requise afin que les résultats du

4 : <http://online.onetcenter.org>.

5: <http://www.cigref.fr>.

6: <http://www.career-space.com>.

processus d'appariement basé compétence soient les plus faibles possibles (A titre d'exemple deux demandeurs d'emploi ayant la même compétence et le même niveau dans cette compétence, seront distingués par le contexte s'ils ont travaillé dans des entreprises de taille différente « contexte économique »).

Par ailleurs, le processus d'annotation est compliqué au point qu'il est dédié aux experts (ingénieur ontologique), car un utilisateur, pour annoter son document, doit naviguer dans l'ontologie, et la manipuler pour stocker ces annotations dans l'ontologie « annotation ».

Pour remédier à ces problèmes et afin de simplifier la tâche de l'utilisateur, nous proposons un processus d'annotation guidé par un outil développé dans l'équipe SIBC [ALLI08], que nous l'avons adapté à nos besoins. Nous avons aussi remarqué dans [YAHIO6] la confusion entre les compétences acquises et requises dans le service d'appariement basé compétence où l'algorithme proposé considère que si une personne possède une qualification (diplôme ou expérience professionnelle) alors elle possède obligatoirement toutes les compétences sous-jacentes.

De plus, nous considérons dans notre travail d'autres critères pour l'appariement des CVs et offres d'emploi qui sont les exigences tolérées par le recruteur ou le chercheur d'emploi (type de contrat, type emploi, rotation, déplacement, salaire). Des poids de pondération sont également considérés et appliqués aux exigences (type de contrat, etc.), aux compétences (sous-jacentes aux métiers/diplômes) et aux qualifications personnelles (âge, sexe, etc.) annotant le document de l'utilisateur (recruteur ou chercheur d'emploi), et qui seront explicitement spécifiés par ce dernier afin d'exprimer son choix.

8. Conclusion

Internet a prouvé que les méthodes classiques du recrutement (les annonces dans les journaux, magazine...), sont lentes et chères pour sélectionner le candidat qualifié dans un bref délai. L'utilisation accrue du web dans le cadre du recrutement (les plus grands sites web annoncent plus de 50 000 CVs enregistrés dans leurs bases) a fait de l'Internet une véritable approche de gestion des connaissances (CVs/offres). Cependant la révolution du marché du travail basée sur l'utilisation des technologies de l'information n'est pas accompagnée d'un développement des outils efficaces pour la gestion des CVs /offres d'emploi. Les outils de modélisation et de recherche existant des CVs/offres d'emploi sont souvent syntaxiques et limités.

Introduire les technologies du web sémantique tel que l'annotation via le vocabulaire d'une ontologie de domaine semble bénéfique pour enrichir les documents avec leurs contenus sémantiques. Pour cela elle est adoptée par certains travaux qui sont en cours de réalisation. L'idée de base consiste à utiliser les éléments d'une ontologie (concept, instance, relation) pour expliciter le contenu sémantique des CVs/offres d'emploi puis utiliser ces éléments pour automatiser leurs rapprochements. Le travail de magister de Leila Yahiaoui [YAHIO6a], préparé au département informatique de l'université de Constantine est le plus complet au niveau modélisation. Mais ce dernier souffre de quelques lacunes, comme la négligence du contexte de la compétence (organisationnel et économique) dans la modélisation de la compétence. Par ailleurs, des critères importants tels que certaines exigences (type de contrat, type d'emploi, rotation, déplacement, salaire), n'ont pas été considérés. Ceci a engendré quelques faiblesses dans le processus d'appariement entre les CVs et les offres d'emploi.

Notre contribution, qui fera l'objet du troisième chapitre de ce mémoire, consiste à pallier à ces insuffisances et de plus, nous considérons dans notre travail, des poids de pondération à appliquer aux exigences (type de contrat, salaire, etc.), aux compétences (sous-jacentes aux métiers/diplômes) et aux qualifications personnelles (âge, sexe, etc.) annotant le document de l'utilisateur (recruteur ou chercheur d'emploi), et qui seront explicitement spécifiés par ce dernier afin d'exprimer son choix.

Chapitre 3

Appariement sémantique entre les CVs et les offres d'emploi

Sommaire

1. Introduction
 2. L'objectif de notre travail
 3. L'architecture du système proposé
 4. Modélisation de la compétence proposée
 - 4.1. Définition des éléments utilisée dans le modèle de la compétence proposée
 - 4.2. Le modèle de la compétence proposée
 5. La modélisation du contenu sémantique a base de l'ontologie- ER
 - 5.1 Description de l'ontologie-ER
 6. Le processus d'annotation des CVs /offre d'emploi
 - 7 L'appariement sémantique
 - 7.1. Convention d'écriture et formule utilisées
 - 7.2. Principe d'appariement des qualifications personnelles
 - 7.3. Principe d'appariement des exigences
 - 7.4. L'appariement des compétences
 - 7.4.1. Principe de l'appariement Exact des compétences « CompExact »
 - 7.4.2 Principe de l'appariement Partiel des compétences « PartielComp »
 - 7.4.3. Principe de l'appariement Rapproché des compétences « RapprochéComp »
 - 7.4.4. Discussion sur les algorithmes d'appariements proposés.
 8. Conclusion
-

« Il y a une science qui étudie l'être en tant qu'être
et les attributs qui lui appartiennent essentiellement »

Aristote

1. Introduction :

Pour automatiser le rapprochement des CVs et des offres d'emploi par l'utilisation des technologies du web sémantique, il est nécessaire de modéliser leurs contenus sémantiques par une ontologie. Le référentiel commun (ontologie) est inspiré des parties communes entre ces documents (diplôme, expérience professionnelle, compétence, qualification personnelle, exigence). Ce référentiel peut être ensuite utilisé par un recruteur (ou un chercheur d'emploi) pour annoter son document avec ses requis (ses acquis). Les méta-données résultantes sont exploitées pour automatiser le processus de rapprochement entre l'offre et les CVs disponibles (ou vice versa). Le résultat de rapprochement permet au recruteur de sélectionner le CV adéquat pour un poste de travail et à un chercheur d'emploi de trouver le poste qui correspond le plus à ses qualifications.

Dans ce chapitre, nous proposons une technique d'appariement des qualifications personnelles (Sexe, nationalité, etc.) requis et acquis, une technique d'appariement des exigences (rotation, contrat, etc.) requis et acquis, trois techniques d'appariement des compétences (appariement exact, appariement partiel, appariement rapproché) requis et acquis. Le degré d'appariement final est le rapport entre la somme des poids des acquis vérifiés et celle des poids des requis. Le chapitre est organisé comme suit : dans la section 2, nous détaillons les objectifs de notre travail. La section 3 présente l'architecture globale du système proposé et dans laquelle nous retrouvons essentiellement un composant ontologique, une interface d'annotation, un composant d'appariement et un composant sélection du formateur. La section 4 décrit le modèle de la compétence proposée. La section 5 présente l'ontologie développée pour modéliser le contenu sémantique des CVs/offres d'emplois. La section 6 présente le processus d'annotation. Dans la section 7, nous décrivons les services d'appariement sémantique à appliquer entre les documents et qui traitent trois aspects : i) les qualifications personnelles, ii) les exigences, iii) les compétences. Nous terminons ce chapitre par une conclusion.

2. L'objectif de notre travail

L'objectif de notre travail consiste à proposer un processus d'appariement sémantique à appliquer

entre les fichiers des annotations des offres d'emploi et des CVs, ces derniers sont annotés via une ontologie qui modélise leur contenu sémantique. L'aboutissement d'un tel objectif est le fruit d'une progression dans la réalisation des sous-objectifs suivants :

1. Proposition d'une modélisation pertinente du contenu sémantique à base d'ontologie des CVs et des offres d'emploi, axée sur la modélisation de la compétence. Le modèle de la compétence proposé est riche et tient compte du type de la compétence (scientifique ou technique ou aptitude), des ressources relatives aux compétences scientifiques (habilité, savoir, savoir-faire), de la catégorie relative aux aptitudes (managériales, action et réalisation,...), du niveau de l'expérience et des contextes organisationnel et économique.
2. Proposition d'un processus d'annotation sémantique de documents, basé sur l'exploitation et l'instanciation de l'ontologie construite.
3. Proposition des services d'appariement sémantique à appliquer entre les CVs et les offres d'emploi et qui traitent trois aspects : i) les qualifications personnelles, ii) les exigences, iii) les compétences.

3. L'architecture du système d'appariement et d'annotation proposé

L'architecture du système proposé est donnée dans la figure 3.1 elle est constituée des composants suivants :

- ▶ *L'ontologie ER* : utilisée pour la description du contenu des CVs, des offres d'emploi, des programmes de formation.
- ▶ *Les serveurs des documents* : ils permettent le stockage des CVs et des offres d'emploi.
- ▶ *Les dépôts des annotations* : les dépôts sont utilisés pour stocker les fichiers des annotations des CVs, ceux des offres d'emploi ainsi que ceux des programmes de formation.
- ▶ *Le Raisonneur + les APIs de protégé* : pour l'accès et l'interrogation des fichiers des annotations et de l'ontologie-ER.
- ▶ *L'interface du système* : elle se compose de :
 - § *Une interface d'annotation* : elle permet à un utilisateur d'annoter son document (CVs/offre d'emploi) ou sa page WEB (formateur), par l'instanciation et l'exploitation de l'ontologie-ER. Le résultat de cette opération est un fichier des annotations qui va être stocké dans l'un des trois dépôts.

- § *L'interface d'appariement* : elle permet aux recruteurs et aux chercheurs d'emploi d'interroger le composant d'appariement, comme elle leur présente les résultats retournés par ce dernier.
- § *L'interface de sélection d'un formateur* : elle permet aux recruteurs d'interroger le composant « sélection d'un formateur », comme elle présente aux recruteurs les résultats retournés par ce dernier.
- *Le composant d'appariement* : il permet de calculer les degrés de rapprochement des acquis avec les requis. Le résultat est un ensemble de quintuplés (URI, DA_{final} , $DA_{\text{QPersonnel}}$, DA_{Exigence} , $DA_{\text{compétence}}$) trié par le degré d'appariement final (DA_{final}) suivi par les trois autres degrés ($DA_{\text{compétence}}$, $DA_{\text{QPersonnel}}$, DA_{Exigence}) leur ordre de tri est en fonction du choix de l'utilisateur, l'URI est l'identifiant d'un CV ou d'une offre d'emploi, DA_{final} est le degré d'appariement de tous les acquis avec tous les requis, $DA_{\text{QPersonnel}}$ est le degré d'appariement selon les qualifications personnelles, DA_{Exigence} est le degré d'appariement selon les exigences, $DA_{\text{compétence}}$ est le degré d'appariement selon les compétences.
- *Le composant sélection formateur* : un chercheur d'emploi sélectionné dans le processus d'appariement dans la majorité des cas ne correspond pas à 100 % au besoin du recruteur. Pour cela l'employeur utilise le composant sélection d'un formateur dont le rôle est la sélection d'un formateur qualifié pour former un candidat choisi par l'employeur.

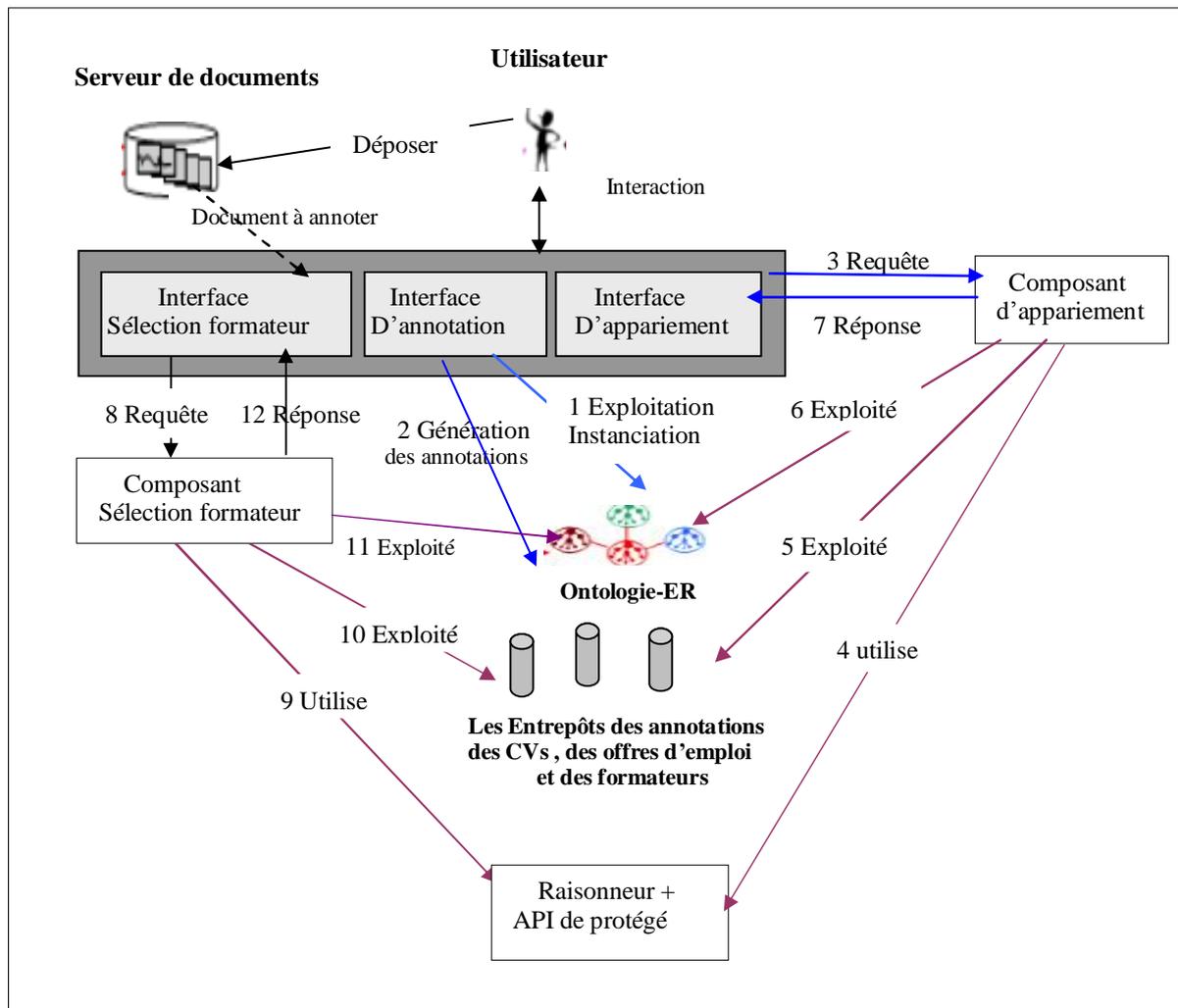


Figure 3.1- Architecture du système proposé.

4. Modélisation de la compétence proposée

La gestion des compétences est issue d'un réel besoin des entreprises de maîtriser leur capital immatériel. Elle est complexe, vu le nombre important de compétences et d'individus à gérer [BLAN04]. Pour une meilleure gestion des compétences, la formalisation est devenue nécessaire. Cette formalisation se base essentiellement sur un modèle pertinent de la compétence.

Le modèle de la compétence est l'ensemble des facteurs de succès qui incluent les comportements principaux exigés pour l'excellente exécution dans un rôle particulier [TECH00]. Les compétences sont les qualifications et les caractéristiques dont l'individu a besoin pour accomplir un travail d'une manière efficace [BAUG97].

4.1. Définition des éléments utilisés dans le modèle de la compétence proposée

Avant de décrire le modèle de la compétence proposée, nous définissons en premier lieu les éléments qui constituent ce dernier :

- **Le savoir (connaissance) :** c'est toute chose qu'on acquiert et enregistre intellectuellement. Il concerne tout ce qui peut être appris dans les systèmes d'éducation ou d'enseignement. Cette catégorie inclut les connaissances théoriques (exemple : la connaissance des lois de la thermodynamique), les connaissances sur des objets existants (telles que la connaissance de l'opération de test d'un équipement) et les connaissances procédurales (telles que la connaissance de la procédure de montage d'une carte électronique pour un type de contrôleur programmable) [TRIC06].
- **Savoir-faire :** il est relié à l'expérience personnelle et les conditions de travail car il est acquis par la pratique. Il s'agit des capacités opérationnelles et expérimentales, de savoir faire normalisé (telles que l'application des procédures de travail). Les savoir-faire et les connaissances sont très importants car ils permettent l'évolution de la compétence de l'individu [TRIC06].
- **L'habilité¹ :** c'est la tendance naturelle de faire quelque chose avec succès (don).
- **Les aptitudes :** c'est l'ensemble des compétences comportementales [TRIC06] telles que :
 - Commandement (la capacité de guider, de motiver, d'influencer dans l'organisation).
 - Adaptabilité (capacité de s'adapter à des nouvelles conditions, être réceptif à des nouveaux avis et idées).
- **Contexte de la compétence**

La compétence est toujours liée à un contexte. Le contexte de la compétence est relié à l'environnement dans lequel la compétence est acquise. Il présente les conditions et les contraintes pour l'acquisition des compétences.

4.2. Le modèle de la compétence proposée

L'objectif de notre travail est de permettre à l'utilisateur (chercheur d'emploi ou recruteur) de rendre explicites toutes les compétences sous-jacentes à son document (CV ou offre d'emploi) d'une façon automatique. Pour cela, il faut représenter formellement ces compétences par un modèle formel et pertinent, dont le but est de pouvoir les expliciter et les manipuler par machine afin de fournir des services d'appariement des compétences puissants.

1: www.encyclopedia.com/dictionary/ability.html

La figure 3.2 présente ce modèle, basé sur la définition suivante : une compétence peut être une *aptitude comportementale* ou une *compétence scientifique ou technique*. Les aptitudes sont classées selon des catégories [CIGRE05], alors que la compétence scientifique/technique est caractérisée par la mobilisation d'un ensemble de *ressources* pouvant être *un savoir, un savoir-faire* ou *une habilité* (un don). Chaque savoir ou savoir-faire est lié à un *thème* particulier, appartenant à une hiérarchie de thèmes *généraux* ou *spécifiques* et à un domaine (le domaine informatique et télécommunication dans notre cas).

La compétence est également caractérisée par un *niveau d'expérience*, dont la valeur dépend du type de celle-ci. Dans le cas d'une compétence scientifique ou technique, ce niveau peut avoir l'une des valeurs suivantes : Débutant (D/1), Intermédiaire (I /2), Moyen (M/3), Compétent (C/4) ou Professionnel (P /5) ; alors qu'il peut avoir juste l'une des deux valeurs, forte (FO/2) ou faible (FA/1), dans le cas d'une aptitude. Le contexte de la compétence sous-entend le *contexte organisationnel et économique*. Le contexte organisationnel représente une évaluation du groupe de formation (cas d'une compétence mobilisée par un diplôme) ou du service de travail (cas d'une compétence mobilisée par un métier), selon le nombre des personnes engagées [DAGM01], il peut être {grand service (6), service moyen (5), petit service (4), petit groupe (3), groupe moyen (2), grand groupe (1)}. Le contexte économique décrit plutôt l'importance de l'organisme formateur (cas d'un diplôme) ou de l'organisme recruteur (cas d'un métier), selon la réputation ou l'importance économique de cet organisme [DAGM01] (chiffre d'affaire de la société, qualification d'une université parmi d'autres, etc.). Un contexte économique peut être {Grande entreprise (6), entreprise moyenne (5), petite entreprise (4), grande école (3), université (2), centre universitaire (1)}.

Remarque : plus de détails concernant le contexte se trouvent dans l'annexe A.

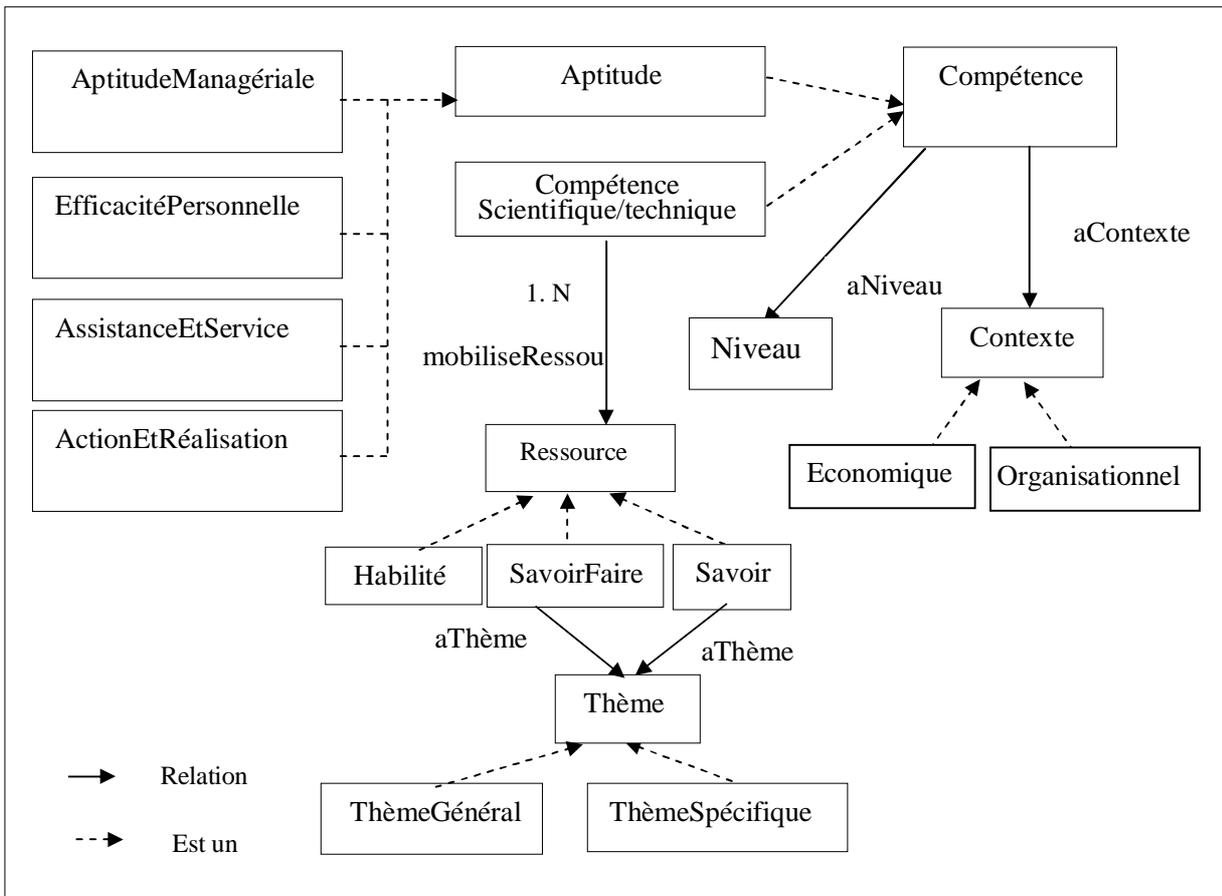


Figure 3.2 - Le Modèle de la compétence proposée

5. La modélisation du contenu sémantique à base de l'ontologie- ER

Les ontologies sont cruciales dans le cadre du e-recrutement car elles permettent aux recruteurs et aux chercheurs d'emploi de partager un référentiel commun pour décrire le contenu de leurs documents d'une façon précise, sémantique et formelle. Ce référentiel servira comme support pour le processus d'annotation générant des données formalisées, permettant ainsi un traitement automatique dans le cadre d'un rapprochement entre les CVs et les offres d'emploi. Dans cette perspective, nous **avons construit** une ontologie nommée « ontologie-ER » [BOUC08] dont les concepts et les relations sont inspirés des parties communes les plus significatives pour les CVs et les offres d'emploi (qualification personnelle, exigence, diplôme, expérience professionnelle, compétence).

5.1 Description de l'ontologie-ER

La figure 3.3 présente les éléments de cette ontologie, que nous décrivons :

- **Document** : c'est le concept décrivant le document à annoter (CV ou offre) à travers son type, son URI, etc.
- **Qualification personnelle** : c'est un concept décrivant les caractéristiques personnelles les plus importantes, qu'un recruteur peut exiger ou qu'un candidat peut avoir. Il s'agit du sexe, de l'âge, de la nationalité, de la ville de résidence, etc.
- **Exigence** : ce concept décrit certaines contraintes imposées par le recruteur ou le chercheur d'emploi telles que : le salaire, le type de contrat, le déplacement, etc.
- **Diplôme** (voir l'annexe A): c'est un concept qui décrit les diplômes appartenant à une famille particulière « FamilDiplôme ». Un référentiel pour ces diplômes « RéférentielDiplomes », concernant un domaine particulier « Domaine », est inspiré de la réforme LMD [LMD04] de l'enseignement supérieur algérien. Le concept diplôme est lié également au concept « Compétence » pour pouvoir attester les compétences mobilisées par un diplôme particulier.
- **Métier** (voir l'annexe A) : c'est un concept qui décrit les métiers appartenant à une famille particulière « FamilMétier ». Un référentiel pour ces métiers « RéférentielMétiers », concernant un domaine particulier « Domaine », est inspiré de CIGREF [CIGRE05]. Le concept métier est lié également au concept « Compétence » pour pouvoir attester les compétences mobilisées par un métier particulier.
- **Compétence** : c'est un concept décrivant la compétence selon le modèle proposé dans la section 4. Ses instances sont liées généralement à un métier ou à un diplôme particulier et aux ressources mobilisées par cette compétence.

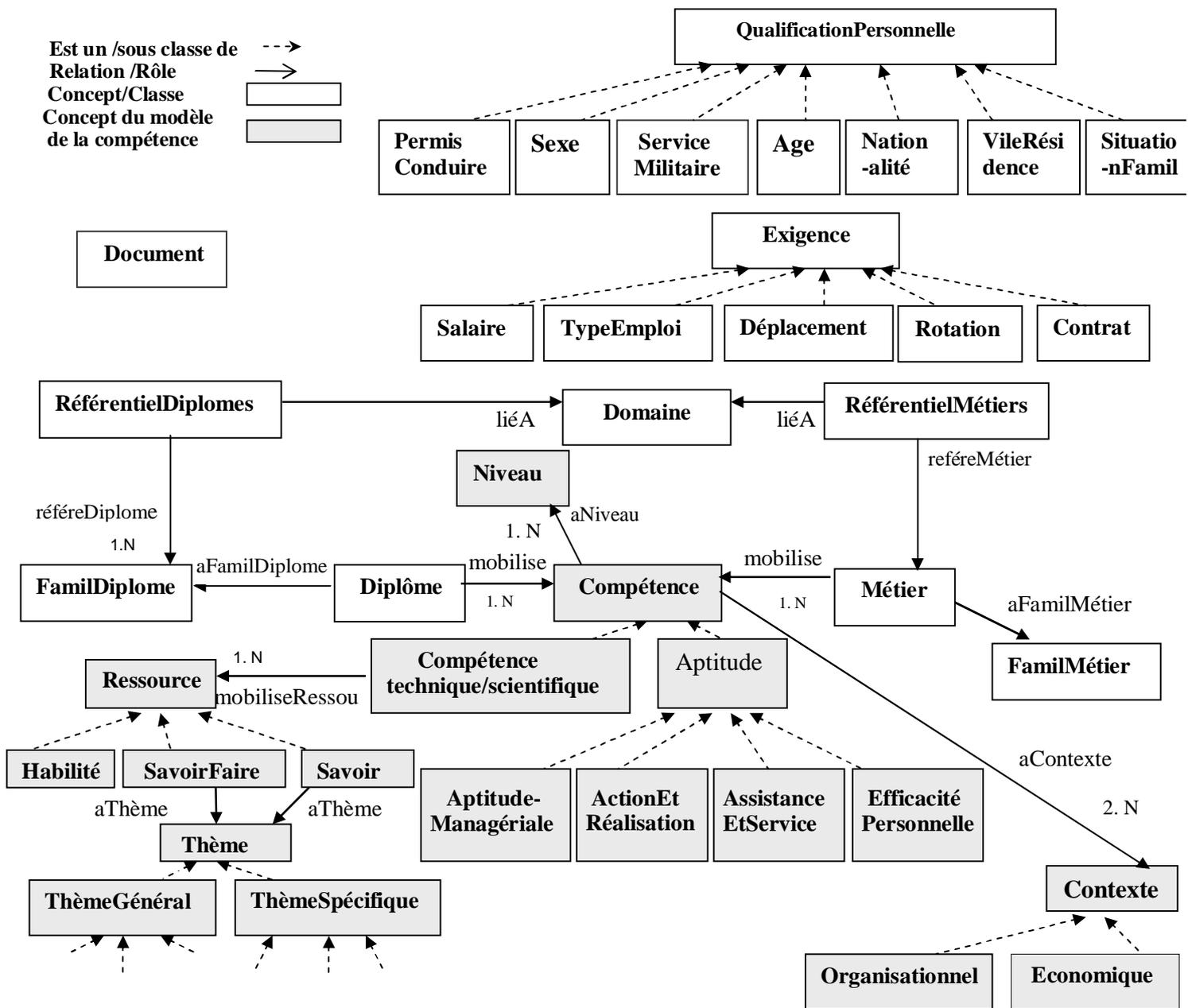


Figure.3.3- Architecture de l'ontologie-ER proposée

6 Le processus d'annotation des CVs /offre d'emploi :

Un recruteur ou un chercheur d'emploi peut exploiter l'ontologie-ER pour annoter son document CV/offre à l'aide d'un outil d'annotation [ALLI08] que nous avons adapté pour l'annotation de ces documents. Les instances des métiers, des diplômes et des compétences sous-jacentes à ces deux derniers sont créées à l'avance pour que l'utilisateur s'en servira durant le processus d'annotation de son document. Il procède alors en plusieurs étapes [BOUC08] :

1. Instancier la classe « Document » pour décrire son document (URI).
2. Instancier les sous-classes de « QualificationPersonnelle » pour décrire les qualifications personnelles acquises par le chercheur d'emploi (cas du CV) ou requises par le recruteur (cas de l'offre). Pour chaque requis /acquis, un poids d'importance est fixé par l'utilisateur ainsi que d'autres informations importantes telles que la distance tolérée entre la ville de résidence requise par lui et la ville de résidence du chercheur d'emploi (ou vice versa). Ainsi la différence maximale d'âge tolérée par le recruteur (chercheur d'emploi) entre l'âge requis par lui (acquis) et l'âge du chercheur d'emploi (requis par le recruteur).
3. Instancier les sous-classes de « Exigence » pour décrire certaines contraintes imposées par le recruteur ou le chercheur d'emploi telles que (la valeur du salaire, le déplacement, etc.), l'utilisateur assigne également un poids à chacune de ces contraintes selon leurs importances pour lui, ainsi que d'autres données, telles que la différence de salaire tolérée par un recruteur entre sa proposition et celle exigée par un chercheur d'emploi.
4. Sélectionner explicitement des compétences sous-jacentes à un métier ou un diplôme figurant dans son document, via un choix dans une liste d'instances prédéfinies, inspirées des différents référentiels de métiers et de diplômes utilisés. Il doit également assigner un poids à chacune de ces compétences selon leurs importances pour lui.

Le résultat de ce processus est un ensemble méta-données réparties sur deux entrepôts de méta-données (celui des CVs et celui des offres d'emploi).

7 L'appariement sémantique :

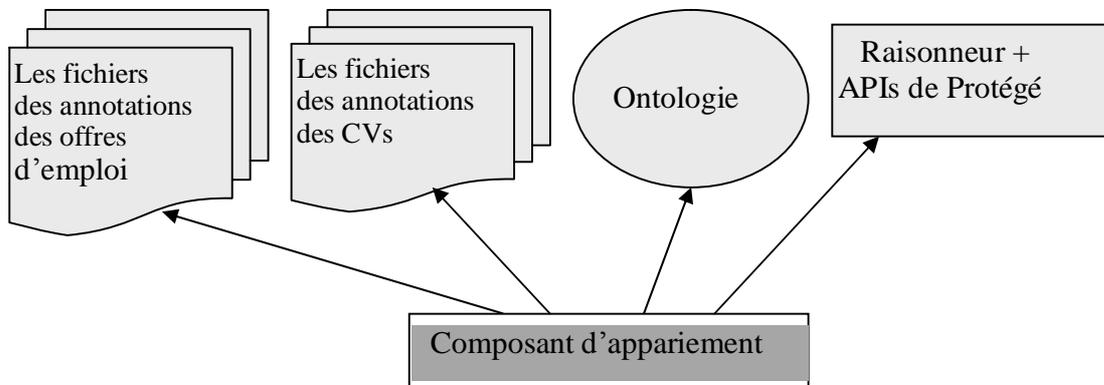


Figure.3.4- Processus d'appariement

L'appariement sémantique (figure 3.4) des requis d'un recruteur (fichiers d'annotations de l'offre d'emploi : FA_{offre}) et les acquis d'un chercheur d'emploi (fichier d'annotations d'un CV : FA_{CV}) revient à calculer leur degré de rapprochement selon les trois critères [BOUC08] : qualifications personnelles, exigences et compétences (voir figure.3.5). Pour chaque critère, nous proposons le service (algorithme) d'appariement adéquat. Pour les compétences, nous proposons trois techniques de rapprochement, exact, rapproché, partiel. Une technique est choisie à la fois par l'utilisateur selon son choix. Le résultat final d'appariement entre une offre d'emploi et un ensemble de CVs (ou vice versa) est une liste de quintuplés (URI, DA_{final} , $DA_{QPersonnel}$, $DA_{Exigence}$, $DA_{compétence}$) trié par le degré d'appariement final (DA_{final}) suivi par les trois autres degrés ($DA_{compétence}$, $DA_{QPersonnel}$, $DA_{Exigence}$) leur ordre de tri est en fonction du choix de l'utilisateur, l'URI est l'identifiant d'un CV ou d'une offre d'emploi, DA_{final} est le degré d'appariement de tous les acquis avec tous les requis, $DA_{QPersonnel}$ est le degré d'appariement selon les qualifications personnelles, $DA_{Exigence}$ est le degré d'appariement selon les exigences, $DA_{compétence}$ est le degré d'appariement selon les compétences.

Dans l'écriture de l'algorithme Appariement CV/offre nous utilisons les services (algorithmes) Exigence, QPersonnel, CompExact, PartielComp, RapprochéComp, chaque service est détaillé dans les sections suivantes. Ainsi nous utilisons les variables FA_{offre} , FA_{cv} , choix, DA_{final} , $DA_{QPersonnel}$, $DA_{Exigence}$, $DA_{compétence}$, TotalQP, TotalEX, TotalComp pour représenter respectivement le fichier des annotations de l'offre, le fichier des annotations du CV, le type de service d'appariement des compétences choisi, le degré d'appariement de tous les acquis avec

tous les requis ,le degré d'appariement des qualifications personnelles , le degré d'appariement des exigences ,le degré d'appariement des compétences, la somme des poids des qualifications personnelles requises, la somme des poids des exigences requises, la somme des poids des compétences requises.

Appariement CV/offre (FAoffre, FACv , choix, DA_{final} ,DA_{QPersonnel} ,DA_{Exigence} ,DA_{compétence})

Exigence (FAoffre, FACv, DA_{Exigence},TotalEX)
 QPersonnel (FAOffre, FACv, DA_{QPersonnel} , TotalQP)

Si choix =1 **alors**
 CompExact (FAOffre, FACv, DA_{compétence}, TotalComp)

Sinon
Si choix =2 **alors**
 PartielComp(FAOffre, FACv, DA_{compétence}, TotalComp)

Sinon
Si choix =3 **alors**
 RapprocheComp(FAOffre, FACv, DA_{compétence}, TotalComp)

$$DA_{final} = \frac{(DA_{Exigence} * TotalEX) + (DA_{QPersonnel} * TotalQP) + (DA_{compétence} * TotalComp)}{TotalEX + TotalQP + TotalComp}$$

Fin

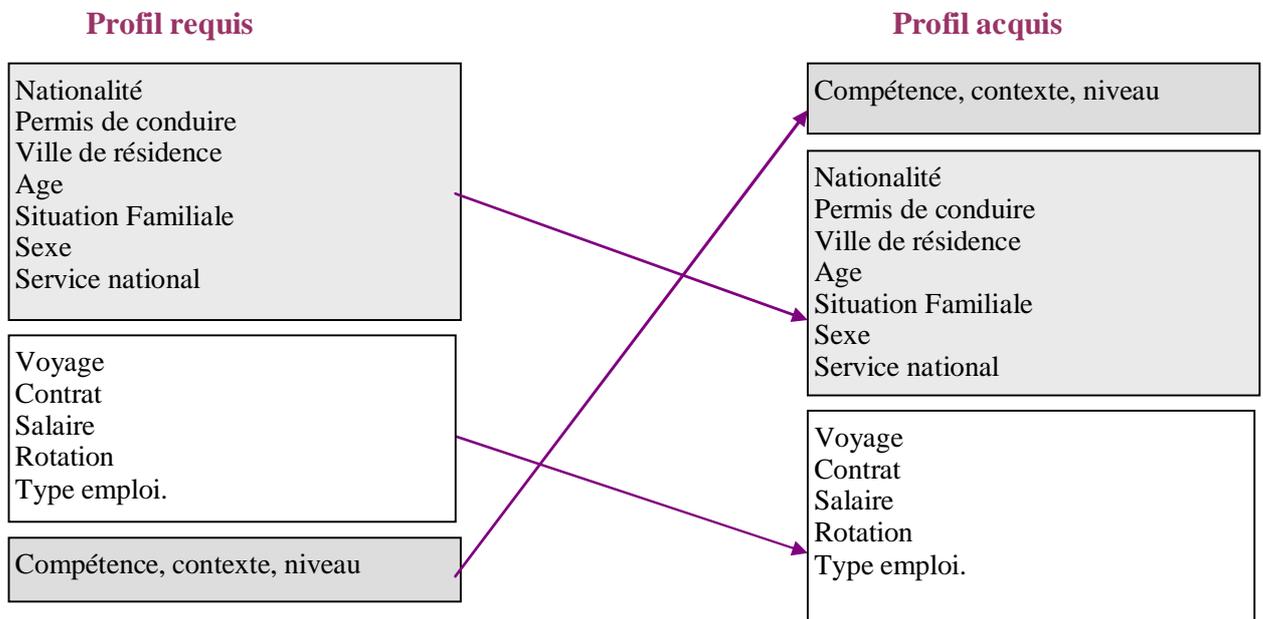


Figure.3.5- Appariement des acquis /requis

7.1. Convention d'écriture et formules utilisées

Nous définissons d'abord formellement la structure des méta-données puis, nous citons quelques conventions d'écritures utilisées dans les algorithmes.

Nous considérons le terme « méta-donnée » comme le synonyme d'instance d'ontologie. La définition d'une structure de méta-donnée est comme suit:

Définition formelle (structure des méta-données) [MAE02].

Une structure de méta données est un 6-tuple $MD = (O, I, L, inst, instr, instl)$, où :

- O : est une ontologie.
- I : Ensemble d'éléments appelés instances.
- L : Ensemble de valeurs littérales.
- $inst : C \rightarrow 2^I$ une fonction appelée instanciation des concepts.
 $inst(C) = I$ peut être écrite $C(I)$.
- Une fonction $instr : P \rightarrow 2^{I \times I}$ appelée instanciation des relations.
 Pour $instr(P) = \{I_1, I_2\}$ on peut l'écrire $P(I_1, I_2)$.
- La fonction $instl : P \rightarrow 2^{I \times L}$ appelée instanciation d'attributs, a comme résultat, une valeur littérale L .

Les services (algorithmes) proposés décrits par la suite utilisent les conventions suivantes :

- $A \rightarrow C$: le concept C subsume le concept A dans la hiérarchie des concepts et A est subsumé par C .
- $C(i)$: i est un individu/instance du concept/classe C alors **classe** $(i) = C$.
- **i.nomAtr** : la valeur de l'attribut « **nomAtr** » de l'individu i ou l'ensemble des individus $\{i_j, j \geq 0\}$ avec lesquels i est relié par le rôle « **nomAtr** » \Rightarrow **nomAtr** $(i, i_j) j \geq 0$.
- **i.nomRole** : l'ensemble des individus $\{i_j, j \geq 0\}$ avec lesquels i est relié par le rôle « **nomRol** » \Rightarrow **nomRol** $(i, i_j) \forall j \geq 0$.

7.2. Principe d'appariement des qualifications personnelles

Les qualifications personnelles constituent la première catégorie d'informations à traiter. Pour illustrer leur appariement, nous considérons des éléments tels que l'âge, la ville de résidence, la nationalité, le sexe, etc. Chaque qualification personnelle requise (R) annotée dans l'offre (FA_{Offre}) est recherchée avec exactitude dans l'ensemble des acquis annotés dans le CV (FA_{CV}). Dans le cas d'une égalité avec un acquis de même type, le poids d'importance du requis est

cumulé. Pour le requis « ville de résidence », le recruteur limite la distance acceptée entre la ville requise et celle de résidence d'un chercheur d'emploi par une distance maximale acceptée. Si la distance entre les deux villes est inférieure à la distance maximale tolérée, elle servira dans l'ajustement du poids du requis avant que ce dernier ne soit cumulé. Le même principe s'applique pour l'âge. L'utilisateur instancie la différence maximale d'âge accepté (MaxAge) par lui, si la différence entre l'âge requis et acquis (ou vice versa) est inférieure à MaxAge, le poids du requis est ajusté avant d'être cumulé. L'appariement des qualifications personnelles est le rapport entre le cumul des poids des requis vérifiés et le cumul des poids de toutes les qualifications requises par le recruteur. La fonction (Distance) est utilisée dans l'algorithme de l'appariement des qualifications personnelles pour retourner la distance entre la ville requise et acquise (ou vice versa). La procédure ExtraireQP est utilisée pour l'extraction des qualifications personnelles depuis un fichier d'annotations.

L'algorithme de l'appariement des qualifications personnelles est détaillé dans la figure.3.6.

7.3. Principe d'appariement des exigences :

Les exigences constituent la deuxième catégorie d'informations à traiter. Pour illustrer leur appariement, nous considérons des éléments tels que la rotation (3x8), le type d'emploi, le salaire, etc. Chaque exigence du recruteur (R) annotée dans le fichier des annotations de l'offre (FA_{offre}), est recherchée avec exactitude dans l'ensemble des exigences d'un chercheur d'emploi annoté dans le fichier des annotations du CV (FA_{CV}). En cas d'égalité avec une exigence de même type, le poids d'importance du requis est cumulé. Le recruteur peut aussi limiter la différence acceptée entre « le salaire » qu'il exige et « le salaire » qui peut être demandé par un chercheur d'emploi par une différence maximale acceptée. Cette dernière sera utilisée pour ajuster le poids à cumuler. Le degré d'appariement des exigences est le rapport entre le cumul des poids des requis vérifiés et le cumul des poids de toutes les exigences annotées par le recruteur. La procédure ExtraireEX est utilisée pour extraire les exigences depuis un fichier d'annotation.

L'algorithme de l'appariement des exigences est détaillé dans la figure.3.7.

QPersonnel (FAOffre, FACv, DA_{QPersonnel}, TotalQP)

QPcv := \emptyset /* les qualifications personnelles du CV */

QPoffre := \emptyset /*les qualifications personnelles de l'offre */

PQpersonnel := 0 /* le cumul des poids de tout les qualifications personnelles vérifié */

TotalOP := 0 /* le cumul des poids de tout les qualifications personnelles requise */

ExtraireQP(FAOffre, QPoffre) /*Extraire les qualification personnel depuis l'offre*/

ExtraireQP(FACv, QPcv) /*Extraire les qualifications personnel depuis le CV*/

Pour chaque R \in QPoffre **Répéter**

Si sexe (A) **alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / sexe (A) et R.sexe =A.sexe **alors**

PQpersonnel := PQpersonnel + R.poids

Si Nationalité (R) **alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / Nationalité (A) et R.nationalité =A.nationalité **alors**

PQpersonnel := PQpersonnel + R.poids

Si ServiceNational (R) **alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / ServiceNational (A) et R.dégagé =A.dégagé **alors**

PQpersonnel := PQpersonnel + R.poids

Si SituationFamiliare (R) **Alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / Situationfamiliale (A) et R.marié =A.marié **alors**

PQpersonnel := PQpersonnel + R.poids

Si VilleR(R) **alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / VilleR (A) et R.ville =A.ville **alors**

PQpersonnel := PQpersonnel + R.poids

Sinon /* MaxDA est la distance maximale acceptée */

Si $\exists A \in$ QPcv / VilleR (A) et distance (R.ville, A.ville) < R.MaxDA **alors**

PQpersonnel := PQpersonnel + (R.poids * (1 - ((1/ R.MaxDA) * Distance)))

Si Age (R) **alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / Age (A) et R.Age =A.Age **alors**

PQpersonnel := PQpersonnel + R.poids

Sinon /* MaxAge la maximale différence d'âge entre l'âge requis et acquis (ou vice versa) */

Si $\exists A \in$ QPcv / (AgeR (A) et (A.age > R.age) et (A.age - R.age) < R.MaxAge) **alors**

PQpersonnel := PQpersonnel + (R.poids * (1 - ((1/ R.MaxAge) * (A.age - R.age))))

Sinon **Si** $\exists A \in$ QPcv / (Age (A) et (R.age > A.age) et ((R.age - A.age) < R.MaxAge) **alors**

PQpersonnel := PQpersonnel + (R.poids * (1 - ((1/ R.MaxAge) * (R.age - A.age))))

Si PermisConduire(R) **alors**

TotalOP := TotalQP + R.poids

Si $\exists A \in$ QPcv / PermisConduire (A) et R.catégorie =A.catégorie **alors**

PQpersonnel := PQpersonnel + R.poids

Fin Répéter

DA_{QPersonnel} := (PQpersonnel / TotalQP) * 100 /* le degré d'appariement des qualifications personnelles*/

Fin

Figure.3.6- Appariement des qualifications personnelles

```

Exigence (FAoffre, FACv, DAExigence, TotalEX)
EXoffre := ∅ /*les exigences de l'offre */ .EXcv := ∅ /*les exigences du CV */
PEExigence := 0 /*le cumul des poids des exigences vérifiés*/
TotalEX := 0 /* le cumul des poids des exigences requise*/
ExtraireEX(FAoffre, EXoffre) /*Extraire les exigences depuis l'offre*/
ExtraireEX(FACv, EXcv) /*Extraire les exigences depuis le CV */
Pour chaque R ∈ EXoffre Répéter
  Si contrat (R) alors
    TotalEX := TotalEX + contratR.poids
    Si ∃ A ∈ EXcv / contrat (A) et R.type = A.type alors
      PEExigence := PEExigence + R.poids
  Si salaire (R) Alors
    TotalEX := TotalEX + R.poids
    Si ∃ A ∈ EXcv / salaire (A) et R. montant) ≥ A. valeur) alors
      PEExigence := PEExigence + R.poids
    Sinon /*.MaxDif : la différence maximal accepté */
      Si ∃ A ∈ QPcv / salaire (A) et (A.valeur - R. montant) < R .MaxDif alors
        PEExigence := PEExigence + (R.poids *(1-((1/ R.MaxDif)*(A. montant - R. montant))))
  Si rotation (R) Alors
    TotalEX := TotalEX + R.poids
    Si ∃ A ∈ EXcv / rotation (A) et R.accepté = A.accepté alors
      PEExigence := PEExigence + R.poids
  Si déplacé (R) Alors
    TotalEX := TotalEX + R.poids
    Si ∃ A ∈ EXcv / déplacé (A) et R.accepté = A.accepté alors
      PEExigence := PEExigence + R.poids
  Si EmploiType (R) Alors
    TotalEX := TotalEX + R.poids
    Si ∃ A ∈ EXcv / TypeEmploi (A) et R.type = A.type alors
      PEExigence := PEExigence + R.poids
Fin répéter

DAExigence := (PEExigence / TotalEX)*100 /* le degré d'appariement des exigences */

Fin

```

Figure 3.7- Appariement des exigences

7.4. L'appariement des compétences

Vu l'importance des compétences pour la sélection d'un chercheur d'emploi le plus qualifié ou de l'offre d'emploi la plus intéressante, nous proposons trois techniques d'appariement des compétences : l'appariement exact (CompExact), l'appariement rapproché (RapprochéComp), l'appariement partiel (PartielComp).

L'appariement exact est appliqué dans le cas où le recruteur cherche un candidat possédant exactement les compétences requises par lui, l'appariement partiel est appliqué si aucun candidat n'est sélectionné par l'appariement exact. Alors le recruteur diminue ses contraintes. L'appariement rapproché est appliqué dans le cas où l'appariement partiel ne donne pas de résultats satisfaisant le recruteur.

7.4.1. Principe de l'appariement Exact « CompExact » :

L'appariement exact est appliqué dans le cas où le recruteur cherche un candidat qui possède exactement les compétences requises pour un poste. Une compétence scientifique ou technique requise (R) est recherchée avec exactitude dans l'ensemble des compétences acquises (Comp_{CV}). Le cas échéant, nous nous intéressons à la compétence acquise dont le produit de son niveau évalué et de son contexte évalué est supérieur ou égal au niveau évalué de la compétence requise multiplié par le contexte évalué de celle-ci. Le même principe est appliqué aux aptitudes.

La procédure CompExact :

Cet algorithme utilise les variables : compétenceoffre, compétencecv, PComp, TotalComp, DA_{compétence} pour représenter respectivement les compétences de l'offre, les compétences de CV, le cumul des poids des compétences requises vérifiées, le cumul des poids de toutes les compétences requises, le degré d'appariement des compétences, d'autres variables sont utilisées pour des résultats intermédiaires. La fonction (EvaContext) est utilisée dans l'algorithme d'appariement exact des compétences. Elle retourne le cumul de la valeur du contexte organisationnel ajusté (multiplié par α) et de la valeur du contexte économique ajusté (multiplié par β). A titre d'exemple, pour une compétence acquise dans un grand service d'une grande entreprise, $EvaContext(compétence)$ vaut la valeur du contexte économique d'une grande entreprise ajustée + la valeur du contexte organisationnel d'un grand service ajusté, dans ce cas elle est égale à $(6 * \alpha + 6 * \beta)$.

Nous proposons pour α la valeur (1/4) et pour β la valeur (3/4) (car le contexte économique est plus important que le contexte organisationnel) mais α et β doivent être ajustées par des experts

du domaine de la compétence. Le rôle de la procédure « ExtraireComp » consiste à extraire des compétences depuis un fichier des annotations. CompExact est détaillée dans ce qui suit :

EvaContext(comp)

contexte : ensemble des contextes d'une compétence.

contexte := {comp.aContexte}, valeur := 0

Pour chaque $C \in$ contexte **répéter**

Si Economique (comp.aContexte) **alors** valeur := C.poids + $\frac{3}{4}$ * valeur

Si Organisationnel (comp.aContexte) **alors** valeur := C.poids + $\frac{1}{4}$ * valeur

Fin répéter

EvaContext := valeur

Fin

CompExact (FAOffre, FACv, DA_{compétence}, TotalComp)

CompétenceOffre := \emptyset /* les compétences de l'offre */

CompétenceCv := \emptyset /* les compétences de Cv */

PComp := 0 /* cumul des poids des compétences vérifiés*/

TotalComp := 0 /* cumul des poids de toutes les compétences requises */

ExtraireComp (FAOffre, CompétenceOffre), ExtraireComp (FACv, CompétenceCv)

Pour chaque compoffre \in CompétenceOffre **Répéter** /* pour chaque compétence de l'offre fait*/

TotalComp := TotalComp + compoffre.poids

Si \exists compcv \in CompétenceCv / compcv.nom = compoffre.nom **et**

(compoffre.aNiveau.valeur * EvaContext(compoffre)) \leq (compcv.aNiveau.valeur)

*EvaContext(compcv)) **alors** PComp := PComp + compoffre.poids

Fin si

Fin répéter

DA_{compétence} := (PComp / TotalComp) * 100 /* le degré d'appariement des compétences*/

Fin

7.4.2 Principe de l'appariement partiel PartielComp :

L'appariement partiel est appliqué dans le cas où le résultat de l'appariement exact ne satisfait pas le recruteur, alors ce dernier diminue ses exigences concernant le niveau et le contexte. Une compétence scientifique ou technique requise (R), est recherchée avec exactitude dans l'ensemble des compétences acquises (Comp_{Cv}). Le cas échéant, nous nous intéressons à la compétence acquise dont le produit de son niveau évalué et de son contexte évalué est supérieur ou égal au niveau évalué de la compétence requise multiplié par le contexte évalué de celle-ci. Si cette contrainte est vérifiée, le poids du requis est cumulé. Sinon nous recherchons la compétence acquise de niveau et de contexte le plus similaire au niveau et au contexte de (R). Dans ce cas, le poids du requis est ajusté avant d'être cumulé. Le même principe est appliqué aux aptitudes. Le

degré de rapprochement entre les compétences acquises et requises, est le rapport entre les poids des requis vérifiés et le cumul des poids de tous les requis.

La procédure PartielComp :

Cet algorithme utilise certains éléments définis auparavant qui sont : CompétenceOffre, CompétenceCv, PComp, TotalComp, $DA_{\text{compétence}}$ d'autres variables sont utilisées SCOrg, SCEco pour représenter respectivement le degré de similarité des contextes organisationnels et le degré de similarité des contextes économiques. D'autres variables simples sont utilisées pour des résultats intermédiaires.

Dans l'écriture de l'algorithme d'appariement PartielComp, nous utilisons les procédures suivantes :

§ La procédure « ExtraireComp » : permet d'extraire les compétences depuis un fichier des annotations.

§ La procédure « SimNiveau » : permet d'estimer le degré de similarité de deux niveaux de compétences, le calcul de degré de similarité est basé sur la formule (4). La procédure est détaillée dans ce qui suit :

SimNiveau (compoffre,compcv)

simn := 0 /* variable intermédiaire*/

simn := compoffre.aNiveau.valeur - compcv.aNiveau.valeur

Si compétence Technique/scientifique (compoffre) **alors**

/*le degré de similarité de deux compétences technique ou scientifique requise et acquise*/

simn := 1/5 * simn. /* 5 niveau*/

Sinon

/*le degré de similarité d'une aptitude requise avec une aptitude acquise*/

simn := 1/2 * simn /* 2 niveau*/

Fin

Si compoffre.aNiveau.valeur -compcv.aNiveau.valeur > 0 **alors**

SimNiveau := 1 - simn

Sinon

SimNiveau := 1

Fin

Fin

§ La procédure « ExtraireContext » permet d'extraire le contexte organisationnel et le contexte économique d'une compétence. Elle est détaillée dans ce qui suit :

ExtraireContext(comp,X,Y)

```

/* extraire les contextes d'une compétence dans la variable contexte*/
Contexte := {compoffre.contexte}
/* extraire le contexte organisationnel */
X := Cont € contexte / organisationnelle (contexte)
/* extraire le contexte économique*/
Y := Cont € contexte / économique (contexte)
Fin

```

§ La procédure « SimContext » : permet d'estimer le degré de similarité de deux contextes de compétences. Premièrement, le degré de similarité des contextes organisationnels SCOrg est calculé, puis le degré de similarité des contextes économiques SCEco est calculé. SimContext est le cumul de ($\alpha * SCOrg$) et ($\beta * SCEco$). Le calcul de SCOrg et SCEco est basé sur la formule (4). SimContext est détaillée dans ce qui suit :

SimContext(compoffre, compcv)

```

/*extraire le contexte organisationnel, économique d'une compétence de l'offre */
ExtraireContext(compoffre, Xof,Yof)
/* extraire le contexte organisationnel, économique d'une compétence de CV */
ExtraireContext(compcv, Xcv ,Ycv)
/*calculé le degré de similarité de deux contextes organisationnels*/
Si X1.poids – Y1.poids > 0 alors
SCOrg := (1- 1/ 6 *(X1.poids – Y1.poids)) /* six contextes Organisationnels*/
Sinon
SCOrg := 1
Fin si
/*calculé le degré de similarité de deux contextes économiques*/
Si X2.poids – Y2.poids > 0 alors
SCEco := (1- 1/ 6 *(X2.poids – Y2.poids)) /* six contextes Economiques */
Sinon
SCEco := 1
Fin si
SimContext := ¼ * SCOrg + ¾ * SCEco /*calculé le degré de similarité de deux contextes*/
Fin

```

L'algorithme de l'appariement Partiel des compétences est détaillé ci-dessous :

```

PartielComp(FAOffre, FACv, DAcompétence, TotalComp)
CompétenceOffre := ∅ /* les compétences de l'offre */
CompétenceCv := ∅ /* les compétences de Cv */
PComp := 0 /* cumul des poids des compétences vérifiées*/
TotalComp := 0 /* cumul des poids de toutes les compétences requises */
S := 0
ExtraireComp (FAOffre, CompétenceOffre), ExtraireComp(FACv, CompétenceCv)
Pour chaque compoffre ∈ CompétenceOffre Répéter
/* pour chaque compétence de l'offre*/
TotalComp := TotalComp + compoffre.poids
/*chercher une compétence de CV qui vérifie les conditions citées*/
Si ∃ compcv ∈ CompétenceCv / compcv.nom = compoffre.nom et
(compoffre.aNiveau.valeur * EvaContext(compoffre)) ≤ (compcv.aNiveau.valeur *
EvaContext(compcv)) Alors
PComp := PComp + compoffre.poids
Sinon/* si la condition de sélection n'est pas vérifiée le poids des requis est ajusté*/
S := Max (SimNiveau (compoffre,compcv) * SimContext(compoffre,compcv))
∀ compcv ∈ CompétenceCv / (compcv.nom = compoffre.nom)
PComp := PComp + compoffre.poids * S
Fin si
Fin Répéter
DAcompétence := (PComp/ TotalComp) * 100
Fin

```

7.4.3. Principe de l'appariement Rapproché "RapprochéComp" :

Dans ce cas, le rapprochement n'est pas sévère. Chaque compétence requise de l'offre est recherchée dans l'ensemble des compétences acquises annotées dans (FA_{CV}), selon ses ressources (cas d'une compétence scientifique/technique), sa catégorie ou son nom (cas d'une aptitude), son niveau et son contexte (cas d'une compétence scientifique/technique ou aptitude). Si la compétence requise est de type aptitude, son nom est cherché avec exactitude. Sinon, sa catégorie est recherchée avec exactitude dans l'ensembles des catégories des aptitudes annotant le CV (FA_{CV}). L'appariement d'une compétence scientifique ou technique requise R avec une compétence acquise A, revient à l'appariement des deux ensembles de ressources mobilisées par ces deux compétences, selon la formule (2) sachant que les ressources sont de trois types (habilité, savoir, savoir-faire). L'habilité (H) de la compétence requise (R) est recherchée avec exactitude dans l'ensemble des habilités de la compétence acquise (A). Si elle est vérifiée, un poids de 1 (tout ou rien) est cumulé dans DA_{habilité}. Le rapprochement des savoirs (ou savoir-faire) d'une compétence requise (R) avec ceux d'une compétence acquise A revient au rapprochement de leurs thèmes.

Chaque thème d'un savoir (ou d'un savoir-faire) d'une compétence (R) est recherché dans l'ensemble des thèmes des savoirs (ou savoir-faire) d'une compétence acquise (A) de même type (général ou spécifique), en utilisant la formule de similarité des concepts (3). Le meilleur degré de similarité sera cumulé dans DAsavoir (ou dans DAsavoirF). Le degré de rapprochement global entre la compétence requise (R) et la compétence acquise (A) est le rapport entre la somme de (DAhabilité, DAsavoir, DAsavoirF) et le nombre de ressources qui constituent R.

La fonction SimNiveau calcule la similarité de leurs niveaux par l'application de la formule (4), alors que la fonction SimContext calcule la similarité de leurs contextes, c'est le cumul de $\alpha * SCOR$ (similarité des contextes organisationnels multiplié par α) et $\beta * SCEc$ (similarité des contextes économiques multiplié par β), les deux fonction SCOR, SCEc sont calculé par l'application de la formule (4).

Les formules suivantes sont utilisées dans l'algorithme d'appariement rapproché des compétences, la formule (4) est utilisée ainsi dans l'algorithme d'appariement partiel des compétences. Soit :

- $c_1, c_2 \in C$, C est l'ensemble de toutes les instances de type « Compétence scientifique/technique »
- P : la relation « mobilise » qui relie une compétence c_i à un ensemble de ressources R .
- $A_S(P, c_i)$ [MAE02] : la fonction qui retourne l'ensemble des instances de type « Ressource » mobilisées par la compétence c_i , tel que :

$$A_S(P, c_i) = \{r_j : r_j \in R \wedge P(c_i, r_j)\}. \quad (1)$$

$F(c_1, c_2, P)$ [MAE02] est une fonction qui calcule la similarité de deux compétences en se basant sur la relation P « mobilise » qui les relie aux instances d'un autre concept « Ressource ». Soit $r_j \in A_S(P, c_1)$, et $b \in A_S(P, c_2)$ et $N = |A_S(P, c_1)|$: le nombre d'instances de cet ensemble.

$$F(c_1, c_2, P) = \frac{\sum_{j=1}^N \max \{ \text{sim}(r_j, b) / b \in A_S(P, c_2) \}}{N} \quad (2)$$

La fonction *sim* concerne dans notre cas le calcul de la similarité entre deux ressources (r_j, b) de type « Ressource », pouvant être de type spécifique : « Savoir », « Savoir-faire » ou « Habilité ».

Si le type spécifique de r_j est différent de celui de b alors $\text{sim}(r_j, b)=0$ (on ne peut pas comparer un savoir à une habilité par exemple)

Si r_j et b ont le même type spécifique « Habilité » alors $\text{sim}(r_j, b)=1$ Sinon (r_j et b sont tous les deux des savoirs ou des savoir-faire)

$t_1 := r_j.a\text{Thème}$ ayant le type T_1 et $t_2 := b.a\text{Thème}$ ayant le type T_2 (T_1 et T_2 sont des sous-classes de « Thème ») si le type spécifique de t_1 est différent de celui de t_2 («ThèmeGeneral » ou « ThèmeSpécifique ») alors $\text{sim}(r_j, b) = 0$

Sinon appliquer la formule (3) [BERN04] :

$$\text{Sim}(r_j, b) = (2 * \log \text{PROF}(\text{MRCA}(T_1, T_2))) / (\log \text{PROF}(T_1) + \log \text{PROF}(T_2)) \quad (3)$$

Sachant que :

- $\text{PROF}(T_1)$ est la profondeur du thème T_1 dans la hiérarchie des thèmes de « ThèmeGeneral » ou « ThèmeSpécifique ». $\text{PROF}(T_1) = (\text{nombre des sous classes de } T_{1+1}) / \text{nombre de classes de la hiérarchie du concept « ThèmeGeneral » ou « ThèmeSpécifique »}$.
- $\text{MRCA}(T_1, T_2)$ est l'ancêtre commun le plus proche entre T_1 et T_2 dans la hiérarchie de thèmes.

Pour le calcul de la similarité (SimNiveau) de deux niveaux de compétence n_1 et n_2 nous utilisons la formule (4) [BIZE05] : α est le nombre des niveaux, il est égal à 5 pour les compétences scientifique/technique et à 2 pour les aptitudes.

Pour le calcul de la similarité (SCEc) de deux contextes économiques ctx_1 et ctx_2 , on utilise la même formule avec $\alpha=6$ le nombre de contexte économique. Même chose pour le contexte organisationnel (SCOr).

$$= \begin{cases} 1 - 1/\alpha (a_1 - a_2) & \text{Si } a_1 - a_2 \geq 0 \\ 1 & \text{Si } a_1 - a_2 < 0 \end{cases} \quad (4)$$

Pour le rapprochement des compétences nous proposons dans ce qui suit la formule (5), sur laquelle base l'algorithme de rapprochement des compétences RapprochéComp :

$$\begin{aligned}
 & \text{RaprocheComp}(\text{Compétence}_{\text{offre}}, \text{Compétence}_{\text{CV}}) = \\
 & \left(\sum_{i \in I} \text{poids}(\text{Comp}_{\text{offre}}^i) * \text{Max}[\text{sim}(\text{Comp}_{\text{offre}}^i, \text{Comp}_{\text{CV}}^j) * \text{SimContext}(\text{Comp}_{\text{offre}}^i, \text{Comp}_{\text{CV}}^j)] * \right. \\
 & \left. \text{SimNiveau}(\text{Comp}_{\text{offre}}^i, \text{Comp}_{\text{CV}}^j) / j \in J \right) / \sum_{i \in I} \text{poids}(\text{Comp}_{\text{offre}}^i) \quad (5)
 \end{aligned}$$

Telles que :

$$\S \text{ Sim}(\text{Comp}_{\text{offre}}^i, \text{Comp}_{\text{CV}}^j) = \begin{cases} F(\text{Comp}_{\text{offre}}^i, \text{Comp}_{\text{CV}}^j, P) & \text{compétences scientifiques/ techniques} \\ 1 & \text{aptitude ont le même nom} \\ 1/4 & \text{aptitude ont la même catégorie} \\ 0 & \text{sinon} \end{cases}$$

Sim est le degré de similarité (rapprochement) de deux compétences. Pour des compétences scientifiques ou techniques, la similarité est calculée par l'application de la formule (2). Pour les aptitudes, elle est égale à 1 s'ils ont le même nom et 1/4 (c'est une valeur subjective, les experts du domaine seuls qui peuvent la valider) s'ils ont la même catégorie sinon elle est égale à 0.

§ *SimContext*($\text{Comp}_{\text{offre}}^i, \text{Comp}_{\text{CV}}^j$) = ($\alpha * \text{SCOr}$) + ($\beta * \text{SCEc}$) la similarité des contextes de deux compétences est le cumul de degré de similarité des contextes organisationnels ($\alpha * \text{SCOr}$) et le degré de similarité des contextes économiques ($\beta * \text{SCEc}$). *SCOr* et *SCEc* : sont calculés par l'application de la formule (4).

§ *SimNiveau* : la similarité de deux niveaux de compétence est calculée aussi par l'application de la formule (4).

§ *poids*($\text{Comp}_{\text{offre}}^i$) : est le poids de la compétence annotée par l'utilisateur.

La procédure RapprochéComp :

Cet algorithme utilise certains éléments définis auparavant qui sont : *CompétenceOffre*, *CompétenceCv*, *PComp*, $\text{DA}_{\text{compétence}}$ *Totalemen*, et d'autres :

SavoirOf : les ressources de type savoir de l'offre.

SavoirCv : les ressources de type savoir de CV.

DAsavoir : degré d'appariement des savoir.

SavoirFOf : les ressources de type savoir faire de l'offre.

SavoirFCv : les ressources de type savoir faire de CV.

DASavoirF : degré d'appariement des savoir-Faire.

HabilitéOf : les ressources de type habilité de l'offre.

HabilitéCv : les ressources de type habilité de CV.

DAhabilité : degré d'appariement des habilités.

D'autres variables simples sont utilisées pour des résultats intermédiaires.

Dans l'écriture de l'algorithme d'appariement RapprochéComp, nous allons utiliser les procédures et les fonctions suivantes :

§ La procédure « ExtraireComp » : permet d'extraire les compétences depuis un fichier des annotations.

§ La fonction « NBRe » : retourne le nombre de ressources d'une compétence.

§ La fonction « Nbr » : retourne le nombre des concepts dans la hiérarchie des thèmes (Héra).

§ La procédure « ExtraireSavoir » : permet d'extraire tous les savoir sous-jacents à une compétence de l'offre d'emploi dans SavoirOf ou d'un CV dans SavoirCv. Elle est détaillée ci-dessous :

ExtraireSavoir(compcv,compoffre,SavoirOf,SavoirCv)

SavoirOf := {R ∈ {compOf.requis} / Savoir (R)}

/* extraire les savoirs d'une compétence d'offre d'emploi*/

SavoirCv := {R ∈ {compCv.requis} / Savoir (R)}

/* extraire les savoirs d'une compétence d'un CV*/

Fin

§ La procédure « ExtraireSavoirF » : permet d'extraire tous les savoir-faire sous-jacents à une compétence de l'offre d'emploi dans SavoirFaireOf ou d'un CV dans SavoirFaireCv.

Elle est détaillée ci-dessous :

ExtraireSavoirF(compcv,compoffre,SavoirFOF,SavoirFCv)

SavoirOf := {R ∈ {compOf.requis} / SavoirFaire (R)}

/* extraire les savoir-faire d'une compétence d'offre d'emploi*/

SavoirCv := {R ∈ {compCv.requis} / SavoirFaire (R)}

/* extraire les savoir-faire d'une compétence d'un CV*/

Fin

§ La procédure « ExtraireHabilité » : permet d'extraire toutes les habilités sous-jacentes à une compétence de l'offre d'emploi dans HabilitéOf ou d'un CV dans HabilitéCv. Elle est détaillée ci-dessous :

ExtraireHabilité(comp_{cv},comp_{offre},Habilité_{Of}, Habilité_{Cv})
 Savoir_{of} := {R ∈ {comp_{of}.requis} / Habilité (R)}
 /* extraire les habilités d'une compétence d'offre d'emploi*/
 Savoir_{Cv} := {R ∈ {comp_{Cv}.requis} / Habilité (R)}
 /* extraire les habilités d'une compétence d'CV*/
Fin

§ La procédure « Rapprochement » : permet de calculer le degré d'appariement des Savoir et le degré d'appariement des savoir-Faire d'une compétence d'un CV avec celle d'une offre emploi. Elle est détaillée ci-dessous :

Rapprochement (Ressource_{of}, Ressource_{cv},DAR)
 SR_I: tableau des degrés de rapprochements des thèmes des ressources.
Pour chaque Rof ∈ Ressource_{of} **répéter**
 I := 0 ; SR_I := ∅. /* Rof, Rcv sont respectivement une ressource de compétence de L'offre et une ressource de compétence de CV*/
Si themespécifique(Rof.atheme) **alors** /*pour chaque thème spécifique*/
Pour chaque Rcv ∈ Ressource_{cv} et themespécifique(Rcv.atheme) **répéter**
 SR_I := Rapproch (Rof, Rcv, Hthemespécifique)
 /*le calcul le degré de rapprochement de Rof, Rcv*/
 I := I + 1
Fin répéter
Si non /*pour chaque thème général*/
Pour chaque Rcv ∈ Ressource_{cv} et themegénéral(Rcv.atheme) **répéter**
 SR_I := Rapproch (Rof, Rcv, Hthemesgénéral)
 /*le calcul le degré de rapprochement de Rof, Rcv */
 I := I + 1
Fin répéter
Fin si
 DAS := DAS + Max (SR_I) /*le degré d'appariement des thèmes des ressources d'une compétence de l'offre avec les thèmes des ressources d'une compétence du CV */
Fin répéter
Fin

Dans l'écriture de cette procédure nous allons utiliser la procédure «Rapproch» qui permet de calculer la similarité de deux thèmes dans une hiérarchie des thèmes par le biais de la formule (3), dont PROF est la profondeur d'un concept, MRCA est l'ancêtre commun le plus proche de deux concepts. «Rapproch» est détaillée comme suit :

Raproch (Rof, Rcv, Hthemes)Stop: = \emptyset

/* calcule la similarité de deux thèmes dans la hiérarchie des thèmes*/

Raproch: = $2 * \log \text{PROF} (\text{MRCA} \{ \text{classe} (\text{Rof.atheme}), \text{classe} (\text{Rcv.atheme}) \}, \text{Hthemes}, \text{stop}) / \log \text{PROF} (\{ \text{classe} (\text{Rof.atheme}) \}, \text{Hthemes}, \text{stop}) + \log \text{PROF} (\{ \text{classe} (\text{Rcv.atheme}) \}, \text{Hthemes}, \text{stop})$ **Fin**

La procédure « Raproch » utilise les deux fonctions suivantes. Elles sont détaillées comme suit :

1- La fonction « PROF » : permet de calculer la profondeur d'un concept dans une hiérarchie.

§ PROF (C, Héra, stop)**F** := $\{ F_i, i \geq 0 / F_i \rightarrow c_k, c_k \in C \} \cup C$

/*F est l'union des subsumes précédant C et des nouveaux subsumes d'un concept*/

Si | Stop | \neq | F | **alors**Stop: = Stop \cup F

PROF (F, Héra, stop) /*Recursive*/

Sinon

PROF := | F | + 1 / Nbr (Héra) /*pas de nouveau subsumé*/

Stop := \emptyset

/* nombre de subsumés du concept +1 divisé par le nombre de concepts dans la hiérarchie*/

Fin si**Fin**

2- La fonction « MRCA » : retourne l'ancêtre commun le plus proche de deux concepts dans une hiérarchie de concept (l'intersection des généralisations transitives des deux concepts).

§ MRCA (A, B)

/*subsumé d'un concept ou ensembles de concept*/

X: = $\{ x_i, i \geq 0 / x_k \rightarrow x_i, x_k \in A \}$

/* subsumé d'un concept ou ensembles de concept */

Y: = $\{ y_i, i \geq 0 / y_k \rightarrow y_i, y_k \in B \}$

/*l'intersection est l'ensemble vide*/

Si X \cap Y = $\{ \}$ **alors**

MRCA (X, Y) /*récursivité*/

Sinon**Si** X = Y **alors**

MRCA: = {A}

SinonMRCA := X \cap Y /* intersection des subsumés des deux concepts*/**Fin si****Fin si****Fin**

-Les procédures « SimContext » et « SimNiveau » ont été détaillées précédemment.

L'algorithme de l'appariement Rapproché des compétences est détaillé comme suit :

```

RapprochéComp (FAOffre, FACv, DAcompétence)
  DAI : tableau.
  CompétenceOffre := ∅ /* les compétences de l'offre */
  CompétenceCv := ∅ /* les compétences de Cv */
  PComp := 0 /* cumul des poids des compétences vérifiés*/
  TotalComp := 0 /* cumul des poids de toutes les compétences requises */
  ExtraireComp(FAOffre, CompétenceOffre), ExtraireComp(FACv, CompétenceCv)
  Pour chaque compoffre ∈ compétenceoffre répéter
    /* pour chaque compétence de l'offre*/ I := 0, DA := ∅,
    TotalComp := TotalComp + compoffre.poids
  Si compétence Technique/scientifique (compoffre) alors
    /*pour chaque compétence de CV*/
    Pour chaque compcv ∈ compétenceCV répéter
      /* extraire les savoir d'une compétence de CV et une compétence de l'offre*/
      ExtraireSavoir(compcv,compoffre,SavoirOf,SavoirCv)
      /* calculer le degré de rapprochement des savoir d'un CV avec les savoir d'une offre*/
      Rapprochement (SavoirOf,SavoirCv,DAsavoir)
      /* extraire les savoir-faire d'une compétence de CV et une compétence de l'offre*/
      ExtraireSavoirF(compcv,compoffre,SavoirFOf,SavoirFCv)
      /* calculer le degré de rapprochement des savoir-faire d'un CV avec les savoir-faire d'une offre*/
      Rapprochement (SavoirFOf,SavoirFCv, DAsavoirF)
      /* extraire les habilités d'une compétence de CV et une compétence de l'offre*/
      ExtraireHabilité(compcv,compoffre,HabilitéOf, HabilitéCv)
      /*calculé le degré de rapprochement des habilités d'un CV avec les habilités d'une offre*/
      Pour chaque Hof ∈ HabilitéOf répéter
        Si ∃ Hcv ∈ HabilitéCv / Hof = Hcv alors /*égalité des deux habilités*/
          DAhabilité := DAhabilité + 1
        Fin si
      Fin répéter /*le degré de rapprochement de deux compétences scientifique/technique*/
      DAI := ((DAsavoir + DAsavoirF + DAhabilité) / NBRe (compoffre))
        *SimNiveau(compoffre,compcv)*SimContext(compoffre, compcv)
      I := I + 1
    Fin répéter /* NBR retourne nombre de ressources d'une compétence*/
    PComp := PComp + ( compoffre.poids)* Max (DAI)
  Sinon
    /* le degré de rapprochement de deux aptitudes.*/
  Si Compoffre.nom = Compcv.nom alors
    PComp := PComp + compoffre.poids * SimNiveau(compoffre,compcv) * SimContext (compoffre,
    compcv)
  Sinon
    Pour chaque compcv ∈ CompétenceCv répéter
      Si AptitudManagériale(compcv) = AptitudManagérial (compoffre) ou
      ActionRéalisation(compcv) = ActionRéalisation(compoffre) ou
      EfficacitéPersonnelle(compcv) = EfficacitéPersonnelle(compoffre) ou
      AssistanceEtService(compcv) = AssistanceEtService(compoffre) alors
        DAI := SimNiveau(compoffre,compcv) * SimContext (compoffre, compcv)
        I := I + 1
      Finsi
    Fin répéter
    PComp := PComp + (compoffre.poids)* Max (DAI) *1/4
  Finsi
  Fin si
  Fin répéter
  DAcompétence := (PComp / TotalComp) *100
Fin

```

7.4.4 Discussion sur les algorithmes d'appariements proposés :

Pour le choix d'un CV le recruteur se concentre généralement sur les compétences scientifiques ou techniques ou les aptitudes, mais dans certains cas les qualifications personnelles telles que l'âge, sexe, nationalité, permis de conduire, service national, ville de résidence ainsi que certaines exigences telles que le salaire, le type de contrat, le type d'emploi, rotation, déplacement peuvent écarter un candidat bien que ses compétences soient satisfaisantes dans l'offre d'emploi.

Pour prendre en charge ces différents cas, ce qui n'est pas fait par les travaux existants, nous avons proposé des techniques d'appariement sémantique entre les CVs et les offres d'emploi qui prennent en charge ces trois aspects, L'appariement de qualification personnelle prend en charge la correspondance entre les qualifications personnelles requises dans offres d'emploi et celles acquises par un candidat (CV); Quant à l'appariement des exigences, il permet d'assurer une correspondance entre les exigences tolérées par le recruteur et le chercheur d'emploi. Pour les compétences, vu leurs pertinences pour la sélection d'un candidat, nous avons proposé trois techniques d'appariement : l'appariement exact qui est appliqué dans le cas où le recruteur cherche un candidat qui possède exactement les compétences requises par lui. L'appariement partiel est appliqué si aucun candidat n'est sélectionné par l'appariement exact, alors le recruteur diminue ses contraintes. L'appariement rapproché est appliqué dans le cas où l'appariement partiel ne donne pas de résultats satisfaisant le recruteur.

L'apport des techniques d'appariement proposé est qu'ils prennent en considération le choix de l'utilisateur en donnant la main à ce dernier de pondérer chaque acquis ou requis avec un poids annoté par lui selon son choix. Ainsi pour les compétences nous avons proposé trois techniques différentes permettant au recruteur à chaque fois de diminuer dans ses contraintes jusqu'à trouver des CVs qui le satisfassent en compétence même partiellement. De plus pour le rapprochement des documents d'un point de vue de la compétence, nous avons introduit des notions concernant le contexte organisationnel et économique dans lequel cette compétence est acquise ou requise, afin que les résultats du processus d'appariement basé compétence soient les plus faibles possibles.

8. Conclusion

Dans ce chapitre, nous avons proposé une modélisation pertinente du contenu sémantique à base d'ontologie des CVs et des offres d'emploi, axée sur la modélisation de la compétence. Le modèle adopté de la compétence est riche parce qu'il tient compte du type de la compétence (scientifique et technique ou aptitude), des ressources relatives aux compétences scientifiques (habilité, savoir, savoir-faire), de la catégorie relative aux aptitudes (managériales, assistance et service, ...), du niveau d'expérience et des contextes organisationnels et économiques.

Nous avons proposé ainsi un processus d'annotation simple qui se base sur l'exploitation et l'instanciation de l'ontologie-ER via un outil d'annotation [ALLI08]. Ce processus permet d'enrichir les documents avec des méta-données, ces dernières sont exploitées par les algorithmes d'appariement originaux que nous avons proposés. Les algorithmes d'appariement proposé permettent aussi bien le rapprochement basé sur les qualifications personnelles et les exigences, que celui basé sur les compétences. Vu l'importance des compétences pour la sélection d'un chercheur d'emploi, nous avons proposé trois services d'appariement des compétences : appariement exact des compétences, appariement partiel des compétences, appariement rapproché des compétences. Le résultat final d'appariement est une liste de quintuplés (URI, DA_{final} , $DA_{\text{QPersonnel}}$, DA_{Exigence} , $DA_{\text{compétence}}$) trié par le degré d'appariement final (DA_{final}) suivi par les trois autres degrés ($DA_{\text{compétence}}$, $DA_{\text{QPersonnel}}$, DA_{Exigence}) leurs ordre de tri est en fonction du choix de l'utilisateur.

Dans le chapitre suivant nous détaillons le processus de développement de l'ontologie-ER.

Chapitre 4

Construction de l'ontologie-ER

Sommaire

1. Introduction
 2. Processus de construction d'une ontologie
 - 2.1 Spécification
 - 2.2 Conceptualisation
 - 2.3 Formalisation.
 - 2.4 Implémentation.
 - 2.5 Evolution de l'ontologie et Test.
 3. Construction d'une ontologie pour le e-recrutement
 - 3.1 Spécification
 - 3.2 Conceptualisation
 - 3.2.1 Construction du glossaire de termes.
 - 3.2.2 Construction des hiérarchies de concepts.
 - 3.2.3 Construction du diagramme de relations binaires.
 - 3.2.4 Dictionnaire de concepts.
 - 3.2.5 Tableau des relations binaires.
 - 3.2.6 Tableau des attributs.
 - 3.2.7 Tableau des axiomes logiques.
 - 3.2.8 Les instances des concepts et les assertions sur les instances.
 - 3.3 Formalisation
 - 3.3.1 Construction de TBOX
 - 3.3.2 Construction de ABOX
 - 3.4 Implémentation.
 - 3.4.1 Création d'un nouveau projet OWL
 - 3.4.2 Création des classes et la hiérarchie des classes
 - 3.4.3 Création des propriétés
 - 3.4.4 Création des instances
 - 3.4.5 Génération de code
 - 3.4.6 Test et évolution
 4. Conclusion
-

« Une petite colline te fait arriver à une grande »

Proverbe africain

1. Introduction

L'objectif de ce chapitre est de décrire le processus de construction de l'ontologie-ER dédiée au e-recrutement, dont le domaine est « Informatique et Télécommunication ». Pour ce faire, nous basons sur la méthodologie METHONTOLOGY [FERN97] qui est le support de base pour la conceptualisation de notre ontologie. La logique de description est le formalisme adopté pour l'expression de l'ontologie semi-formelle, résultat de la phase de conceptualisation. Basé sur cette formalisation, OWL [PATE03] est le langage de définition d'ontologie est choisi afin de codifier l'ontologie en utilisant l'éditeur d'ontologies PROTEGE OWL¹ [RECT04]. RACER² [HAAR03] est le système d'inférences utilisé afin de tester la consistance de l'ontologie tout au long son processus de développement.

2. Processus de construction d'une ontologie

Nous décrivons par la suite le processus que nous avons adopté dans le développement de l'ontologie-ER (une ontologie de domaine). Les grandes étapes de ce processus sont inspirées de la méthodologie de construction d'ontologies « Methontology » [FERN97]. L'application de chacune des étapes de ce processus est basée sur l'exploitation du travail de HEMMAM [HEMA05]. On part dans le processus de connaissances brutes et on arrive à une ontologie pour le e-recrutement, opérationnelle représentée par le langage OWL. Ce processus est composé de cinq étapes :

- Spécification des besoins.
- Conceptualisation.
- Formalisation.
- Implémentation.
- Evolution de l'ontologie et Test.

1: <http://protégé/standford/edu/plugins/owl>

2: <http://www.sts.tu-harburg.de/~r.f.moeller/racer/>

2.1 Spécification

Cette étape consiste à établir un document formel de spécification des besoins. Ce dernier permet de décrire l'ontologie à construire à travers les cinq aspects suivants :

1. **Le domaine de connaissance** : déterminer aussi précisément que possible le domaine que va couvrir l'ontologie.
2. **L'objectif** : le but de l'ontologie à créer pour le domaine considéré.
3. **Les utilisateurs** : identifier au maximum les futurs utilisateurs de l'ontologie à créer.
4. **Les sources d'informations** : déterminer les sources d'informations d'où les connaissances seront obtenues. Exemple, les experts du domaine, les documents techniques, etc.
5. **La portée de l'ontologie** : déterminer à priori la liste des termes (les plus importants) pour le domaine à représenter.

2.2 Conceptualisation :

L'étape de conceptualisation est la plus importante dans le processus de construction d'ontologie présenté par METHONTOLOGY. Elle est inspirée de la méthodologie qui consiste à identifier et à structurer, à partir des sources d'informations, les connaissances du domaine. Elle permet d'aboutir à un ensemble de représentations intermédiaires semi-formelles indépendamment des langages de formalisations à utiliser pour représenter l'ontologie par la réalisation des tâches suivantes :

1. Construction du glossaire de termes.
2. Construction du diagramme de classification des concepts.
3. Construction du diagramme de relations binaires et de classification des concepts.
4. Construction du dictionnaire de concepts (DC).
5. Décrire les relations dans une table de relations binaires.
6. Spécifier des contraintes sur les attributs dans une table d'attributs.
7. Spécifier des axiomes sur les concepts dans une table d'axiomes logiques.
8. Décrire les instances des concepts dans une table d'instances et les assertions dans une table des assertions.

A la fin de cette phase, nous obtenons une ontologie conceptuelle.

2.3 Formalisation

Cette phase consiste à formaliser l'ontologie conceptuelle obtenue dans l'étape précédente afin de faciliter sa représentation ultérieure dans un langage complètement formel et opérationnel. Notre choix est porté sur le formalisme de représentation de la logique de description.

Elle permet de représenter les connaissances relatives à un domaine de référence à l'aide de "descriptions" qui peuvent être des concepts, des rôles et des individus. Les concepts modélisent des classes d'individus et les rôles des relations entre classes. Une sémantique est associée aux descriptions par l'intermédiaire d'une fonction d'interprétation. La relation de subsomption permet d'organiser les concepts et les rôles en hiérarchie. La classification et l'instanciation sont les opérations qui sont à la base du raisonnement sur les descriptions, ou raisonnement terminologique.

Il existe de nombreux constructeurs permettant de former toute une famille de logiques de description. Par ailleurs, la logique de description SHIQ [HORR02] regroupe un ensemble plus riche de constructeurs. Le tableau (Tableau.4.1) décrit les plus importants, ainsi que leur

Constructeur	Syntaxe	Sémantique (I : une interprétation)
Universel	\top	ΔI (ΔI : ensemble de tous les objets)
Absurde	\perp	\emptyset
Négation	$\neg C$	$\Delta I \setminus CI$
Conjonction	$C \cap D$	$CI \cap DI$
Disjonction	$C \cup D$	$CI \cup DI$
Restriction universelle	$\forall x. C$	$\{x \in \Delta I / \forall y, (x, y) \in R_i \Rightarrow y \in CI\}$
Restriction existentielle	$\exists x. C$	$\{x \in \Delta I / \exists y, (x, y) \in R_i \text{ et } y \in CI\}$
Cardinalité minimum	$(\geq n \ z)$	$\{x \in \Delta I / \{y (x, y) \in R_i\} \geq n\}$
Cardinalité maximum	$(\leq n \ z)$	$\{x \in \Delta I / \{y (x, y) \in R_i\} \leq n\}$
Conjonction des rôles	$r \cap s$	$\{(x, y) \in \Delta I \times \Delta I / (x, y) \in rI \wedge (x, y) \in sI\}$
Disjonction des rôles	$r \cup s$	$\{(x, y) \in \Delta I \times \Delta I / (x, y) \in rI \vee (x, y) \in sI\}$

Tableau 4.1- Les constructeurs essentiels d'une logique de description.

sémantique en s'appuyant sur la syntaxe de SHIQ [PATL03], qui présente une logique de description très expressive et qui offre un certain nombre de constructeurs des axiomes,

représentés dans la figure 4.1, pour exprimer le passage du modèle conceptuel (ontologie conceptuelle) à l'ontologie formalisée en LD.

Rôles	
Atomique	R
Inverse	R^{-}

Concepts	
Universel/absurde	T / \perp
Atomique	A, B
Non	$\neg C$
Et	$C \sqcap D$
Ou	$C \sqcup D$
Existe	$\exists R.C$
Pour tous	$\forall R.C$
Au moins	$\geq n R.C$ ($\geq n R$)
Au plus	$\leq n R.C$ ($\leq n R$)
énumération	$\{i_1, \dots, i_n\}$

Axiomes assertionnels (A_Box)	
Instance	$C(a)$
Rôle	$R(a,b)$
même	$a = b$
différents	$a \neq b$

Axiomes de Rôle (R_Box)	
Sous rôle	$R \sqsubseteq S$
transitivité	Trans (S)

Axiomes de concep $C \equiv D$	
Sous-classe	$C \sqsubseteq D$
Equivalence	

Figure 4.1 - Syntaxe de la logique de description SHOIN

L'ontologie formelle résultante est constituée d'une partie terminologique (TBOX) où sont introduits les concepts, et d'une partie (ABOX) où sont introduites les assertions de concepts et de rôles.

La partie terminologique TBOX comprend des définitions et des organisations formelles des concepts de l'ontologie. Cela est fait par l'utilisation des axiomes de la logique de description, décrits dans ce qui suit :

$\langle \text{Concept} \rangle^{\otimes} \langle \text{concept-primitif} \rangle \mid \langle T \rangle \mid \langle \perp \rangle \mid$ $\langle \text{Concept} \rangle \dot{\subset} \langle \text{Concept} \rangle \mid \langle \text{Concept} \rangle \dot{\supset} \langle \text{Concept} \rangle \mid$ $\neg \langle \text{Concept} \rangle \mid " \langle \text{role} \rangle . \langle \text{Concept} \rangle \mid \$ \langle \text{role} \rangle . \langle \text{Concept} \rangle \mid$ $\text{^} n \langle \text{role} \rangle . \langle \text{Concept} \rangle \mid \text{& n} \langle \text{role} \rangle . \langle \text{concept} \rangle \mid$
--

La partie assertionnelle ABOX, définit les individus, les rôles en fonction des concepts qui l'utilisent, par l'utilisation de l'une des formes suivantes :

a : C où C est un concept défini et a est un individu.
(a1, a2) : R où R est un rôle défini et a1, a2 sont deux individus définis.

2.4 Implémentation

Cette étape consiste à traduire le résultat de la phase précédente (ontologie formelle), dans un langage opérationnel (OWL) de définition d'ontologie. Afin de faciliter le processus de codification, nous utilisons PROTEGE OWL version 3.1.1 disposant d'une interface permettant l'édition, la visualisation, le contrôle (vérification des contraintes) d'ontologies [NOY00], issu du modèle des frames et contenant des classes (concepts), des slots (propriétés) et des facettes (valeurs des propriétés et contraintes), ainsi que des instances des classes et des propriétés.

2.5 Évolution et test

Cette étape sert aussi à suivre l'évolution de l'ontologie, c'est-à-dire les nouveaux concepts à ajouter dans sa partie terminologique (TBOX). Une classification a lieu chaque fois qu'une définition de concept est nouvellement créée. Le mécanisme de raisonnement de base des logiques de description est la classification de concepts. Elle est réalisée par un algorithme de classification, appelé « le classifieur ». Le classifieur utilise la description d'un nouveau concept pour le placer à l'endroit correspondant dans la hiérarchie. Afin de trouver la place appropriée au nouveau concept, l'algorithme de classification détermine les relations de subsomption entre ce concept et les autres. Ces relations peuvent être spécifiées directement, trouvées par transitivité ou calculées à partir de la sémantique des conditions des rôles. La recherche de la place correcte pour le nouveau concept comporte trois étapes :

1. La recherche des subsumants les plus spécifiques SPS (concepts qui subsument le concept à classer et dont les fils ne le subsument pas)

2. La recherche des subsumés les plus généraux SPG (concepts subsumés par le concept à classer et dont les pères ne sont pas subsumés par lui).

3. Insertion du nouveau concept dans la hiérarchie.

Le *TEST* consiste à exploiter les services d'inférence fournis par la logique de description afin d'améliorer la qualité de l'ontologie. Pour ce faire, l'outil Racer est choisi [HAAR01] car c'est le moteur d'inférence le plus utilisé dans ce domaine pour ses performances et sa stabilité. Racer travaille sur les ontologies modélisées par son langage, mais il accepte aussi celles décrites en RDF ou OWL, ces dernières étant traduites vers le langage utilisé par Racer. Ce moteur d'inférence possède également son propre langage de requête RQL (Racer query Language) pour interroger les ontologies sur la ABox et la TBox. Se système se connecte facilement à l'outil PROTEGE-OWL

3. Construction d'une ontologie pour le e-recrutement

Dans cette section, nous construisons notre ontologie-ER. A cette fin, nous suivrons les étapes du processus de construction d'ontologie décrite dans la section 2.

3.1 Spécification

Pour commencer le développement de l'ontologie, nous entamons d'abord la phase de spécification qui consiste à établir un document de spécification des besoins.

Au sein de ce document, nous dériverons l'ontologie à construire à travers les cinq aspects suivants:

- ***Le domaine de connaissance*** : le domaine de l'ontologie-ER est « l'Informatique et la Télécommunication ».
- ***L'objectif*** : c'est la modélisation du contenu sémantique des documents (CVs/offres d'emploi) et principalement les compétences sous-jacentes aux métiers et aux diplômes.

Cette conceptualisation partagée sera utilisée par l'utilisateur afin d'annoter son document avec ses requis (ou acquis). Ces annotations seront utilisées pour assurer un rapprochement sémantique et automatique ultérieur entre les offres et les demandes d'emploi.

- **Les utilisateurs** : cet aspect présente l'ensemble des utilisateurs qui pouvant exploiter l'ontologie. Dans notre cas, ce sont les recruteurs et les chercheurs d'emploi. Ils utilisent l'ontologie-ER pour annoter leurs documents avec les éléments de cette dernière.
- **Les sources d'informations** : les sources d'informations sur lesquelles nous nous sommes basées, sont : les prototypes des ontologies des travaux voisins, les sites d'emploi, les référentiels de métiers et de diplômes existants, les sites dédiés à la rédaction des CVs (tels que Europass), etc.
- **La portée de l'ontologie** : Cet aspect détermine à priori la liste des termes de l'ontologie-ER (les plus importants) parmi lesquels, nous pouvons citer : métier, diplôme, compétence, contexte, savoir, savoir faire, salaire, rotation, déplacement, contrat, sexe, permis de conduire, âge, type de l'emploi, etc.

Quant aux scénarios d'usage les plus importants et les questions auxquelles l'ontologie est sensée fournir des réponses, nous citons :

- ▶ Quelles sont les compétences sous-jacentes à un métier particulier et leurs types ?
- ▶ Quelles sont les compétences sous-jacentes à un diplôme particulier et leurs types ?
- ▶ Quelle est la famille d'un métier ?
- ▶ Quelle est la famille d'un diplôme ?
- ▶ Quels sont les diplômes d'un domaine particulier ?
- ▶ Quels sont les métiers d'un domaine particulier ?
- ▶ Quel est le domaine d'un référentiel particulier ?
- ▶ Quelles sont les ressources d'une compétence ?

Le résultat de cette étape est le document de spécification illustré dans la figure 4.2.

- **Domaine** : documents CVs et offres d'emploi reliés à la spécialité « Informatique et Télécommunications ».
- **Développé par** : Boucetta zouhel
- **Objectif opérationnel** (But) : la modélisation du contenu sémantique des documents (CVs/offres d'emploi), principalement les compétences sous-jacentes ou métiers et ou diplômes.
- **Utilisateurs** : les recruteurs et les chercheurs d'emploi.
- **Degré de formalisme** : formel
- **Porté (Liste des termes importants)**: Métier, Diplôme, compétence, contexte, , savoir, savoir faire , salaire, rotation,déplacement, contrat, sexe, permis de conduire, âge,type d'emploi, etc.
- **Granularité** (niveau de détail) : fine.
- **Sources de connaissances** : *Autres ontologies, portail d'emploi, analyse de documents, référentiel existants de métiers et de diplômes, interviews, etc.*

Figure.4.2- Le document de spécification

3.2 Conceptualisation

Une fois que la majorité des connaissances est acquise, on doit les organiser et les structurer en utilisant des représentations intermédiaires semi-formelles qui sont faciles à comprendre et indépendantes de tout langage d'implémentation. Cette phase contient plusieurs étapes qui sont :

- Construction du glossaire de termes.
- Construction du diagramme de classification de concepts.
- Construction du diagramme de relations binaires.
- Dictionnaire de concepts.
- Construction du tableau des relations binaires.
- Construction du tableau des attributs.
- Construction du tableau des axiomes logiques.
- Construction du tableau des instances et du tableau des assertions.

3.2.1 Construction de glossaire de termes

Ce glossaire contient la définition de tous les termes relatifs au domaine (concepts, attributs, relations) qui seront représentés dans l'ontologie finale.

Le tableau ci-dessous (Tableau 4.2) fournit seulement une liste de quelques termes utilisés dans l'ontologie:

<i>CONCEPT</i>	<i>DESCRIPTION</i>
Référentiel Métiers	Un référentiel des Métiers du domaine «Informatique et Télécommunication»
Métier (job, travail)	Décrit les métiers du domaine «Informatique et Télécommunication»
FamilMétier	Décrit les familles de métiers du domaine « Informatique et Télécommunication»
Compétence	Décrit tout type de compétences
Aptitude (compétence comportementale)	Décrit les compétences comportementales
Compétence Scientifique Technique	Décrit les compétences scientifiques et techniques
Domaine	Décrit le domaine du référentiel des métiers ou du référentiel des diplômes.
Qualification Personnelle	Décrit les qualifications personnelles acquises ou requises
ServiceMilitaire	Décrit la situation envers le service militaire (dégagé ou non).
Age	décrit l'âge, acquis (CV) ou requis (offre d'emploi)
Sexe	décrit le sexe, acquis (CV) ou requis (offre d'emploi)
PermisConduire	Décrit la catégorie du permis de conduire requis par un recruteur ou possédé par un chercheur d'emploi.

CONCEPT	DESCRIPTION
document	Décrit le document à annoter (CV/offre d'emploi) à travers son URI et son type.
Contexte	Décrit le contexte d'une compétence.
Economique	Décrit le contexte économique d'une compétence.
Organisationnel	Décrit le contexte organisationnel d'une compétence.
Diplôme (certificat, attestation)	Décrit les diplômes valides du domaine «Informatique et Télécommunication»
FamilDiplôme	Décrit les familles de diplômes du domaine « Informatique et Télécommunication»
Référentiel Diplômes	Décrit le référentiel des diplômes du domaine «Informatique»
Ressource	Décrit les ressources d'une compétence.
Habilité	Décrit les habilités d'une compétence.
SavoirFaire	Décrit les savoirs faire d'une compétence.
Savoir	Décrit les savoirs d'une compétence.
Thème	Décrit le thème d'un savoir ou d'un savoir faire
ThèmeGénéral	Décrit le thème Général d'un savoir ou d'un savoir faire
Salaire	Décrit le salaire toléré par le recruteur ou le chercheur d'emploi
Contrat	Le type de contrat toléré par le recruteur ou le chercheur d'emploi
Thème Spécifique	Décrit le thème spécifique d'un savoir ou d'un savoir faire
VilleRésidence	Décrit la ville de résidence d'un chercheur d'emploi, ou celle requise par un recruteur
Nationalité	Décrit la nationalité d'un chercheur d'emploi, ou celle requise par le recruteur
(...)	(...)

Tableau.4.2- La table du glossaire de termes

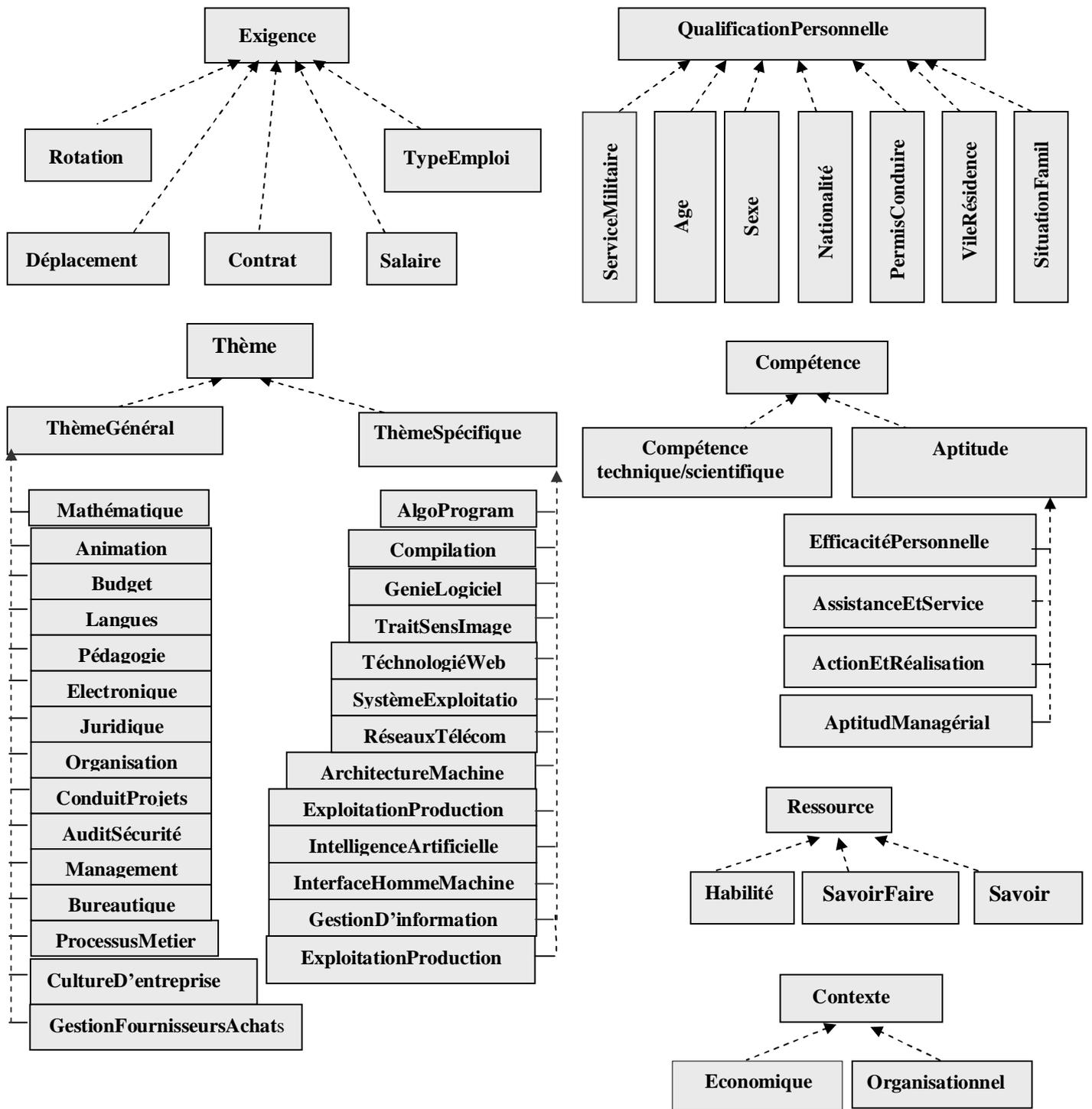
3.2.2 Construction des hiérarchies de concepts

Dans cette étape, nous construisons les diagrammes de classification de concepts. La construction des hiérarchies de concepts de l'ontologie-ER est basée principalement sur l'analyse des CVs et des offres d'emploi et leurs éléments communs les plus significatifs, de certains documents techniques ou référentiels ([CIGR05], [LMD04], [YAHIO6a]), de l'analyse d'autres ontologies ([YAHIO6A], [SCHM06a], [TRIC02], IT-Skill³). Initialement, nous répertorions les concepts en ensemble d'organisations, ensuite nous relierons les concepts entre eux, si nécessaire, par des relations « Est-un ». Les différentes hiérarchies de classification de concepts démontrent l'organisation des concepts de l'ontologie en un ordre hiérarchique qui exprime la relation « Est-un ». La relation « Est-un » entre les classes définit que, la classe C1 est une sous classe de la classe C2, si et seulement si toute instance de la classe C1 est une instance de la classe C2. La figure 4.3 représente les différentes hiérarchies de concepts construits.

Un concept universel « Thing », qui généralise tous les concepts, racines des différentes hiérarchies de concepts, peut être utilisé pour former une seule hiérarchie globale, afin d'éviter d'avoir des concepts isolés. La figure 4.4 représente la hiérarchie globale de tous les concepts de l'ontologie-ER.

La hiérarchie du concept « Thème » ne peut être entièrement représentée par un graphe car elle est volumineuse, mais elle est représentée sous forme de tableaux en *Annexe A*. La hiérarchie du concept « ThèmeGénéral », est inspirée du référentiel de [CIGR05] enrichi par d'autres concepts. La hiérarchie du concept « ThèmeSpécifique », est inspirée des référentiels [LMD04], [CIGR05], [ABRA04], [YAHIO6a] et l'ontologie IT-Skill et des avis des experts.

3: <http://www.wiwiss.fu-berlin.de/suhl/radek/semmf>

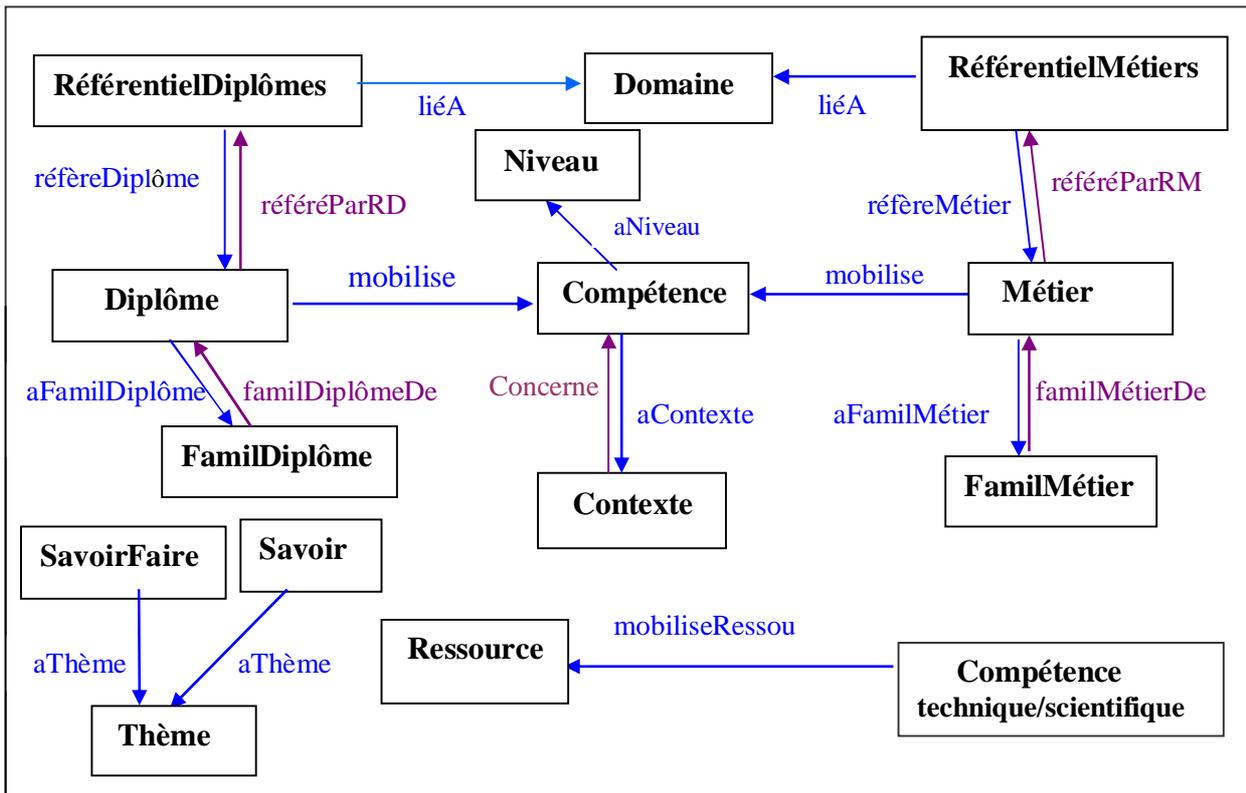


-----> Est un

Figure 4.3 - Les hiérarchies de concepts

3.2.3 Construction du diagramme des relations binaires

Dans cette étape, nous construisons le diagramme des relations binaires. Dans ce diagramme, les classes sont représentées par des rectangles et les relations par des arcs orientés (du domaine vers le co-domaine) et étiquetés par le nom de la relation. La relation inverse d'une relation est aussi schématisée dans ce diagramme. La figure 4.5 illustre le diagramme des relations binaires.



La figure 4.5- Le diagramme des relations binaires.

3.2.4 Dictionnaire de concepts

Dans cette étape nous allons assurer une description formelle des concepts qui ont été présentés dans la hiérarchie des classes. Ce processus correspond à la création du dictionnaire de concepts accordé au METHONTOLOGY. Dans ce dictionnaire, nous définissons pour chaque concept : les attributs, les relations dont la source est ce concept, le père. Le tableau 4.3 représente le dictionnaire des concepts de l'ontologie-ER.

<i>CONCEPT</i>	<i>ROLE</i>	<i>PARENT</i>	<i>ATTRIBUTS</i>
document	-	Thing	typDocument, uriDocument
Contexte	-	Thing	poids, nom, expAnglais, valeurMin, valeurMax
Economique	-	Contexte	poids, nom, expAnglais, valeurMin, valeurMax
Organisationnel	-	Contexte	poids, nom, expAnglais, valeurMin, valeurMax
Diplôme (certificat, attestation)	mobilise	Thing	nom, expAnglais, expFrançais
FamilDiplôme	-	Thing	nom, expAnglais
Nationalité	-	QualificationPersonnelle	poids, nationalité
Ressource	-	Thing	nom
Habilité	-	Ressource	nom
SavoirFaire	aThème	Ressource	nom
Savoir	aThème	Ressource	nom
Thème	-	Thing	nom
ThèmeGénéral	-	Thème	nom
Thème Spécifique	-	Thème	nom
Salaire	-	Exigence	poids, montant, maxDif
Contrat	-	Exigence	poids, type
TypeEmploi	-	Exigence	poids, type
(...)	(...)	(...)	(...)

Tableau 4.3 – Dictionnaire de concepts de l'ontologie-ER

3.2.5 Tableaux de relations binaires

Les relations binaires sont représentées sous forme de propriétés qui lient un concept à un autre. Pour chaque relation dont la source est dans l'arbre de classification de concepts, nous définissons : son nom, sa description, le nom du concept source, le nom du concept cible, la cardinalité et le nom de la relation inverse. Le tableau 4.4 représente la table des relations binaires de l'ontologie-ER.

RELATION	DESCRIPTION	CONCEPT (S) SOURCE (S)	CONCEPT (S) CIBLE (S)	CARDINALITÉS		RELATION INVERSE
				S	C	
aThème	Relie un savoir ou un savoir faire avec son thème.	Savoir SavoirFaire	Thème	1. N	1. N	
aFamil Diplôme	Relie un diplôme à une famille de diplômes	Diplôme	FamilDiplôme	1. N	1	familD iplôme De
aFamil Métier	Relie un métier à une famille de métiers	Métier	FamilMétier	1. N	1	famil Métier De
liéA	Relie un référentiel de diplômes ou un référentiel de métiers à son domaine	RéférentielDi-- plômes RéférentielMétiers	Domaine	1. N	1	
mobilise	Relie un métier ou un diplôme à un ensemble de compétences	Métier Diplôme	Compétence	1. N	1. N	
réfèreMétier	Relie un référentiel de métiers à l'ensemble des métiers qu'il définit	Référentiels -Métier	Métier	1. N	1..N	référé ParR M
(...)	(...)	(...)	(...)	(...)	(...)	(...)

Tableau 4.4- La table des relations binaires de l'ontologie-ER.

3.2.6 Tableaux des attributs

Les attributs sont des propriétés qui prennent leurs valeurs dans les types prédéfinis (String, Integer, Boolean, Date...). Pour chaque attribut nous spécifions : son nom, description, les concepts qu'il contient, son type et l'intervalle de ses valeurs possibles, sa cardinalité, sa valeur par défaut. Le tableau 4.5 représente la table des attributs des concepts de l'ontologie-ER.

<i>ATTRIBUT</i>	<i>DESCRIPTION</i>	<i>CONCEPT(s)</i>	<i>TYPE</i>	<i>Car d</i>	<i>Val - Déf</i>	<i>Dom_ Vals</i>
age	Age du candidat, ou age exigé par un recruteur	Age	Entier>0	1		
maxDif	La différence maximale acceptée entre le salaire requis et celui toléré par le chercheur d'emploi (ou vice versa)	Salaire	Entier>0	1		
maxDA	La distance maximale acceptée entre la ville requise et celle de résidence du chercheur d'emploi (ou vice versa).	VilleRésidence	Entier>0	1		
marié	- L'état civil du candidat cas D'un CV (célibataire ou marié) - L'état civil requis par le recruteur (célibataire ou marié)	SituationFamil	Chaîne de caractère	1		Oui, non
Identification	Identifier les référentiels	RéférentielDiplômes, RéférentielMétiers	Chaîne de caractère	1		
(...)	(...)	(...)	(...)	(..)	(.)	(..)

Tableau 4.5- La table des attributs de concepts de l'ontologie-ER.

3.2.7 Tableau des axiomes logiques

Ce tableau contient des définitions de concepts à l'aide des expressions logiques qui sont toujours vraies. Dans ce tableau nous définissons pour chaque axiome sa description en langage naturel, le nom du concept auquel l'axiome se réfère et l'expression logique. On ne peut pas citer tous les axiomes. Le tableau 4.6 représente un extrait de cet ensemble.

<i>CONCEPT</i>	<i>AXIOME</i>	<i>DESCRIPTION</i>
Ressource	$\forall (X), \text{Ressource}(X) \Rightarrow \text{Savoir}(X) \vee \text{SavoirFaire}(X) \vee \text{Habilité}(X)$	Une ressource est un savoir ou un savoir faire ou une habilité.
Compétence	$\forall (X), \text{Compétence}(X) \Rightarrow \text{Aptitude}(X) \vee \text{CompétenceScientifique Technique}(X)$	Une compétence est soit comportementale soit, scientifique ou technique.
Thème Spécifique	$\forall (X), \text{ThèmeSpécifique}(X) \Rightarrow \text{Thème}(X) \wedge (\neg \text{Thème Général}(X))$	Les concepts ThèmeSpécifique et ThèmeGénéral sont incompatibles
Aptitude	$\forall (X), \text{Aptitude}(X) \Rightarrow \text{Compétence}(X) \wedge (\neg \text{CompétenceScientifiqueTechnique}(X))$	Les concepts Aptitude et CompétenceScientifique ou - Technique, sont incompatibles
Thème	$\forall (X), \text{Thème}(X) \Rightarrow \text{ThèmeGénéral}(X) \vee \text{Thèmespécifique}(X)$	Chaque thème est soit un thème général, soit un thème spécifique.
Contexte	$\forall (X), \text{Contexte}(X) \Rightarrow \text{Economique}(X) \vee \text{Organisationnel}(X)$	Chaque contexte est soit économique, ou organisationnel.
Exigence	$\forall (X), \text{Exigence}(X) \Rightarrow \text{Rotation}(X) \vee \text{Déplacement}(X) \vee \text{TypeEmploi}(X) \vee \text{Contrat}(X) \vee \text{Salaire}(X)$	Chaque exigence est soit, rotation, soit déplacement, soit type emploi, soit contrat, soit salaire.
(...)	(...)	(...)

Tableau 4.6- La table des axiomes de concepts de l'ontologie-ER

3.2.8. Les instances des concepts et les assertions sur les instances

Les concepts et les relations de l'ontologie-ER sont instanciés à l'avance par l'expert du domaine, afin de fournir les différentes instances, dont l'utilisateur (recruteur ou chercheur d'emploi) peut se servir durant l'annotation de son document.

Les instances et les assertions liées aux concepts relatifs aux métiers (Métier, FamilMétier, compétence) et diplôme (Diplôme, FamilDiplôme, compétence) sont inspirées respectivement de [CIGR05], [LMD04] enrichi.

Dans cette section nous allons présenter une description de quelques instances de l'ontologie. Pour cela, nous spécifierons les noms des individus et les valeurs des attributs pour chacun d'eux. Le Tableau.4.7 illustre quelques instances pour chaque classe, quelques assertions sur ces instances sont présentées dans la table des assertions (Tableau 4.8).

<i>CONCEPT</i>	<i>INSTANCE</i>	<i>ATTRIBUTS</i>	<i>VALEURS</i>
Économique	eco_graEnt	poids expAnglais, nom	6 Great Company Grande Entreprise
Domaine	dom_infTélé	nom, expAnglais	Informatique et Télécommunication computers and Télécommunication
Organisationnel	org_graSer	poids expAnglais, nom valeurMin valeurMax	6 small group petit groupe 5 14
Compétence	comp_traEqui	nom expAnglais, expFrançais	TraEqui Team work Travail en équipe
Diplôme	dip_ingInflnD	nom expFrançais expAnglais	Ingénieur ID Ingénieur en Informatique Industrielle Engineer Computers Industrial
(..)	(...)	(...)	(...)

Tableau 4.7- La table des instances de concepts de l'ontologie-ER.

<i>RELATION</i>	<i>INSTANCE SOURCE</i>	<i>INSTANCE(S) CIBLE(S)</i>
aFamildiplôme	dip_ masterSI	FD_ master
aFamilmétier	mét_ adminBD	FM_ produExploi
Mobilise	mét_ adminBD	assi_ pédagogie
aContexte	comp_ java	eco_ graEnt
liéA	RM_ réfémétinforma	dom_ infTélé
mobiliseRessou	comp_ java	hab_ raisonner
(...)	(...)	(...)

Tableau 4.8- La table des attributs de concepts de l'ontologie-ER.

3.3 Formalisation

Dans cette étape, nous utilisons le formalisme des logiques de description pour formaliser le modèle conceptuel que nous avons obtenu dans l'étape de conceptualisation.

3.3.1 Construction de TBOX

Les définitions des différents concepts sont illustrées dans le tableau 4.9 où nous définissons les concepts relatifs à notre domaine, en utilisant les constructeurs fournis par les logiques de descriptions pour donner des descriptions structurées aux concepts. Nous spécifions ainsi dans le tableau 4.9 les relations de subsomption qui existent entre les différents concepts.

Pour les rôles, nous les définissons en donnant les couples des concepts sources et cibles de chacune, et/ou en spécifiant son rôle inverse. Le tableau 4.10 représente les définitions de quelques rôles de notre ontologie.

<i>CONCEPT</i>	<i>DEFINITION</i>	<i>RELATION DE SUBSOMPTION</i>
SavoirFaire	Ressource \cap (= 1 nom .String) \cap (=1 aTheme.Thème)	SavoirFaire \subseteq Ressource
Déplacement	Exigence \cap (= 1 accepté .String) \cap (=1 poids.Int)	Déplacement \subseteq Exigence
Métier	(=1 nom.String) \cap (=1 expFrançais.String) \cap (=1 expAnglais.String) \cap (=1 aFAMILMétier. FamilMétier) \cap (\geq 1 mobilise. Compétence)	Métier \subseteq thing
RéférentielDiplômes	(=1 nom.String) \cap (= 1 expFrançais.String) \cap (=1 Identification .String) \cap (\geq 1 auteur .String) \cap (=1 liéA.domaine) \cap (\geq 1 réfèreDiplôme. diplôme).	RéférentielDiplômes \subseteq thing
(...)	(...)	(...)

Tableau 4.9- Définitions des concepts de l'ontologie-ER (dans TBOX).

<i>ROLES</i>	<i>COUPLE (DOMAINE, CO-DOMAINE)</i>	<i>ROLE INVERSE</i>
aFAMILMétier	(Métier, FamilMétier)	familMétierDe
réfèreDiplôme	(RéférentielMétiers, Métier)	référéParRM
RéfèreDiplôme	(RéférentielDiplômes, Diplôme)	référéParRD
aFAMILDiplôme	(Diplôme, FamilDiplôme)	familDiplômeDe
liéA	(RéférentielDiplômes, Domaine)	-
mobiliseRessou	(CompétenceScientifiqueTechnique, Ressource)	-
mobilise	(Diplôme, Compétence)	-
mobilise	(Métier, Compétence)	-
aContexte	(Compétence, Contexte)	conceme
(...)	(...)	(...)

Tableau 4.10- Définitions des rôles de l'ontologie-ER (dans TBOX).

3.3.2 Construction de ABOX

Le langage assertionnel est dédié à la description des faits, en spécifiant les individus (avec leurs classes) et les relations entre eux de la manière suivante :

A : C

Pour dire que A est une instance de la classe C.

(A1, A2) : R

Pour dire que les deux individus A1 et A2 sont reliés par la relation R.

Dans les tableaux (tableau 4.11, tableau 4.12) nous définissons quelques assertions :

<i>CONCEPT</i>	<i>DESCRIPTION</i>
Contexte	eco_graEnt : Contexte, eco_moyEnt : Contexte, eco_petEnt : Contexte. eco_graEco: Contexte, eco_univ : Contexte...
Habilité	hab_raison : Habilité, hab_mémoris : Habilité, ...
Domaine	dom_infTélé : Domaine ...
Organisationnel	org_graSer : Organisationnel, org_petSer : Organisationnel org_moySer : Organisationnel...
Diplôme	dip_licProgl : Diplôme, dip_licSI : Diplôme, dip_ingSIA : Diplôme, dip_docRéseau: Diplôme, dip_docSys : Diplôme, dip_ingSPD : Diplôme.
(...)	(...)

Tableau 4.11- Description assertionnel de concepts.

<i>RELATION</i>	<i>DESCRIPTION</i>
aFAMILMétier	(mét_ adminBDD , FM_ produExploi) : aFAMILMétier ; ...
aFAMILDiplôme	(dip_ masterSI , FD_ master) : aFAMILDiplôme ; ...
mobilise	(mét_ adminBDD , assi_ pédagogie): mobilise ; ...
aContexte	(com_java, eco_graEnt): aContexte ; ...
liéA	(RM_réféMétInforma , dom_infTélé) : liéA; ...
(...)	(...)

Tableau.4.12- Description assertionnel de relations.

Nous avons attribué un modèle conceptuel à l'ontologie-ER dans cette section. Dans la section suivante, nous détaillons son implémentation en utilisant éventuellement les outils qui ont été développés pour cet objectif.

3.4. Implémentation

L'implémentation consiste à traduire l'ontologie formelle de l'étape de formalisation ou d'évolution, dans un langage opérationnel de définition d'ontologies. Notre choix porte sur le langage OWL pour la codification de l'ontologie, et cela pour toutes les fonctionnalités sémantiques que permet OWL et qui sont plus riches que celles de DAML+OIL. D'autre part, OWL est considéré comme un langage standard de définition des ontologies pour le Web. Par ailleurs, OWL fait partie du formalisme des logiques de descriptions. Sa sémantique peut être définie via une transcription vers la logique de descriptions SHIQ.

3.4.1 Création d'un nouveau projet OWL

Lors du premier démarrage de « protégé », une boîte de dialogue (*Welcome to protégé*) s'ouvre (Figure 4.6). Afin de créer un nouveau projet OWL, nous devons cliquer sur le bouton « Create New Project... » puis le bouton « Close » qui se trouvent sur cette boîte. De ce fait, une autre boîte de dialogue (Créate New Project) se crée, sur laquelle plusieurs choix sont offerts, permettant de créer de nouveaux projets OWL selon plusieurs types. Dans notre cas il s'agit d'un fichier OWL « OWL Files (.owl or .rdf) ». Une fois que ce choix est validé, l'interface de l'outil

s'affiche (Figure 4.7) permettant d'éditer, de visualiser et d'enregistrer des ontologies en OWL.

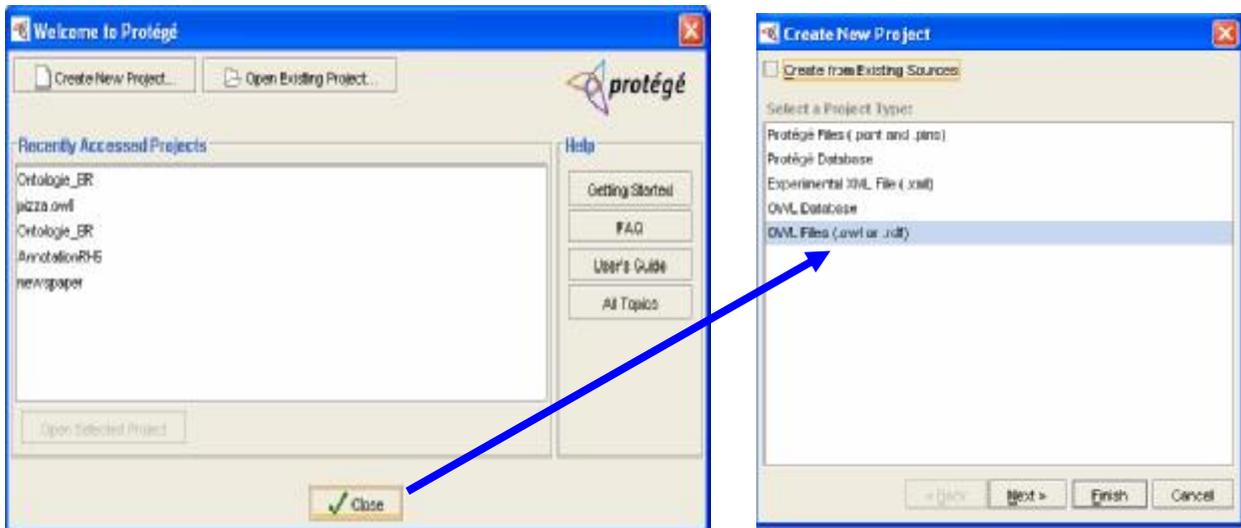


Figure 4.6- Création d'un nouveau projet OWL

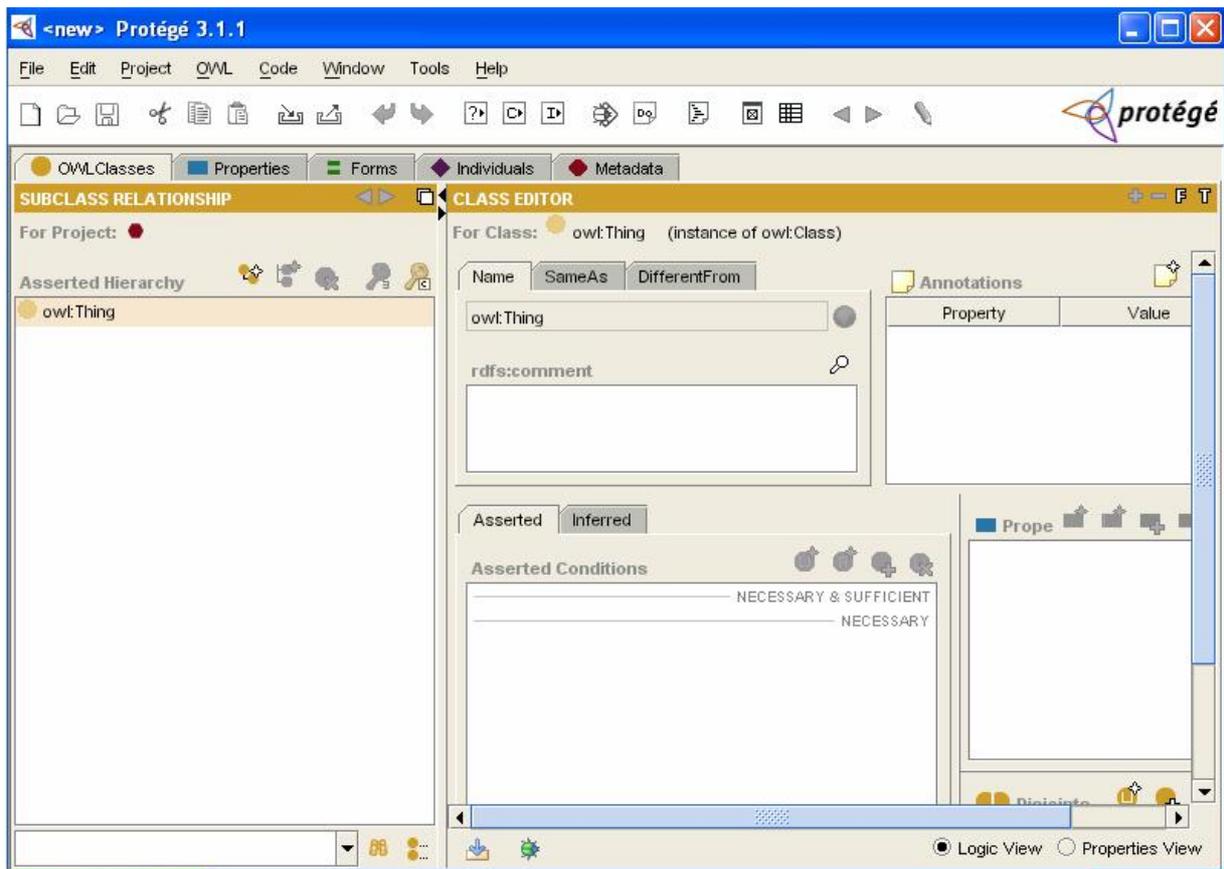


Figure 4.7- Interface de l'outil Protégé-OWL (version 3.1.1)

Comme illustrée à la Figure 4.7 l'interface utilisateur de plugin OWL contient divers onglets associés à des tâches spécifiques. Parmi ces onglets, nous présentons ceux utilisés dans l'implémentation de l'ontologie-ER.

3.4.2 Création des classes et la hiérarchie des classes

L'onglet (OWLClasse) permet de créer les classes (concepts) et la hiérarchie de classes.

Une classe universelle (`owl : thing`) est utilisée comme racine pour cette hiérarchie, et la création des sous-classes se fait par le choix de la classe mère, suivi par un simple clique sur le bouton de création des sous-classes (Figure 4.8). Dans cet onglet, les classes disposant des sous-classes apparaissant sur l'onglet sont précédées par le signe (▼), alors que les classes disposant des sous-classes qui n'apparaissent pas sur l'onglet sont précédées par le signe (▶). Ces sous-classes peuvent être montrées par un simple clique sur ce signe.

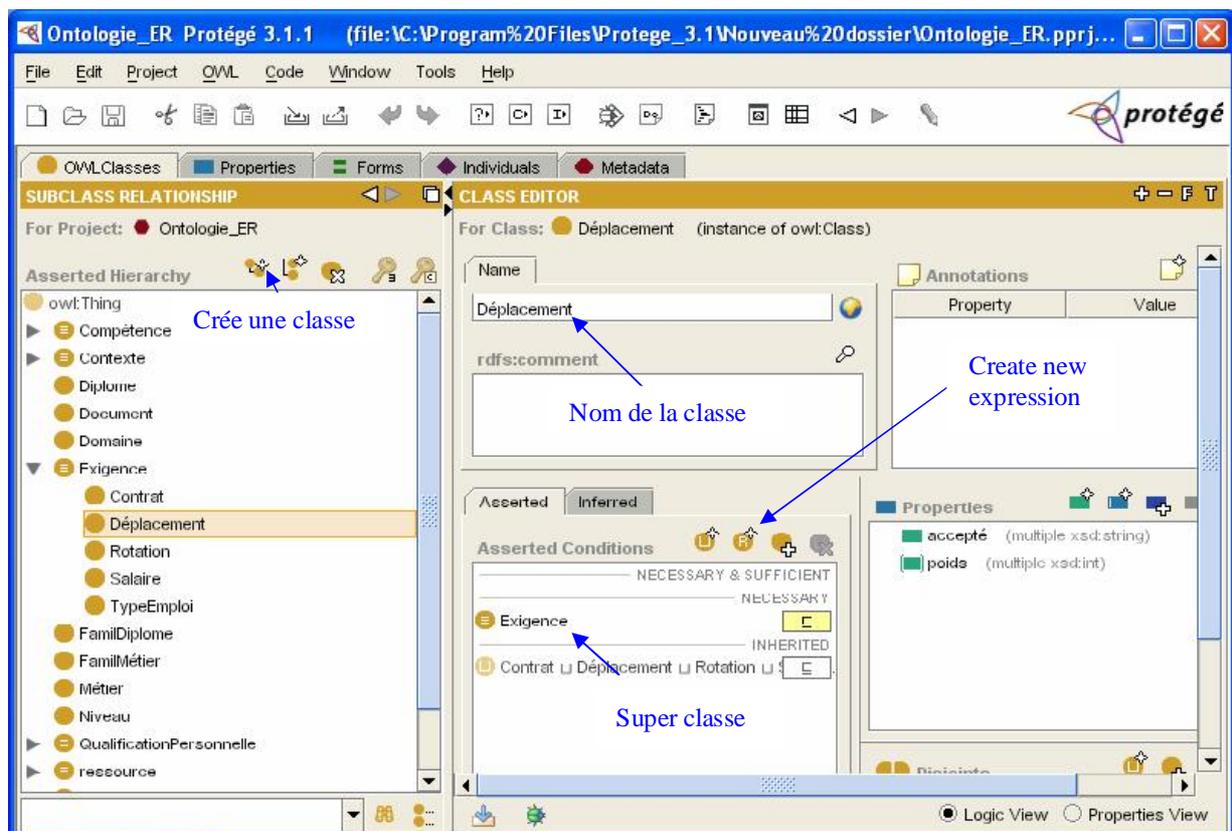


Figure 4.8- La hiérarchie des classes.

Les classes définies peuvent être introduites à travers un éditeur des expressions en logique de descriptions (Figure 4.9). Celui-ci s'affiche en cliquant sur le bouton « Create new expression » (Figure 4.8).



Figure 4.9- Editeur d'expressions en logique de descriptions

3.4.3 Création des propriétés

L'onglet (Properties) permet de créer les propriétés des classes qui peuvent être de deux types différents (attribut ou rôle). Il s'agit des attributs (datatypeProperty) de concepts et des rôles (objectProperty) qui relient les concepts.

La création d'une propriété (datatypeProperty) pour une classe particulière, consiste à choisir la classe en question, puis cliquer sur le bouton de création d'une nouvelle propriété (datatypeProperty). Les caractéristiques de cette propriété peuvent être renseignées dans une nouvelle fenêtre (Figure 4.10). Il s'agit de son domaine (rdfs : domain) et co-domaine (rdfs : range), son domaine de valeurs, un commentaire, les types de données possibles pour les attributs sont celles offertes par XMLschéma (xsd : boolean, xsd : integer, xsd : positiveInteger, xsd : float, xsd : string, xsd : decimal, ect).

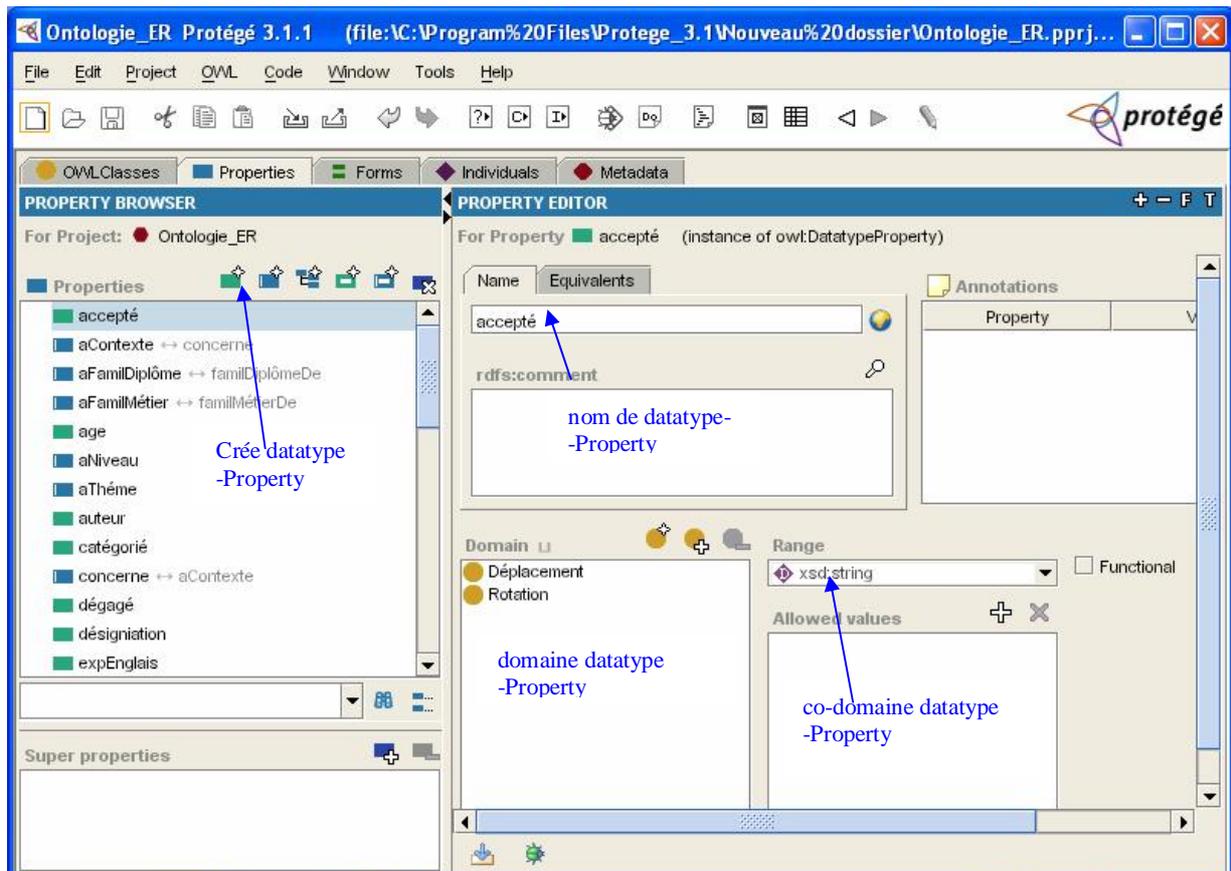


Figure 4.10- Création des propriétés (datatypeProperty) des classes.

Les propriétés (objectProperty) peuvent être hiérarchisées dans l'onglet (Properties). Les caractéristiques des propriétés peuvent être renseignées dans la fenêtre (Figure 4.11). Il s'agit de son domaine (rdfs : domain) et co-domaine (rdfs : range), la relation inverse qui lui correspond (owl : inverseOf), un commentaire, ainsi que les caractéristiques de cette propriété (fonctionnalité, fonctionnalité inverse, symétrie, transitivité) et les autres propriétés équivalentes et différentes.

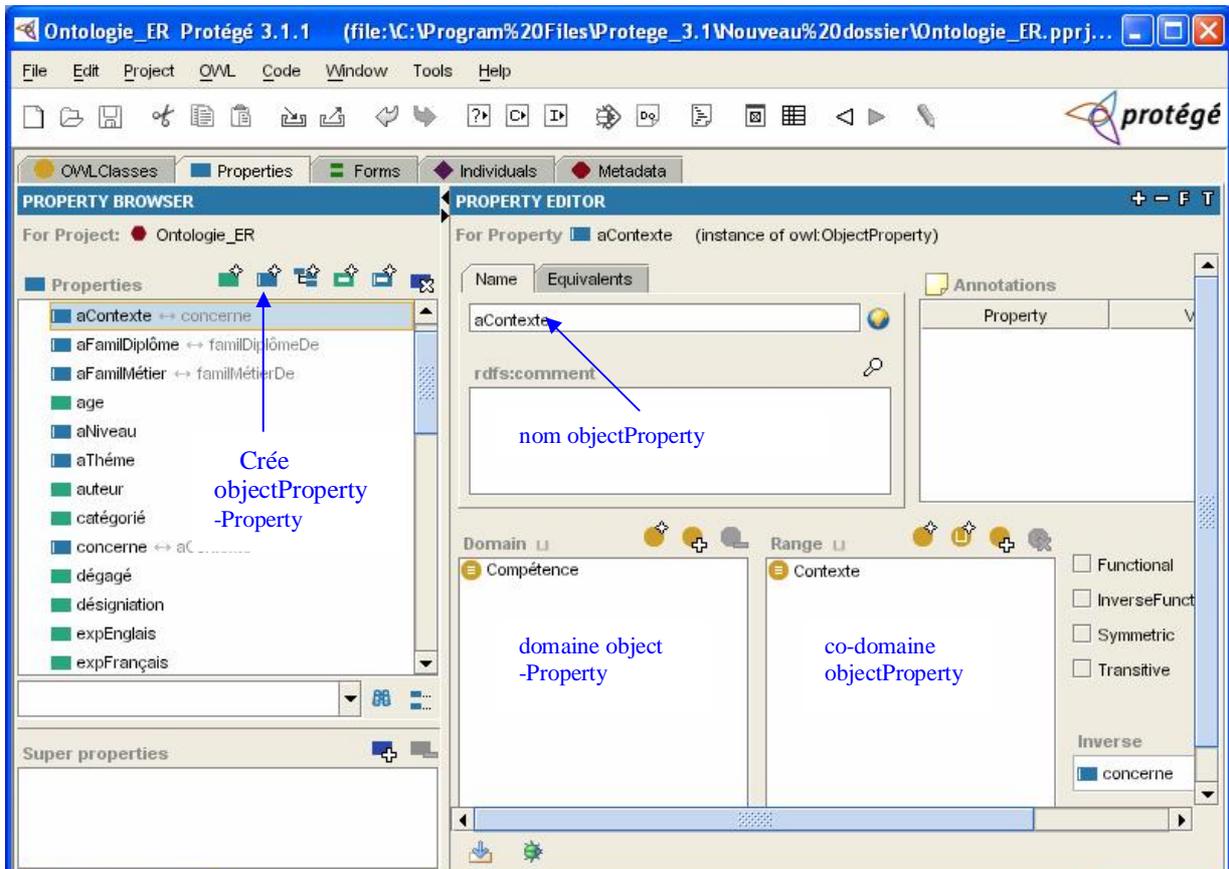


Figure 4.11- Création des propriétés (objectProperty) des classes.

Des restrictions sur les cardinalités et les valeurs possibles concernant les propriétés créées pour certaines classes peuvent être ajoutées à partir de la fenêtre (Figure 4.12). Celle-ci peut être créée en cliquant sur le bouton « Create restriction... »  et s'affiche la figure 4.13.

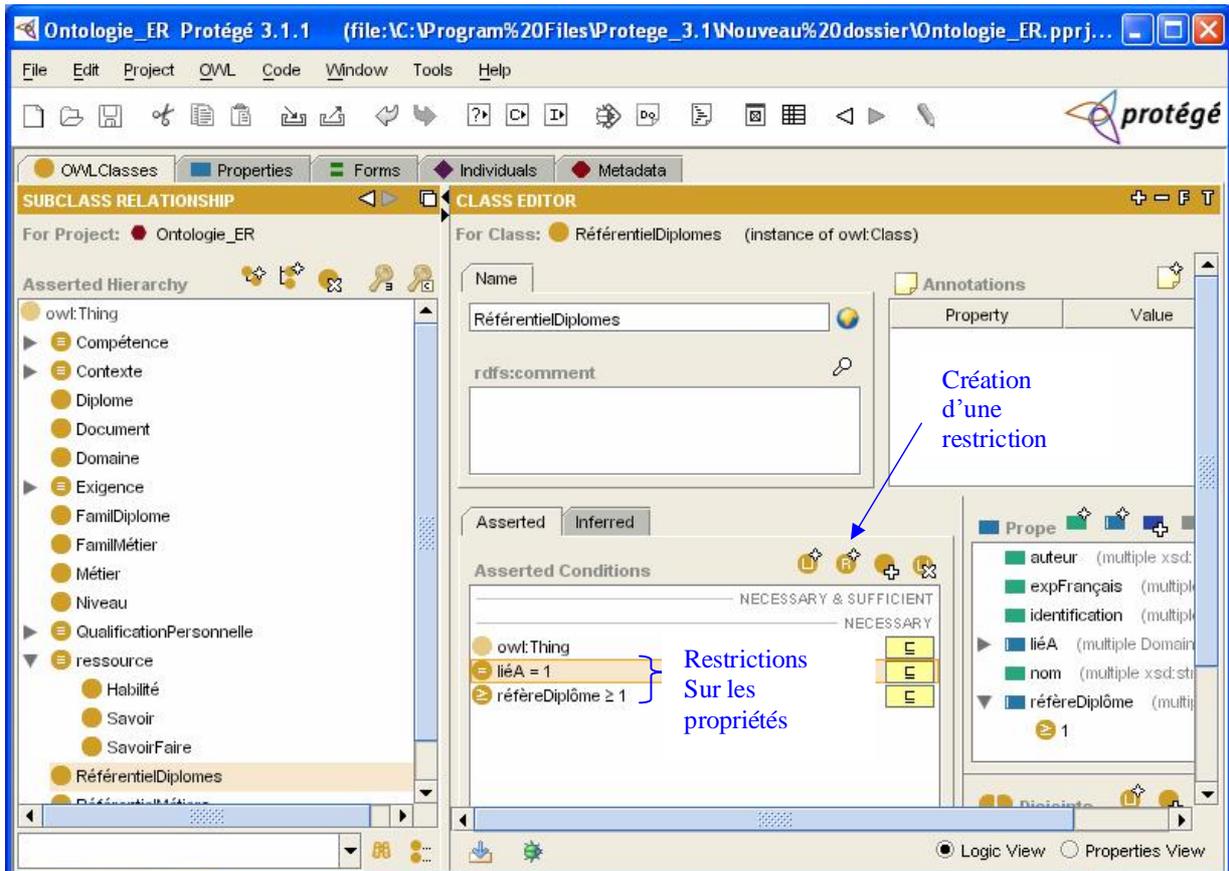


Figure 4.12- Définition des restrictions sur les propriétés.

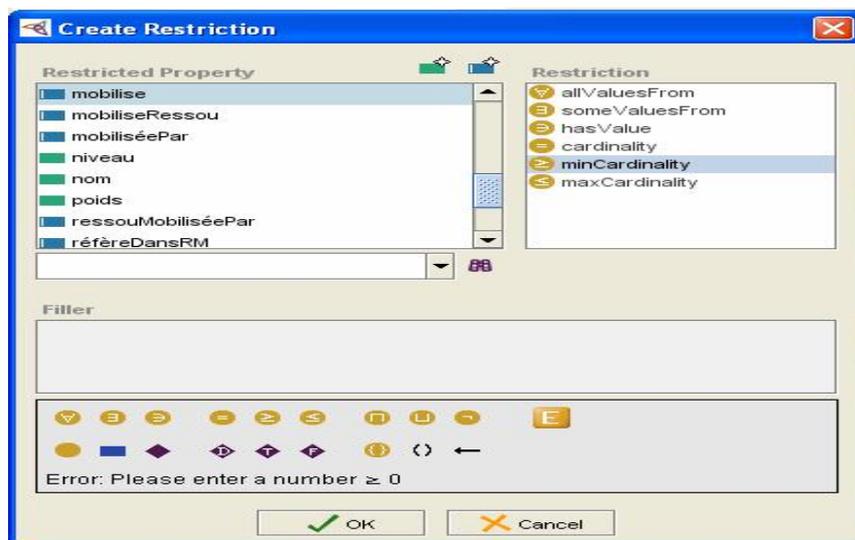


Figure 4.13- Création d'une restriction.

3.4.4 Création des instances

Les individus (instances) peuvent être créés en cliquant sur l'onglet (individuals) de l'interface de Protégé-OWL (Figure 4.14), sur lequel nous sélectionnons la classe à instancier puis nous créons l'instance. Les champs (propriétés) sont à compléter, soit par les valeurs des attributs, soit par les noms des instances avec lesquelles cette instance est reliée par un rôle.

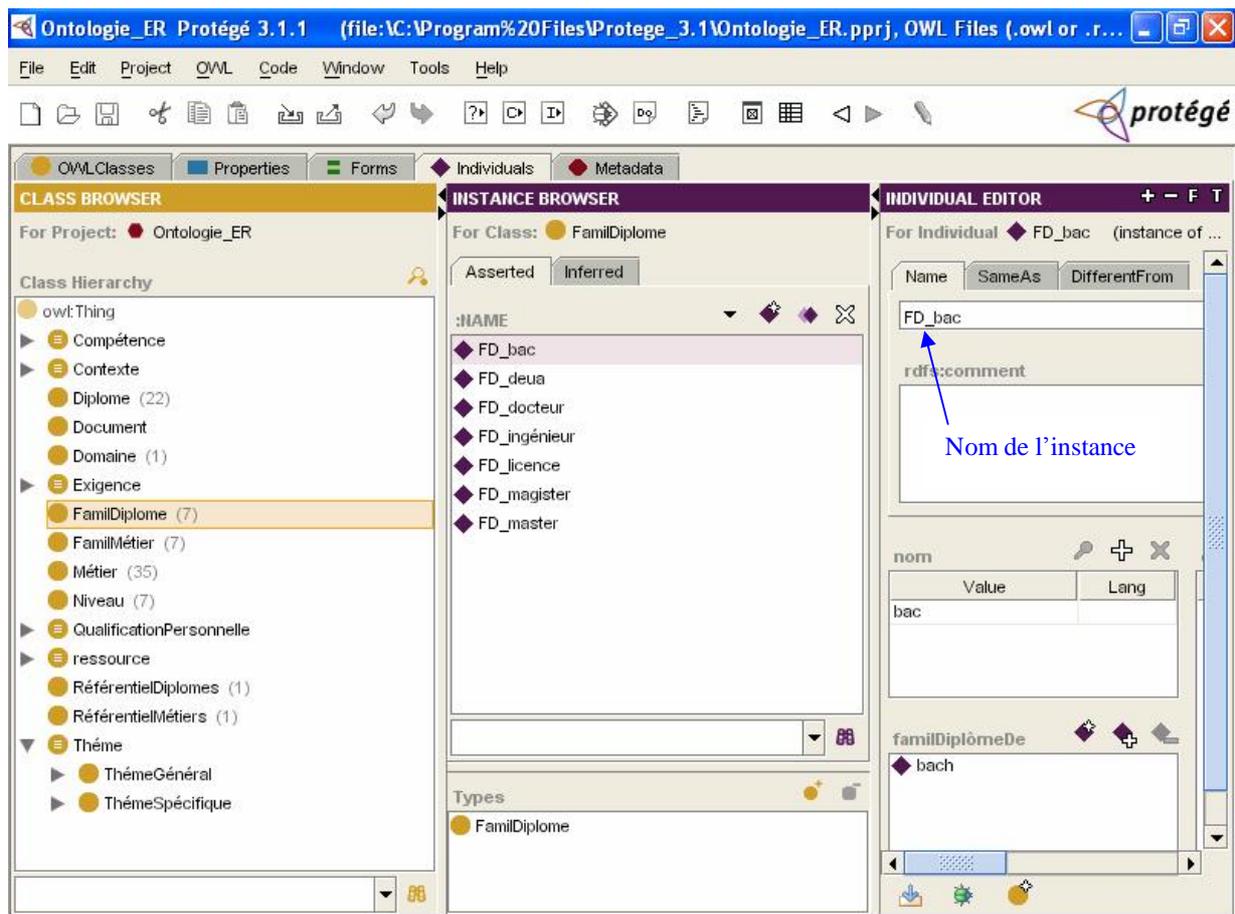


Figure 4.14- Définition des instances.

3.4.5 Génération du code

L'outil PROTEGE OWL a été conçu pour dégager et libérer le développeur de la complexité du codage, même pour implémenter une petite ontologie, celle-ci va prendre plusieurs lignes de code et nécessite un grand effort. Le code de l'ontologie-ER sera donné à l'annexe B de ce mémoire.

3.4.6 Test & évolution de l'ontologie

Comme nous l'avons indiqué auparavant, nous utilisons le raisonneur Racer pour vérifier notre ontologie. Une fois installé, celui-ci se lance par double cliquer sur son icône d'application qui ouvre une fenêtre de console, et lance le raisonneur avec une communication HTTP activée. L'interface utilisateur courante permet de tester la consistance [KNUB04] et la classification. Le test de la consistance peut être invoqué via le menu « OWL » (Figure 4.8) et s'affiche (la figure 4.15). Il consiste à déterminer si une classe peut être instanciée ou non. Le test de classification est invoqué entre les classes. Il consiste à inférer une nouvelle hiérarchie de classes, à partir des définitions de celles-ci. Il permet d'enlever l'inconsistance entre ces dernières.

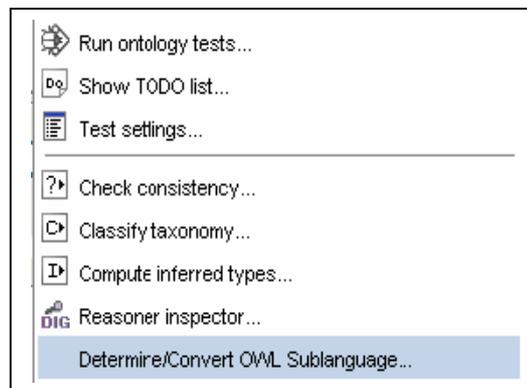


Figure 4.15- Une partie du menu «OWL » de l'interface de Protégé-OWL.

Indépendamment du raisonneur Racer, le plugin OWL, fournit un mécanisme pour exécuter des tests supplémentaires sur l'ontologie construite. Ces tests sont des petits programmes java prédéfinis (liste disponible à travers le menu « OWL », test settings), qui peuvent être étendus par les programmeurs. Ils prennent une classe, une propriété, un individu ou l'ontologie comme entrée, vérifient des conditions sur celles-ci, et retournent un message d'erreur dans le cas d'échec. Un exemple de test prédéfini est celui qui assure l'invariante «l'inverse d'une propriété fonctionnelle est une propriété fonctionnelle inverse», et vice versa. Si une propriété, dans l'ontologie a violé les conditions qui lui sont attachées, le système affiche une erreur et peut même offrir la possibilité de la corriger.

4. Conclusion

Nous avons développé l'ontologie-ER dédiée au e-recrutement. La méthode METHONTOLOGY est adoptée pour la conceptualisation de l'ontologie afin d'atteindre un ensemble de représentations intermédiaires qui facilitent sa formalisation via la logique de description SHOIN. En nous basant sur cette formalisation, nous avons choisi, le langage OWL, pour codifier l'ontologie formelle, en utilisant l'éditeur graphique PROTEGE OWL, afin de guider l'implémentation et de produire un code OWL. Par ailleurs, pour vérifier et raffiner l'ontologie OWL au cours du processus de développement, nous utilisons le système RACER. Ce dernier peut lire des représentations au format OWL, et effectuer des inférences sur le niveau terminologique et assertionnel. Les principaux services d'inférences fournis par le système RACER sont le test de consistance d'un concept « check consistency » et le test de subsomption d'un concept « classify taxonomy ».

Pour motiver notre travail d'un point de vue pratique, le chapitre suivant montre l'application des différents services d'appariement proposés et leurs traitements des différents cas de figure.

Chapitre 5

Etude de cas et implémentation

Sommaire

1. Introduction
 2. Application des services d'appariement
 - 2.1 L'appariement Exact et partiel des compétences
 - 2.1.1 L'appariement de l'offre d'emploi offre1 avec CV1
 - 2.1.1 L'appariement de l'offre d'emploi offre1 avec CV2
 - 2.1.3. L'appariement partiel des compétences entre l'offre1 avec CV1
 - 2.1.4. L'appariement partiel des compétences l'offre1 avec CV2
 - 2.2 L'appariement rapproché des compétences
 - 2.2.1 L'appariement de l'offre d'emploi offre2 avec CV3
 - 2.2.2 L'appariement de l'offre d'emploi offre2 avec CV4
 - 2.2.3 L'appariement de l'offre d'emploi offre2 avec CV5
 3. Aspect d'implémentation
 4. Conclusion
-

«J'écoute et j'oublie, je vois et je me souviens, je fais et je comprends »

Proverbe chinois

1. Introduction

Dans ce chapitre, nous présentons une étude de cas qui concrétise notre contribution. Nous choisissons des CVs et des offres d'emplois variés pour montrer l'application des différents services d'appariement proposés et leurs traitements des cas particuliers. Ces services traitent les trois aspects i) les qualifications personnelles. ii) les exigences. iii) les compétences. Leurs aspects d'implémentation sont détaillés dans la section 3.

2. Application des services d'appariement

Afin de mieux comprendre le déroulement des services d'appariement proposés, nous choisissons des CVs et des offres d'emploi variés puis nous utilisons le processus d'annotation proposé dans le chapitre 3 pour annoter ces derniers. Ces CVs et ces offres d'emploi permettent de bien montrer l'application des différents services proposés. L'algorithme CV/offre d'emploi proposé utilise les variables : SimComp, SimNiveau, SimContext pour représenter respectivement le degré de similarité des compétences, le degré de similarité des niveaux des compétences, le degré de similarité des contextes des compétences.

2.1 L'appariement Exact et partiel des compétences

Pour mieux comprendre l'exécution des deux services d'appariement des compétences Exact et partiel, nous choisissons une offre d'emploi (offre1) et deux CVs (CV1, CV2). **Nous les annotons par l'outil** FETCH [ALLI08] voir figure 5.1. Les fichiers des annotations des CVs/offres d'emploi sont détaillés dans l'annexe B. Nous exposons les fichiers des annotations de l'offre et des deux CVs par la suite en langage naturel pour des raisons de lisibilité.

En premier lieu, nous montrons l'application de l'appariement exact des compétences, puis nous montrons l'application de l'appariement partiel (l'appariement des qualifications personnelles et l'appariement des exigences sont appliqués dans tous les cas). Le tableau 5.1 synthétise le résultat de l'appariement de l'offre 1 avec CV1 et CV2 par l'application des services cités auparavant.

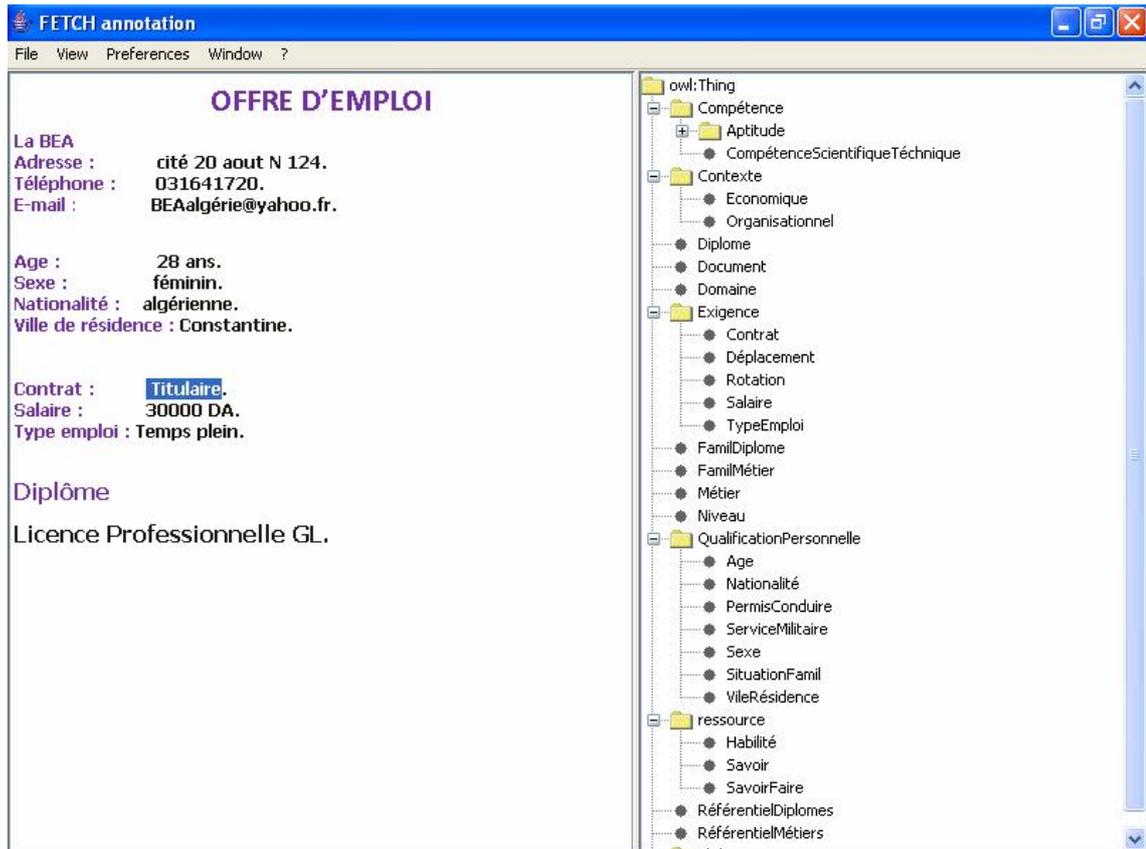


Figure.5.1 Processus d'annotation de l'offre1.

Poids : 05	Age 28	maxDag : 3 ans
Poids : 05	Sexe féminin	
Poids : 05	Nationalité algérien	
Poids : /	Permis de conduire /	
Poids : /	Situation Familiale /	
Poids : 05	Ville de résidence Constantine	maxDA : 150 Km
Poids : /	Service national /	
Poids : 00	Déplacé : /	
Poids : 10	Contrat : titulaire	
Poids : 10	Salaire : 30 000DA	maxDif : 6000 DA
Poids : 00	Rotation : /	
Poids : 10	Type emploi : temps plein	
Poids : 50	Compétence : C ++	niveau : Compétent
Contexte économique : Grande entreprise		Contexte organisationnel : grand service
Poids : 30	Compétence : Résistance aux pressions	niveau : forte
Contexte économique : Grande entreprise		Contexte organisationnel : grand service

Figure 5.2. Fichier des annotations de l'offre1.

Age	28
Sexe	féminin
Nationalité	algérien
Permis de conduire	non
Situation Familiale	célibataire
Ville de résidence	Constantine
Service national	dégagé
Déplacé :	non
Contrat :	titulaire
Salaire :	25 000DA
Rotation :	non
Type emploi :	temps plein
Compétence :	C++
niveau :	Débutant
Contexte économique :	entreprise moyenne
Contexte organisationnel :	service moyen
Compétence :	Résistance aux pressions
niveau :	Faible
Contexte économique :	entreprise moyenne
Contexte organisationnel :	service moyen

Figure 5.3. Fichier des annotations du CV1.

Age	30
Sexe	féminin
Nationalité	algérien
Permis de conduire	non
Situation Familiale	célibataire
Ville de résidence	Skikda
Service national	non dégagé
Déplacé :	non
Contrat :	titulaire
Salaire :	37 000DA
Rotation :	non
Type emploi :	temps plein
Compétence :	C++
niveau :	Intermédiaire
Contexte économique :	Grande entreprise
Contexte organisationnel :	petit service
Compétence :	Résistance aux pressions
niveau :	forte
Contexte économique :	Grande entreprise
Contexte organisationnel :	petit service

Figure 5.4. Fichier des annotations du CV2.

2.1.1 L'appariement de l'offre d'emploi offre1 avec CV1 (nous utilisons leurs fichiers d'annotations pour extraire les requis et les acquis).

► ***Appariement des qualifications personnelles***

Poids : 05	Age :	28	28	maxDAg : 3 ans
Poids : 05	Sexe :	féminin	féminin	
Poids : 05	Nationalité :	algérien	algérien	
Poids : /	Permis de conduire :	/	non	
Poids : /	Situation Familiale :	/	célibataire	
Poids : 05	Ville de résidence :	Constantine	Constantine	maxDA: 150 Km

Poids : / Service national : / dégagé

$$PQ_{\text{personnel}} := 05 + 05 + 05 + 05 = 20$$

$$\text{TotalOP} := 20$$

$$DA_{Q_{\text{Personnel}}} := (20 / 20) * 100 := 100 \%$$

► *Appariement des exigences*

Poids : / Déplacé : / non

Poids : 10 Contrat : titulaire titulaire

Poids : 10 Salaire : 30 000DA 25 000DA maxDif : 6000 DA

Poids : / Rotation: / non

Poids : 10 Type emploi : temps plein temps plein

$$PE_{\text{Exigence}} := 10 + 10 + 10 := 30$$

$$\text{TotalEX} := 30$$

$$DA_{\text{Exigence}} := (30 / 30) * 100 := 100 \%$$

► *Appariement exact des compétences*

Dans ce qui suit, nous allons comparer les compétences qui annotent l'offre1 (C++, résistance aux pressions) avec (C++, résistance aux pressions) qui annotent le CV1 par l'application du service ExactComp.

Compétence : C ++

Compétence : C++

niveau : Compétent

niveau : Débutant

Contexte économique : Grande entreprise **Contexte économique :** Entreprise moyenne

Contexte organisationnel : Grand service **Contexte organisationnel :** Service moyen

$$\text{Sim}(C ++, C ++) := 1$$

Pour la compétence « C++ » de l'offre 1 nous calculons le produit du niveau par EVAcontext.

$$\text{EVAcontext}(\text{grande entreprise, grand service}) = 6 * 3/4 + 6 * 1/4 = 6$$

$$\text{Niveau} * \text{EVAcontext}(\text{grande entreprise, grand service}) = 4 * 6 = 24$$

Pour la compétence « C++ » de CV1 nous calculons le produit du niveau par EVAcontext.

$$\text{EVAcontext}(\text{entreprise moyenne, service moyen}) = 5 * 3/4 + 5 * 1/4 = 5$$

Niveau * EVAcontext (entreprise moyenne, service moyen) = $1 * 5 = 5$

24 n'est pas inférieur à 5 alors PComp : = 0

Compétence : Résistance aux pressions **Compétence : Résistance aux pressions**

niveau : forte **niveau : faible**

Contexte économique : Grande entreprise **Contexte économique : entreprise moyenne**

Contexte organisationnel : grand service **Contexte organisationnel : service moyen**

Sim (résistance aux pressions, résistance aux pressions) = 1.

Pour la compétence « résistance aux pressions » de l'offre1 nous calculons le produit du niveau par EVAcontext.

EVAcontext (grande entreprise, grand service) = $6 * 3/4 + 6 * 1/4 = 6$

Niveau * EVAcontext (grande entreprise, grand service) = $2 * 6 = 12$

Pour la compétence « C++ » de CV nous calculons le produit du niveau par EVAcontext.

EVAcontext (entreprise moyenne, service moyen) = $5 * 3/4 + 5 * 1/4 = 5$

Niveau * EVAcontext (grande entreprise, grand service) = $1 * 5 = 5$

12 n'est pas inférieur à 5 alors PComp : = 0

Le degré d'appariement final des compétences CompExact : = $0 + 0 = 0$

Le degré d'appariement de tous les acquis avec tous les requis est :

$DA_{Final} : = (30 + 20) / (30 + 20 + 80) * 100 = 38.46 \%$

2.1.2 L'appariement de l'offre d'emploi offre1 avec CV2 (nous utilisons leurs fichiers d'annotations pour extraire les requis et les acquis).

► *Appariement des qualifications personnelles*

Poids : 15	Age : 28	30	maxDAg : 3 ans
Poids : 05	Sexe : féminin	féminin	
Poids : 05	Nationalité : algérien	algérien	
Poids : /	Permis de conduire : /	non	
Poids : /	Situation Familiale : /	célibataire	
Poids : 15	Ville de résidence : Constantine	skikda	maxDA: 150 Km

Poids : / Service national : / dégagé

$$PQ_{\text{personnel}} := (15 * (1 - ((1/3) * (30 - 28)))) + 05 + 05 + (15 * (1 - ((1/150) * 80)))$$

$$= 20$$

$$\text{TotalOP} := 40$$

$$DA_{Q_{\text{Personnel}}} := (20/40) * 100 = 50 \%$$

► *Appariement des exigences*

Poids : / Déplacé : / non
 Poids : 05 Contrat : titulaire titulaire
 Poids : 10 Salaire : 30 000DA 37 000DA maxDif : 6000 DA
 Poids : / Rotation : / non
 Poids : 05 Type emploi : temps plein temps plein

$$PE_{\text{Exigence}} := 05 + 0 \text{ (par ce que : } 7\,000 \leq 6\,000) + 05 = 10$$

$$\text{TotalEX} := 20$$

$$DA_{\text{Exigence}} := (10 / 20) * 100 = 50 \%$$

► *Appariement exact des compétences*

Dans ce qui suit, nous allons comparer les compétences annotant l'offre1 qui sont (C++, résistance aux pressions) avec (C++, résistance aux pressions) annotant CV2 par l'application du service ExactComp.

Compétence : C ++

niveau : Compétent

Contexte économique : Grande entreprise

Contexte organisationnel : Grand service

$$\text{Sim}(C ++, C ++) = 1$$

Compétence : C++

niveau : Intermédiaire

Contexte économique : Grande entreprise

Contexte organisationnel : petit service

Pour la compétence « C++ » de l'offre1 le produit du niveau par EVAcontext.

égal a 24 (déjà calculé).

Pour la compétence « C++ » de CV2 nous calculons le produit du niveau par EVAcontext.

$$\text{EVAcontext (grande entreprise, petit service)} = 6 * 3/4 + 4 * 1/4 = 5.5$$

Niveau * EVAcontext (grande entreprise, petit service) = $2 * 5.5 = 11$

24 n'est pas inférieur à 11 alors PComp : = 0.

Compétence : Résistance aux pressions **Compétence : Résistance aux pressions**

niveau : forte

niveau : forte

Contexte économique : Grande entreprise **Contexte économique : Grande entreprise**

Contexte organisationnel : grand service **Contexte organisationnel : petit service**

Sim (résistance aux pressions, résistance aux pressions) = 1

Pour la compétence « résistance aux pressions » de l'offre1 le produit du niveau par EVAcontext égal 12 (déjà calculé).

Pour la compétence « résistance aux pressions » de CV2 nous calculons le produit du niveau par EVAcontext.

EVAcontext (grande entreprise, petit service) = $6 * 3/4 + 4 * 1/4 = 5.5$

Niveau * EVAcontext (grande entreprise, petit service) = $2 * 5.5 = 11$.

12 n'est pas inférieur à 11 alors PComp : = 0.

Le degré final d'appariement des compétences CompExact: = $0 + 0 = 0$.

Le degré d'appariement de tous les acquis avec tous les requis est :

DA_{Final} : = $(20 + 10) / (40 + 20 + 80) * 100 = 21.42 \%$

2.1.3. L'appariement partiel des compétences entre l'offre1 avec CV1 (nous utilisons leurs fichiers d'annotation pour extraire les requis et les acquis).

Dans ce qui suit, nous allons calculer le produit du degré de similarité du contexte et le degré de similarité du niveau, de chaque compétence de l'offre avec celle du CV de même type (compétence scientifique /technique ou aptitude).

SimNiveau (niveau (C ++), niveau (C ++)) = SimNiveau (compétent, débutant) =
= $1 - (1/5(4 - 1)) = 2/5$.

SimContext(contexte (C ++), contexte (C ++))

= $3/4 * \text{similarité économique contexte} + 1/4 * \text{similarité contexte organisationnel}$

= $3/4 * \text{SCEco (grande entreprise, entreprise moyenne)} + 1/4 * \text{SCOrg (grand service, service moyen)} = 5/6$

SimNiveau * SimContext = $2/5 * 5/6 = 1/3$.

PComp = $1/3 * 50 = 16.66$

SimNiveau (niveau (résistance aux pressions), niveau (résistance aux pressions)) =

$$\text{SimNiveau (forte, faible)} = 1 - (1/2(2 - 1)) = 1/2$$

SimContext (contexte (résistance aux pressions), contexte (résistance aux pressions))

$$= \frac{3}{4} * \text{similarité contexte économique} + \frac{1}{4} * \text{similarité contexte organisationnel}$$

$$= \frac{3}{4} * \text{SCEco (grande entreprise, entreprise moyenne)} + \frac{1}{4} * \text{SCOrg (grand service, service moyen)} = 5/6$$

$$\text{SimNiveau} * \text{SimContext} = 1/2 * 5/6 = 5/12$$

$$\text{PComp} = 5/12 * 30 = 12.5$$

Le degré d'appariement partiel des compétences entre l'offre 1 et le CV1 est :

$$\text{PComp} = 12.5 + 16.66 = 29.16$$

$$\text{TotalComp} = 80.$$

$$\text{DA}_{\text{Compétence}} = (((16.66 + 12.5) / 80) * 100) = 36.45\%$$

Le degré d'appariement de tous les acquis avec tous les requis est :

$$\text{DA}_{\text{Final}} := ((20 + 10 + 29.16) / (40 + 20 + 80)) * 100 = 63.68 \%$$

2.1.4. L'appariement partiel des compétences l'offre1 avec CV2 (nous utilisons leurs fichiers d'annotations pour extraire les requis et les acquis).

Dans ce qui suit, nous allons calculer le produit de degré similarité de contexte par le degré de similarité de niveau, de chaque compétence annote l'offre avec la compétence de même nom qui annote le CV.

SimNiveau (niveau (C ++), niveau (C ++)) = SimNiveau (compétent, intermédiaire) =

$$= 1 - (1/5 * (4 - 2)) = 3/5$$

SimContext(contexte (C ++), contexte (C ++))

$$= \frac{3}{4} * \text{similarité contexte économique} + \frac{1}{4} * \text{similarité contexte organisationnel}$$

$$= \frac{3}{4} * \text{SCEco (grande entreprise, grande entreprise)} + \frac{1}{4} * \text{SCOrg (grand service, petit service)} = \frac{3}{4} + \frac{1}{4} * (1 - 1/6(6 - 4)) = \frac{3}{4} + \frac{1}{4} * (2/3) = 0.91$$

$$\text{SimNiveau} * \text{SimContext} = 3/5 * 0.91 = 0.549$$

$$\text{PComp} = 0.55 * 50 = 27.5$$

SimNiveau (niveau (résistance aux pressions), niveau (résistance aux pressions)) =

$$\text{SimNiveau (forte, forte)} = 1$$

SimContext (contexte (résistance aux pressions), contexte (résistance aux pressions))

$$= \frac{3}{4} * \text{similarité contexte économique} + \frac{1}{4} * \text{similarité contexte organisationnel}$$

$$= \frac{3}{4} * \text{SCEco (grande entreprise, grande entreprise)} + \frac{1}{4} * \text{SCOrg (grand service, petit service)} = \frac{3}{4} * 1 + \frac{1}{4} * (1 - 1/6(6 - 4)) = \frac{3}{4} + \frac{1}{4} * 2/3 = 0.91$$

$$\text{SimNiveau} * \text{SimContext} = 1 * 0.91 = 0.91.$$

$$\text{PComp} = 0.91 * 30 = 27.3.$$

$$\text{TotalComp} = 80.$$

Le degré d'appariement partiel des compétences entre l'offre d'emploi 1 et le CV2 est :

$$\text{PComp} : = 27.5 + 27.3 = 54.8$$

$$\text{DA}_{\text{Compétence}} = ((54.8)/80) * 100 = 69 \%$$

Le degré d'appariement de tous les acquis avec tous les requis est :

$$\text{DA}_{\text{Final}} : = (20 + 10 + 54.8) / (40 + 20 + 80) * 100 = 60.57 \%$$

Le tableau 5.1 résume le résultat d'appariement de l'offre 1 avec CV1 et CV2:

Degré d'appariement	DA _{Final} dans le cas d'appariement Exact des compétences	Exact Compétence	DA _{Final} dans le cas d'appariement Partiel des compétences	Partiel Compétence	Qualification personnelle	Exigence
CV1	38.46 %	0 %	63.68 %	36.45 %	100%	100%
CV2	21.42 %	0 %	60.57 %	69 %	50%	50%

Tableau 5.1. Les degrés d'appariement de l'offre1 avec CV1 et CV2

2.2 L'appariement rapproché des compétences

Nous choisissons une offre d'emploi (offre2) et trois CV (CV3, CV4, CV5) variés pour montrer l'application du service d'appariement rapproché des compétences et aussi l'appariement des qualifications personnelles et celui des exigences et pour bien montrer aussi le traitement des cas particuliers. Les CVs et l'offre d'emploi nous les annotons par l'outil FETCH voir la figure 5.5 et la figure 5.6. Les fichiers des annotations des CVs/offres d'emploi sont détaillés dans l'annexe B. Nous exposons les fichiers des annotations de l'offre et des deux CVs par la suite en langage naturel pour des raisons de lisibilité aux lecteurs.

Le résultat de l'appariement pour chaque couple (offre1, CV_x) et $x \in \{3, 4, 5\}$: est le degré d'appariement de tous les acquis avec tous les requis (DA_{final}) le degré d'appariement des qualifications personnelles (DA_{QPersonnel}), le degré d'appariement des exigences (DA_{Exigence}), le degré d'appariement des compétences (DA_{compétence}) ils sont

synthésisés dans le tableau 5.2 et triés selon DA_{Final} puis par $DA_{compétence}$ puis par $DA_{QPersonnel}$ puis $DA_{Exigence}$ (c'est un choix fait par exemple par un utilisateur) .

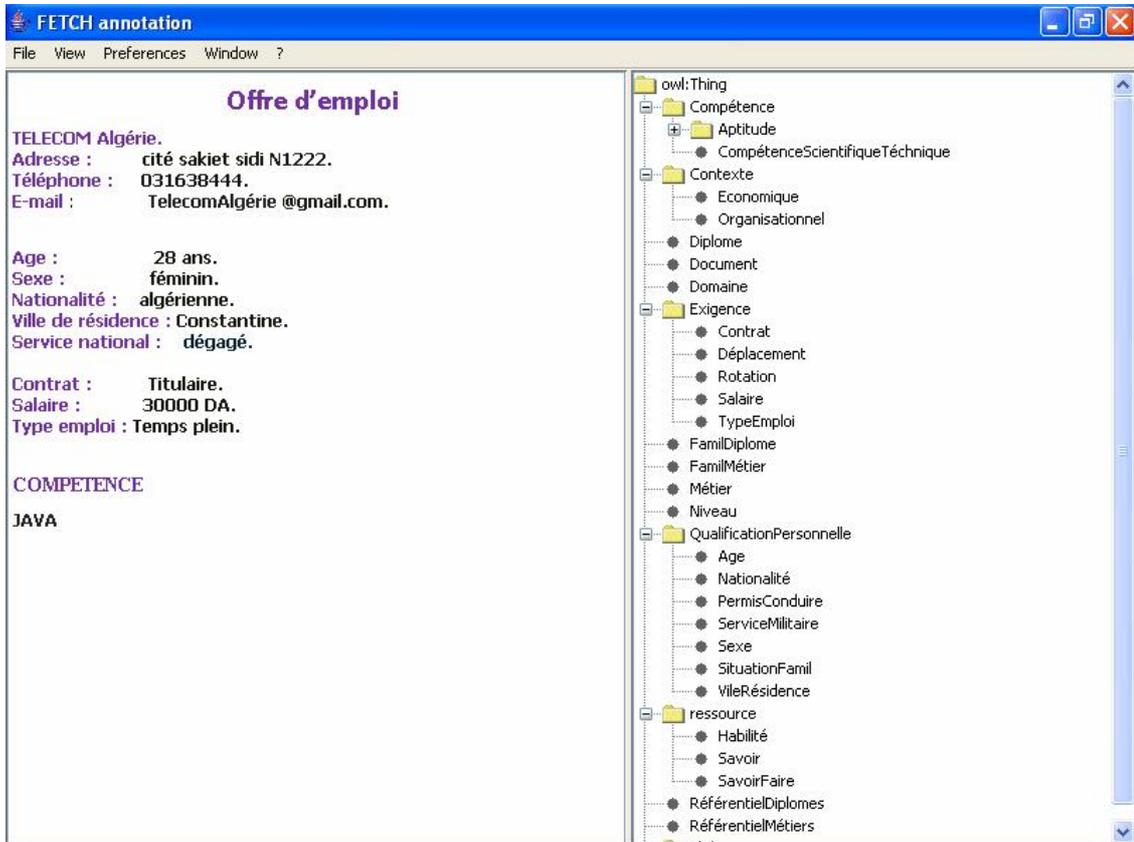


Figure.5.5. Processus d'annotation de l'offre2.

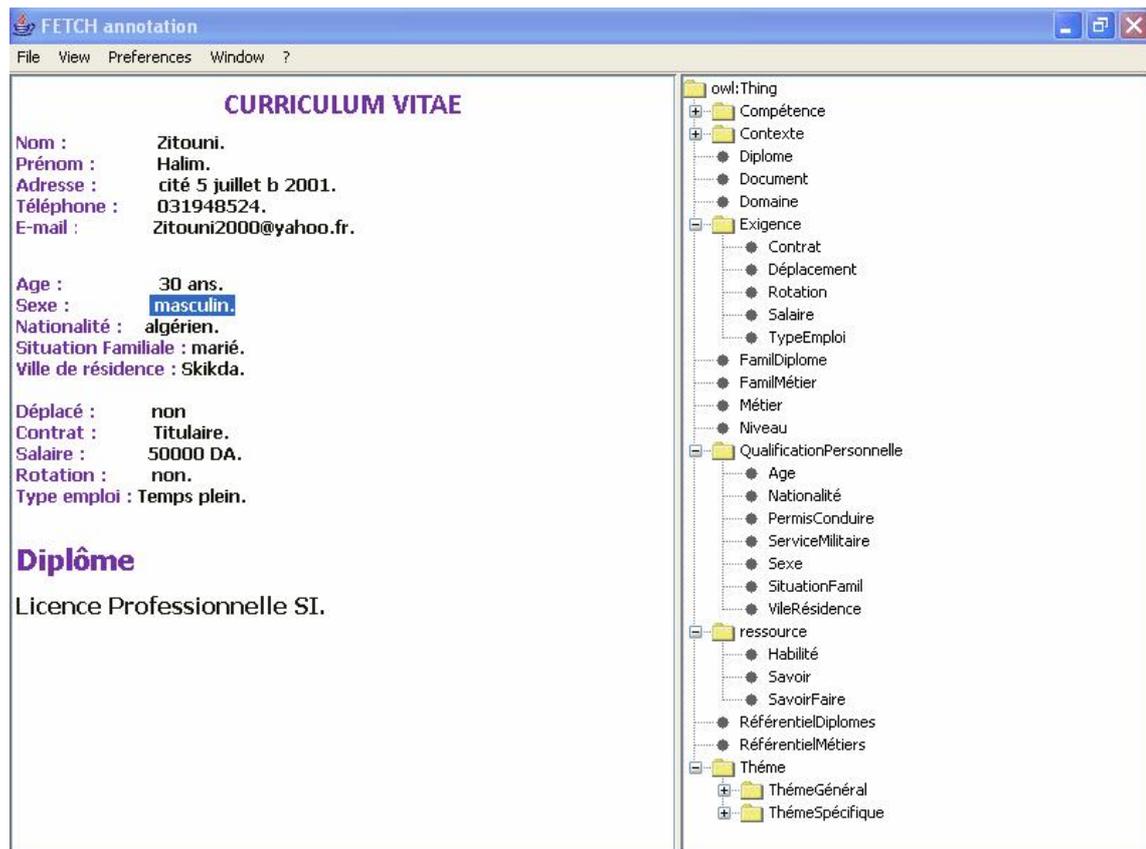


Figure.5.6. Processus d’annotation de CV3.

Poids : 15	Age : 28	maxDAg : 2 ans
Poids : 05	Sexe : féminin	
Poids : 05	Nationalité : algérien	
Poids : 00	Permis de conduire : /	
Poids : 00	Situation Familiale : /	
Poids : 15	Ville de résidence : Constantine	maxDA: 90 Km
Poids : 05	Service national : dégage	
Poids : 00	Déplacé : /	
Poids : 05	Contrat : titulaire	
Poids : 10	Salaire : 30 000DA	maxDif : 5000 DA
Poids : 05	Rotation : non	
Poids : 05	Type emploi : temps plein	
Poids : 50	Compétence : java	niveau : Compétent
Contexte économique : Grande entreprise		
Contexte organisationnel : service moyen		
Poids : 30	Compétence : Travail en équipe	niveau : forte
Contexte économique : Grande entreprise		
Contexte organisationnel : service moyen		

Figure.5.7. Fichier des annotations de l’offre2.

Age : 30
Sexe : masculin
Nationalité : algérien
Permis de conduire : /
Situation Familiale : marié
Ville de résidence : Skikda
Service national : dégage
Déplacé : non
Contrat : titulaire
Salaire : 50 000DA
Rotation : non
Type emploi : temps plein
Compétence : pascal **Niveau :** Compétent
Contexte économique : Grande entreprise **Contexte organisationnel :** Grand service
Compétence : coopération **Niveau :** forte
Contexte économique : petite entreprise **Contexte organisationnel :** petit service

Figure 5.8. Fichier des annotations de CV3.

Age 26
Sexe féminin
Nationalité : algérien
Permis de conduire : /
Situation Familiale : célibataire
Ville de résidence : Constantine
Service national : dégage
Déplacé : non
Contrat : titulaire
Salaire : 34 000DA
Rotation : non
Type emploi : temps plein
Compétence : lisp **Niveau :** Compétent
Contexte économique : petite entreprise **Contexte organisationnel :** petit service
Compétence : gestion de situation **Niveau :** faible
Contexte économique: petite entreprise **Contexte organisationnel:** petit service

Figure 5.9. Fichier des annotations de CV4.

Age : 34
Sexe : féminin
Nationalité palestinien
Permis de conduire : /
Situation Familiale : célibataire
Ville de résidence : Sétif
Service national : dégage
Déplacé : non
Contrat : titulaire
Salaire : 30 000DA
Rotation : non
Type emploi : temps plein
Compétence : C++ **Niveau :** professionnel
Contexte économique : Centre universitaire **Contexte organisationnel :** Grand group
Compétence : Autonomie **Niveau :** faible
Contexte économique : Grande entreprise **Contexte organisationnel :** Grand service
Compétence : Travail en équipe **Niveau :** forte
Contexte économique : Centre universitaire **Contexte organisationnel :** Grand group

Figure 5.10. Fichier des annotations de CV5.

$$(15 * (1 - ((1 / (90)) * 80))) + 05 = 16.8$$

$$\text{TotalOP} : = 45$$

$$\text{DA}_{\text{Personnel}} : = (16.8 / 45) * 100 : = 37.33 \%$$

► *Appariement des exigences*

Poids : /	Déplacé : /		non
Poids : 05	Contrat : titulaire		titulaire
Poids : 10	Salaire : 30 000DA	50 000DA	maxDif : 5000 DA
Poids : 05	Rotation : non		non
Poids : 05	Type emploi : temps plein		temps plein

$$\text{PExigence} : = 05 + 0 \text{ (le zéro est due pace que : } 30\,000 < 50\,000) + 05 + 05 : = 15$$

$$\text{TotalEX} : = 25$$

$$\text{DA}_{\text{Exigence}} : = (15 / 25) * 100 : = 60 \%$$

► *Appariement rapproché des compétences*

Dans ce qui suit nous allons comparer les compétences qui annotent l'offre2 qui sont {java, travail en équipe} avec {pascal, coopération} qui annotent CV3 par l'application du service RapprochéComp. Chaque compétence scientifique ou technique mobilise un ensemble de ressources (un extrait de la hiérarchie des thèmes des ressources est illustré dans la figure 5.11, le nombre de concepts dans la hiérarchie des thèmes spécifique égale à 361), java mobilise les ressources suivantes : (savoir mathématique, savoir anglais, savoir en langage orienté objet, savoir faire algorithmique, savoir-faire java, habilité de raisonner, perception des enjeux), pascal mobilise les ressources suivantes : (savoir mathématique, savoir anglais, savoir en langage impératif, savoir faire algorithmique, savoir-faire pascal, habilité de raisonner, perception des enjeux).

Le calcul de la similarité de «java » et «pascal » selon leurs ressources, leurs contextes, leurs niveaux est le suivant :

$$\text{DA}_{\text{habilité}} = \text{Sim}(\text{raisonner}, \text{raisonner}) + \text{Sim}(\text{perception des enjeux}, \text{perception des enjeux}) = 1 + 1 = 2$$

$$\text{DA}_{\text{savoir}} = \text{Sim}(\text{mathématique}, \text{mathématique}) + \text{Sim}(\text{anglais}, \text{anglais}) + \text{Sim}(\text{langage orienté objet}, \text{langage impératif})$$

$$\begin{aligned} \text{DA}_{\text{savoir}} &= 1 + 1 + \text{Sim}(\text{langage orienté objet}, \text{langage impératif}) \\ &= 1 + 1 + 0.73 = 2.73 \end{aligned}$$

$$DA_{\text{savoirF}} = \text{Sim}(\text{algorithmique}, \text{algorithmique}) + \text{Sim}(\text{java}, \text{pascal}) \\ = 1 + \text{SimComp}(\text{java}, \text{pascal}) = 1 + 0.49 = 1.49.$$

$$\text{SimComp} = F(R, A, \text{mobilise}) = DA_{\text{habilité}} + DA_{\text{savoir}} + DA_{\text{savoirF}} / N \\ = (2 + 2.73 + 1.49) / 7 = 0.89.$$

$$\text{SimNiveau}(\text{niveau}(\text{java}), \text{niveau}(\text{pascal})) = \text{SimNiveau}(\text{compétent}, \text{compétent}) = 1.$$

$$\text{SimContext}(\text{contexte}(\text{java}), \text{contexte}(\text{pascal})) \\ = \frac{3}{4} * \text{similarité} \text{ contexte économique} + \frac{1}{4} * \text{similarité} \text{ contexte organisationnel} \\ = \frac{3}{4} * \text{SCOrg}(\text{grande entreprise}, \text{grande entreprise}) + \frac{1}{4} * \text{SCEco}(\text{service moyen}, \\ \text{grand service}). \\ = \frac{3}{4} * 1 + \frac{1}{4} * 1 = 1.$$

$$\text{PComp} := 50 * 1 * 1 * 0.89 = 44.5.$$

Pour calculer la similarité des deux aptitudes « travail en équipe » et « coopération », nous calculons d'abord la similarité de leurs niveaux puis de leur contexte, puis de leur nom (sinon de leur catégorie).

$$\text{SimNiveau}(\text{niveau}(\text{travail en équipe}), \text{niveau}(\text{coopération})) = \text{SimNiveau}(\text{forte}, \text{forte}) \\ = 1.$$

$$\text{SimContext}(\text{contexte}(\text{travail en équipe}), \text{contexte}(\text{coopération})) \\ = \frac{1}{4} * \text{similarité} \text{ contexte organisationnel} + \frac{3}{4} * \text{similarité} \text{ contexte économique} \\ = \frac{1}{4} * \text{SCOrg}(\text{service moyen}, \text{petit service}) + \frac{3}{4} * \text{SCEco}(\text{grande entreprise}, \text{petite} \\ \text{entreprise}) \\ = \frac{1}{4} * (1 - (1/6 * (5 - 4))) + \frac{3}{4} * (1 - (1/6 * (6 - 4))) = 0.7$$

$$\text{PComp} := \text{poids} * \text{SimNiveau} * \text{SimContext} * \text{SimComp} = 30 * 1 * 0.7 * \frac{1}{4} = 5.25$$

$\text{SimComp} = 1/4$ parce que « travail en équipe » et « coopération » sont de la même catégorie.

$$\text{PComp} := 44.5 + 5.25 = 49.75$$

$$\text{TotalComp} := 30 + 50 = 80$$

$$DA_{\text{Compétence}} := (49.75 / 80) * 100 = 62.18 \%$$

Le degré d'appariement de tous les acquis avec tous les requis est :

$$DA_{\text{Final}} := (16.8 + 15 + 49.75) / (45 + 25 + 80) * 100 = 54.36 \%$$

2.2.2 L'appariement de l'offre d'emploi offre2 avec CV4 (nous utilisons leurs fichiers d'annotations pour extraire les requis et les acquis).

Nous procédons de la même manière pour le CV4, nous comparerons les requis du recruteur avec les acquis du chercheur d'emploi par l'application des services proposés (appariement des qualifications personnelles, appariement des exigences, appariement rapproché des compétences).

► ***Appariement des qualifications personnelles***

Poids : 15	Age : 28	26	maxDAg : 3 ans
Poids : 05	Sexe : féminin	féminin	
Poids : 05	Nationalité : algérien	algérien	
Poids : /	Permis de conduire : /	/	
Poids : /	Situation Familiale : /	célibataire	
Poids : 15	Ville de résidence : Constantine	Constantine	maxDA: 90 Km
Poids : 05	Service national : dégage	dégagé	

$$PQ_{\text{personnel}} := (15 * (1 - ((1/3) * (28 - 26)))) + 05 + 05 + 15 + 05 = 35$$

$$\text{TotalOP} := 45$$

$$DA_{Q_{\text{Personnel}}} := (35/45) * 100 = 77.77 \%$$

► ***Appariement des exigences***

Poids : /	Déplacé : /	non	
Poids : 05	Contrat : titulaire	titulaire	
Poids : 10	Salaire : 30 000DA	34 000DA	maxDif : 5000 DA
Poids : 05	Rotation : non	non	
Poids : 05	Type emploi : temps plein	temps plein	

$$PE_{\text{Exigence}} := 05 + (10 * (1 - ((1/5000) * (34000 - 30000)))) + 05 + 05 := 17$$

$$\text{TotalEX} := 25$$

$$DA_{\text{Exigence}} := (17 / 25) * 100 = 68 \%$$

► ***Appariement rapproché des compétences***

Dans ce qui suit nous allons comparer les compétences requises avec celles acquises, par l'application du service RapprochéComp, sachant que les compétences annotant l'offre2 sont {java, travail en équipe} et {lisp, gestion de situation} pour CV4.

Les ressources de java comme indiquées précédemment sont (savoir mathématique, savoir anglais, savoir en langage orienté objet, savoir faire algorithmique, savoir-faire java, habilité de raisonner, perception des enjeux).

Lisp mobilise les ressources suivantes : (savoir mathématique, savoir anglais, savoir en langage fonctionnel, savoir faire algorithmique, savoir-faire lisp, habilité de raisonner, perception des enjeux).

Le calcul de la similarité de « java » et « lisp » selon leurs ressources, leurs contextes, leurs niveaux est le suivant :

$$DA_{\text{habilité}} = \text{Sim}(\text{raisonner}, \text{raisonner}) + \text{Sim}(\text{perception des enjeux}, \text{perception des enjeux}) = 1 + 1 = 2.$$

$$DA_{\text{savoir}} = \text{Sim}(\text{mathématique}, \text{mathématique}) + \text{Sim}(\text{anglais}, \text{anglais}) + \text{Sim}(\text{langage orienté objet}, \text{langage fonctionnel})$$

$$DA_{\text{savoir}} = 1 + 1 + \text{Sim}(\text{langage orienté objet}, \text{langage fonctionnel}) = 1 + 1 + 0.67 = 2.67$$

$$DA_{\text{savoirfaire}} = \text{Sim}(\text{algorithmique}, \text{algorithmique}) + \text{Sim}(\text{java}, \text{lisp}) = 1 + \text{Sim}(\text{java}, \text{lisp}) = 1 + 0.49 = 1.49$$

$$\text{SimComp} = F(R, A, \text{mobilise}) = DA_{\text{habilité}} + DA_{\text{savoir}} + DA_{\text{savoirfaire}} / N = (2 + 2.67 + 1.49) / 7 = 0.88$$

$$\text{SimNiveau}(\text{niveau}(\text{java}), \text{niveau}(\text{lisp})) = \text{SimNiveau}(\text{compétent}, \text{compétent}) = 1.$$

$$\text{SimContext}(\text{contexte}(\text{java}), \text{contexte}(\text{lisp}))$$

$$= \frac{3}{4} * \text{similarité contexte économique} + \frac{1}{4} * \text{similarité contexte organisationnel}$$

$$= \frac{3}{4} * (\text{grande entreprise}, \text{petite entreprise}) + \frac{1}{4} * (\text{service moyen}, \text{petit service})$$

$$= \frac{3}{4} * (1 - \frac{1}{6}(6-5)) + \frac{1}{4} * (1 - \frac{1}{6}(5-4)) = \frac{3}{4} * \frac{5}{6} + \frac{1}{4} * \frac{5}{6} = \frac{5}{6}$$

$$PComp := \text{poids} * \text{SimComp} * \text{SimNiveau} * \text{SimContext} = 50 * 0.88 * 1 * \frac{5}{6} = 36.52.$$

Pour calculer la similarité des deux aptitudes « travail en équipe » et « gestion de situation » en premier lieu, on calcule la similarité de leurs noms (sinon de leurs catégorie), puis de leurs niveaux, puis de leurs contextes.

Parce que « travail en équipe » et « gestion de situation », ont des noms et des catégories différentes, donc leur similarité est égale à zéro.

$$PComp := 36.52$$

$$\text{TotalComp} := 30 + 50 = 80$$

$$DA_{\text{Compétence}} := (36.52 / 80) * 100 = 45.65\%$$

Le degré d'appariement de tous les acquis avec tous les requis est :

$$DA_{\text{Final}} := (36.52 + 17 + 35) / (80 + 25 + 45) * 100 = 59.01 \%$$

2.2.3 L'appariement de l'offre d'emploi offre2 avec CV5 (nous utilisons leurs fichiers d'annotations pour extraire les requis et les acquis).

On va répéter les mêmes étapes vues précédemment entre offre2 et CV5.

► ***Appariement des qualifications personnelles***

Poids : 15	Age : 28	34	maxDAg : 3 ans
Poids : 05	Sexe : féminin	féminin	
Poids : 05	Nationalité : algérien	palestinien	
Poids : /	Permis de conduire : /	/	
Poids : /	Situation Familiale : /	célibataire	
Poids : 15	Ville de résidence : Constantine	Sétif	maxDA: 90 Km
Poids : 05	Service national : dégage	dégage	

$$PQ_{\text{personnel}} := 0 + 05 + 0 + 0 + 05 = 10$$

$$\text{TotalOP} := 45$$

$$DA_{Q_{\text{Personnel}}} := (10 / 45) * 100 = 22.22 \%$$

► ***Appariement des exigences***

Poids : /	Déplacé : /	non	
Poids : 05	Contrat : titulaire	titulaire	
Poids : 10	Salaire : 30 000DA	30 000DA	maxDif : 5000 DA
Poids : 05	Rotation : non	non	
Poids : 05	Type emploi : temps plein	temps plein	

$$PE_{\text{Exigence}} := 05 + 10 + 05 + 05 := 25$$

$$\text{TotalEX} := 25$$

$$DA_{\text{Exigence}} := (25 / 25) * 100 = 100 \%$$

► ***Appariement rapproché des compétences***

Dans ce qui suit nous allons comparer les compétences annotant l'offre2 qui sont {java, travail en équipe} avec {C++, travail en équipe, autonomie} annotant le CV5 par l'application du service RapprochéComp.

Les ressources de java sont : (savoir mathématique, savoir anglais, savoir en langage orienté objet, savoir faire algorithmique, savoir-faire java, habilité de raisonner, perception des enjeux).

C++ mobilise les ressources suivantes : (savoir mathématique, savoir anglais, savoir en langage orienté objet, savoir faire algorithmique, savoir-faire C++, habilité de raisonner, perception des enjeux).

Le calcul de la similarité de « java » et « C++ » selon leurs ressources, leurs contextes, leurs niveaux est le suivant :

$$DA_{\text{habilité}} = \text{Sim}(\text{raisonner, raisonner}) + \text{Sim}(\text{perception des enjeux, perception des enjeux}) = 1 + 1 = 2$$

$$DA_{\text{savoir}} = \text{Sim}(\text{mathématique, mathématique}) + \text{Sim}(\text{anglais, anglais}) + \text{Sim}(\text{langage orienté objet, langage orienté objet})$$

$$DA_{\text{savoir}} = 1 + 1 + \text{Sim}(\text{langage orienté objet, langage orienté objet}) = 1 + 1 + 1 = 3$$

$$DA_{\text{savoirF}} = \text{Sim}(\text{algorithmique, algorithmique}) + \text{Sim}(\text{java, C++}) \\ = 1 + \text{Sim}(\text{java, C++}) = 1 + 0.67 = 1.67$$

$$\text{SimComp} = F(R, A, mobilise) = DA_{\text{habilité}} + DA_{\text{savoir}} + DA_{\text{savoirF}} / N = (2 + 3 + 1.67) / 7 = 0.95$$

$$\text{SimNiveau}(\text{niveau (java), niveau (C++)}) = \text{SimNiveau}(\text{compétent, professionnel}) = 1$$

$$\text{SimContext}(\text{contexte (java), contexte (C++)})$$

$$= \frac{1}{4} * \text{similarité contexte organisationnel} + \frac{3}{4} * \text{similarité contexte économique}$$

$$= \frac{1}{4} * (\text{service moyen, grand group}) + \frac{3}{4} * (\text{grande entreprise, centre universitaire})$$

$$= \frac{1}{4} * (1 - (1/6 * (5 - 1))) + \frac{3}{4} * (1 - (1/6 * (6 - 1))) = 0.21.$$

$$PComp := \text{poids} * \text{SimComp} * \text{SimNiveau} * \text{SimContext} = 50 * 0.95 * 1 * 0.21 = 9.97.$$

Pour calculer la similarité des deux aptitudes « travail en équipe » et « travail en équipe », et de « travail en équipe » et « autonomie », en premier lieu, on calcule la similarité de leur nom (sinon de leur catégorie), puis de leurs niveaux, puis de leurs contextes.

Puisque « travail en équipe » et « autonomie » n'ont pas le même nom et comme ils n'ont pas aussi la même catégorie, alors leur similarité est égale à zéro.

$$\text{SimNiveau}(\text{niveau (travail en équipe), niveau (travail en équipe)}) = \text{SimNiveau}(\text{forte, forte}) = 1$$

$$\text{SimContext}(\text{contexte (travail en équipe), contexte (travail en équipe)})$$

$$= \frac{1}{4} * \text{similarité contexte organisationnel} + \frac{3}{4} * \text{similarité contexte économique}$$

$$= \frac{1}{4} * (\text{service moyen, grand group}) + \frac{3}{4} * (\text{grande entreprise, centre universitaire})$$

$$= \frac{1}{4} * (1 - (1/6 * (5 - 1))) + \frac{3}{4} * (1 - (1/6 * (6 - 1))) = 0.21.$$

$$PComp := \text{poids} * \text{SimComp} * \text{SimNiveau} * \text{SimContext} = 30 * 1 * 1 * 0.21 = 6.3.$$

$$PComp := 6.3 + 9.97 = 19.42.$$

$$\text{TotalComp} := 30 + 50 = 80.$$

$$DA_{\text{Compétence}} := (19.42 / 80) * 100 = 24.27\%$$

Le degré d'appariement de tous les acquis avec tous les requis est :

$$DA_{\text{Final}} := (10 + 25 + 19.42) / (45 + 25 + 80) * 100 = 36.28\%$$

Degré d'appariement	Final	Compétence	Qualification personnelle	Exigence
CV4	59.01 %	45.65%	77.77 %	68 %
CV3	54.36 %	62.18%	37.33 %	60 %
CV5	36.28%	24.27%	22.22 %	100%

Tableau 5.2. Les degrés d'appariement de l'offre2 avec CV3 et CV4 et CV5

On peut synthétiser d'après les exemples montrés précédemment que l'introduction de degré de rapprochement des contextes dans le calcul de degré d'appariement des compétences influence sur ce dernier, la similarité faible des contextes minimise le degré de rapprochement des compétences même si ces compétences sont similaires (leurs ressources sont similaires ou leurs noms ou leurs catégories) ainsi que leurs niveaux.

3. Aspect d'implémentation

Les fichiers d'annotation des CVs et des offres d'emploi ainsi que l'ontologie-ER sont codés en OWL. Le raisonneur RACER [HAAR04] est un programme Java qui peut interpréter des documents OWL et offre des services de raisonnement aussi bien pour le niveau terminologique de l'ontologie (T_box) qui définit les concepts, que pour le niveau assertionnel (A_box) qui définit les individus et les rôles qui lient ces derniers. Ces deux niveaux constituent une base de connaissance.

RACER dispose d'un langage de requêtes riche RQL (Racer Query Language) [HAAR04] offre des fonctionnalités de raisonnement sur le A-Box et le T-Box qui permettent l'implémentation des services d'appariement basés compétences proposées telles que :

§ Déterminer les descendants et les parents d'un concept dans une hiérarchie : Cette fonctionnalité permet l'implémentation du service appariement RapprochéComp

où précisément la procédure Rapproch permet de calculer le degré de rapprochement des thèmes de ressource (savoir ou savoir faire) en exploitant la hiérarchie des thèmes général ou spécifique.

§ Le calcul du type direct (la classe) d'un individu (instance): cela pourra être utilisé pour les tests de type d'un requis, tel que « si aptitude (requis) », ou « si Aptitude (requis) » ou «si CompétenceScientifiqueTechnique(requis)» dans le service appariement rapproché des compétences «RapprochéComp »

Le code en RQL de ces requêtes est le suivant :

-(retrieve () (|java|) (|CompétenceScientifiqueTechnique|)) permet de tester si la compétence java est une compétence scientifique ou technique.

-(retrieve () (|coopération|) (|Aptitude|)) permet de tester si la compétence coopération est une Aptitude.

§ L'extraction des instances d'une classe particulière, même selon différents critères, à base des rôles et des attributs de ses instances. On utilise cette fonctionnalité par exemple pour l'implémentation des deux procédures « ExtraireComp(FAOffre,CompétenceOffre) »,« ExtraireComp(FACv,CompétenceCv)» qui sont utilisées par les services : appariement Rapproché des compétences,

appariement Exact des compétences, appariement Partiel des compétences. Exemple des requêtes utilisées par ces deux procédures sont :

- (retrieve (?x) (?x Compétence)) : pour extraire les instances de la classe compétence.
- (retrieve () (?x Aptitude)) : répond avec « vrai » s'il existe des instances de la classe citée.
- (retrieve (?x) (?x java aContexte)) : pour extraire le contexte organisationnel et économique d'une compétence.
- (retrieve (? x) (? x java mobiliseResou)) : pour extraire les ressources d'une compétence scientifique ou technique.

Pour l'implémentation des services «appariement des qualifications personnelles » et « appariement des exigences », Java est le langage utilisé pour le codage de ces services. Nous avons utilisé éventuellement les APIs de PROTEGE OWL puisque les deux services n'ont pas besoin d'un raisonnement ontologique. Parmi ces APIs on trouve les APIs :

- ▶ *edu.stanford.smi.protegex.owl.jena*: la classe (JenaOWLModel)
- ▶ *edu.stanford.smi.protegex.owl.model*: les classes (OWLNamedClass, OWLProperty, OWLIndividual, RDFSLiteral).

Et d'autres APIs fournies par l'éditeur PROTEGE OWL. Pour interroger les fichiers des annotations OWL. Le code complet de ces services se trouve à l'annexe C, un extrait est illustré à la figure 5.12 et un résumé des classes est schématisé dans la figure 5.13.

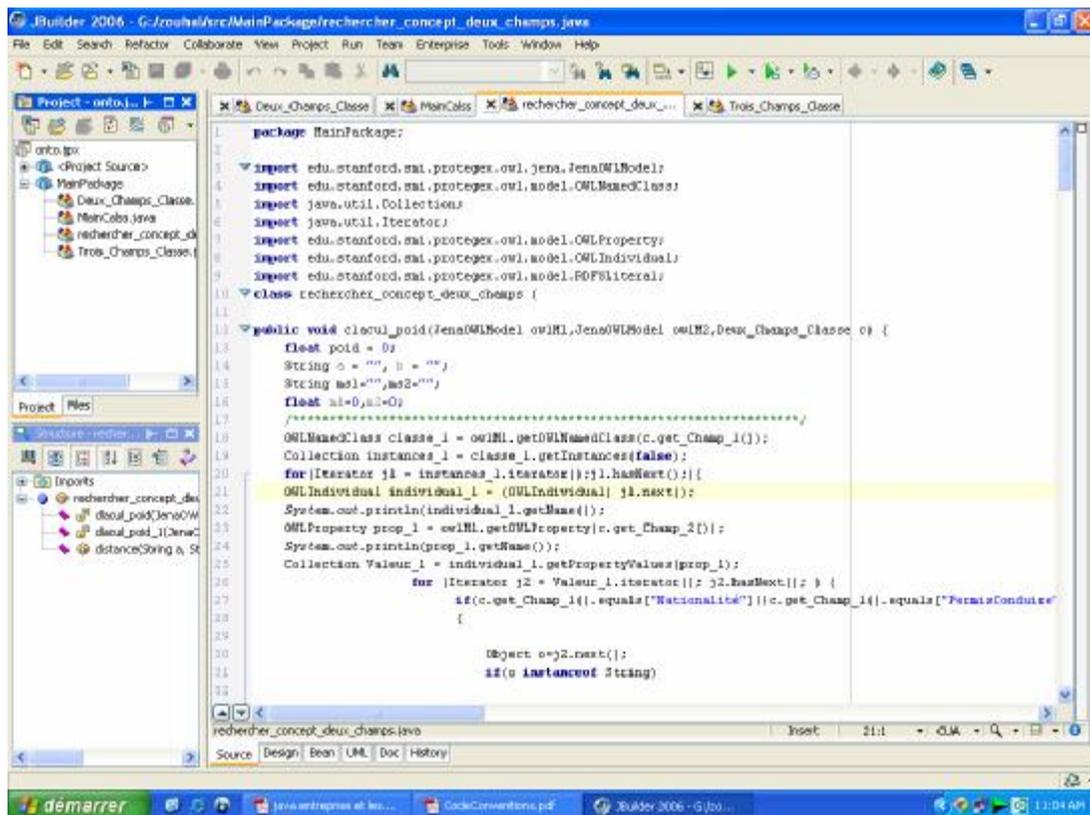


Figure 5.12. Un extrait du code.

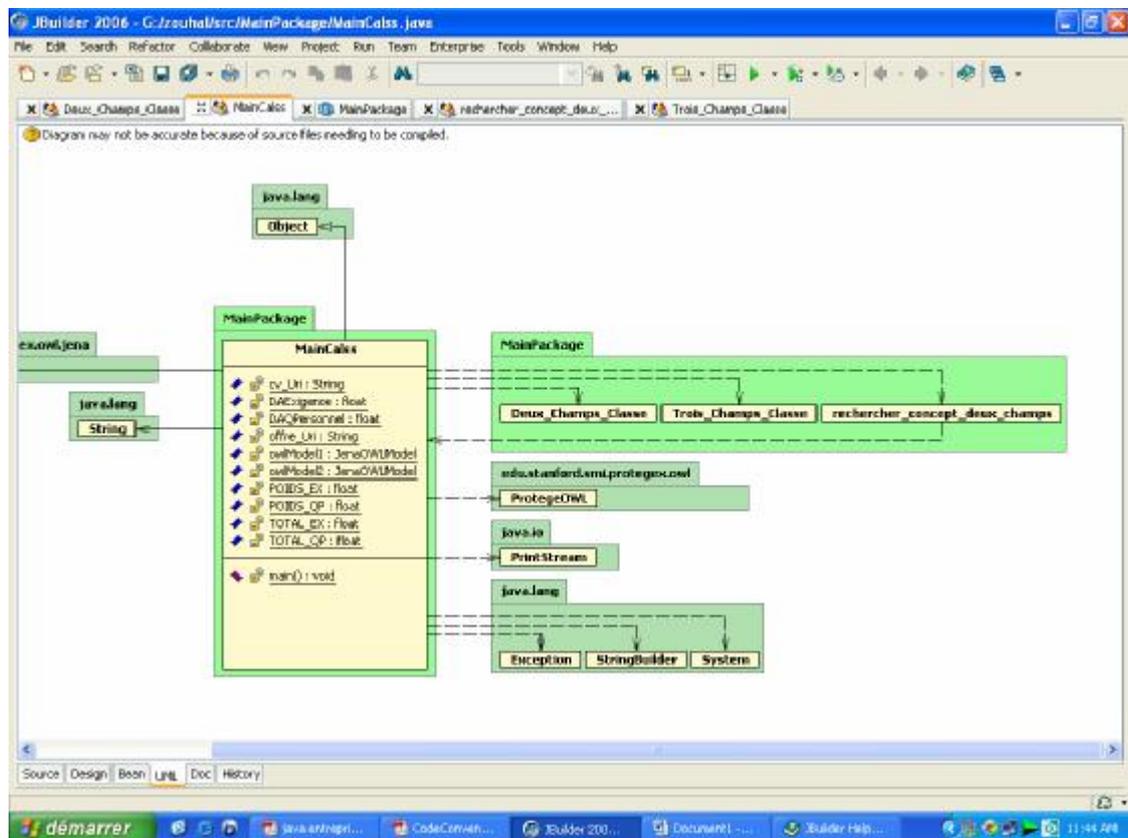


Figure 5.13. Résumé du code.

4. Conclusion

Dans ce chapitre, nous avons appliqué les services proposés dans le chapitre 3 sur des offres d'emplois et des CVs. En premier lieu nous avons choisi une offre d'emploi et deux CVs pour montrer l'exécution des services i) appariement exact des compétences, ii) appariement partiel des compétences et au même temps l'appariement des qualifications personnelles et l'appariement des exigences. Nous avons montré aussi l'application des services i) appariement des qualifications personnelles ii) appariement des exigences iii) appariement rapproché des compétences sur une offre d'emploi et trois CVs variés. Nous avons essayé de traiter la majorité des cas spéciaux, montré que le contexte influence largement le degré d'appariement des compétences, plus les contextes des compétences sont similaires plus le degré de rapprochement des compétences est grand et vice versa.

L'implémentation des services d'appariement basé compétences est possible via les fonctionnalités offertes par le raisonneur RACER, en particulier le langage de requêtes RQL. Pour les services appariement des qualifications personnelles et appariement des exigences nous les **avons implémentés** via les APIs de PROTEGE-OWL dans une application java.

Dans ce qui suit nous présentons une conclusion de notre travail et des perspectives pour des travaux futurs.

Conclusion générale et perspectives

*« Ce n'est pas la fin.
Ce n'est même pas le commencement de la fin.
Mais, c'est peut-être la fin du commencement »*

Winston Churchill

Internet est devenu le canal principal pour le recrutement. Cependant les outils dédiés à la recherche des CVs/offres d'emplois sont purement syntaxiques et les résultats retournés sont loin de satisfaire l'utilisateur. L'introduction des technologies émergentes du Web Sémantique est devenue une nécessité. Parmi ces technologies on trouve l'annotation sémantique, elle est l'approche la plus utilisée par les travaux qui visent l'automatisation du rapprochement des CVs/offres d'emploi, vue sa simplicité. Elle consiste à l'assignement des éléments d'une ontologie (instance de classe, instance de relation) aux éléments d'un document.

En premier lieu, dans ce mémoire nous avons décrit l'architecture globale du système qui supporte l'appariement sémantique des CVs/offre d'emploi, par la suite nous avons détaillé chaque composant et montré l'application de notre approche par une étude de cas.

Pour la modélisation du contenu sémantique des CVs et des offres d'emplois, nous avons construit une ontologie dédiée au e-recrutement. Cette ontologie permet une meilleure gestion des compétences et des connaissances, elle est axée sur la modélisation de la compétence. Le modèle de la compétence proposé est riche et pertinent parce qu'il tient compte du type de la compétence (scientifique et technique ou aptitude), des ressources relatives aux compétences scientifiques (habileté, savoir, savoir-faire), de la catégorie relative aux aptitudes (managériales, résolution des problèmes, ...), du niveau d'expérience et des contextes organisationnel et économique de la compétence. Les concepts et les relations de l'ontologie-ER que nous avons proposée sont inspirés des parties communes les plus significatives des CVs et des offres d'emploi (qualification personnelle, exigence, diplôme,

expérience professionnelle, compétence). Les métiers sont inspirés de CIGREF et les diplômes sont inspirés du référentiel LMD. Le domaine de cette dernière est « l'Informatique et la Télécommunication ». Nous avons adopté METHONTOLOGY pour la conceptualisation de l'ontologie, afin d'atteindre un ensemble de représentations intermédiaires qui facilitent sa formalisation via la logique de descriptions SHOIN. En nous basant sur cette formalisation, nous avons choisi le langage OWL, pour codifier l'ontologie formelle, nous avons utilisé l'éditeur graphique PROTEGE-OWL, pour guider l'implémentation de l'ontologie et pour produire un code OWL. Par ailleurs, pour vérifier et raffiner l'ontologie OWL au cours du processus de développement, nous avons utilisé le système RACER. Ce dernier, peut lire des représentations au format OWL, effectuer des inférences sur le niveau terminologique et assertionnel. Les principaux services d'inférences fournis par le système RACER sont le test de consistance d'un concept « check consistency » et le test de subsomption d'un concept « classify taxonomy ».

Pour l'annotation, nous avons proposé un processus pour les recruteurs et les chercheurs d'emplois, qui consiste à instancier et exploiter l'ontologie-ER par un outil d'annotation [ALLI08] que nous avons adapté pour l'annotation des documents HTML. Ce dernier facilite l'enrichissement des documents avec des annotations. L'appariement sémantique est l'objectif visé par ce mémoire, entre les requis d'un recruteur (fichiers d'annotations de l'offre d'emploi : FA_{offre}) et les acquis d'un chercheur d'emploi (fichier d'annotations d'un CVs : FA_{CV}). IL revient à calculer leur degré de rapprochement. Pour cela, nous avons proposé des services d'appariement qui traitent trois aspects : les qualifications personnelles, les exigences, les compétences. Puisque les compétences sont cruciales pour la sélection d'un chercheur d'emploi (l'offre qui correspond le plus au profil d'un chercheur d'emploi), nous avons proposé trois techniques pour les rapprocher : exacte, partielle et rapprochée. Le résultat d'appariement d'une offre d'emploi avec des CVs (ou vice versa) est un ensemble de quintuplés ($URI, DA_{\text{final}}, DA_{\text{QPersonnel}}, DA_{\text{Exigence}}, DA_{\text{compétence}}$) trié par le degré d'appariement final (DA_{final}) suivi par les trois autres degrés ($DA_{\text{compétence}}, DA_{\text{QPersonnel}}, DA_{\text{Exigence}}$) leur ordre de tri est en fonction du choix de l'utilisateur, l'URI est l'identifiant d'un CV ou d'une offre d'emploi, DA_{final} est le degré d'appariement de tous les acquis avec tous les requis, $DA_{\text{QPersonnel}}$ est le degré d'appariement selon les qualifications personnelles, DA_{Exigence} est le degré d'appariement selon les exigences, $DA_{\text{compétence}}$ est le degré d'appariement selon les compétences.

Nous avons montré dans ce mémoire par une étude de cas l'application des algorithmes proposés. Nous avons pris en considération dans cette étude les différents cas spécifiques des

algorithmes proposés. Nous avons implémentés l'algorithme d'appariement des qualifications personnelles et l'algorithme d'appariement des exigences par l'exploitant des APIs de Protégé-OWL, dans une application JAVA, nous avons ainsi discuté les aspects d'implémentation des algorithmes d'appariement basé compétence qui sont possibles via le raisonneur RACER et le langage de requête RQL.

Dans l'état actuel de cette étude, nous envisageons plusieurs améliorations ou extensions possibles :

1. Pour augmenter la pertinence des résultats d'appariement, nous comptons prendre en compte des contextes plus complexes tels que le contexte technologique (la technologie d'un outil utilisé).
2. Il serait également intéressant de considérer, dans la modélisation de l'ontologie, l'opérateur de formation, comme troisième acteur de notre système (portails de recrutement sémantique), en plus du recruteur et du chercheur d'emploi.
3. La proposition des services adéquats pour la sélection d'un formateur, pour former un chercheur d'emploi choisi durant le processus d'appariement.
4. Le test de notre approche sur un nombre important de documents réels (CVs et offres d'emploi), ce qui nous permettra de juger vraiment son efficacité.
5. Généraliser l'ontologie construite à d'autres domaines.

Glossaire

Ce glossaire regroupe et donne la définition des principaux termes utilisés dans ce mémoire.

IC: Ingénieur des Connaissances.

DTD: Document Type Definition.

DAML+OIL: DARPA Agent Markup Language + Ontology Inference Layer.

HTML: Hypertext Markup Language.

OWL: Ontology Web Language.

RACER: Renamed Abox and Concept Expression Reasoner.

RDF: Resource Description Framework.

RDF(S): Resource Description Framework Schema.

TOVE: TOronto Virtual Enterprise.

URI: Uniform Resource Identifiers.

URL: Uniform Resource Locator.

W3C: World Wide Web Consortium.

WWW: World Wide Web.

XHTML: eXtensible Hypertext Markup Language.

XML: eXtensible Markup Language.

DEUA : diplôme universitaire des études appliquées.

IA : intelligence artificielle.

STIC : science et technologie de l'information et de la communication.

SPD : systèmes parallèles et distribués.

SIA : systèmes d'information avancés.

GL : génie logiciel.

SI : système d'information.

1.Hiérarchie des thèmes de ressources

1.1 Les sous thèmes de « THEME GENERAL »

Animation	Animation de groupe
	Animation de réunion
	Animation et coordination d'un réseau de correspondants fonctionnels et opérationnels d'application
	Méthodes et support de communication
	Méthodes et support de formation
Audit / sécurité	Audit et contrôle de gestion
	Etablissement d'un diagnostic de situation
	Méthodes d'analyse des risques
	Méthodes d'analyse et d'organisation
	Sécurité au sens général
	Sécurité physique et industrielle
	Techniques et procédures d'audit
Budget	Coûts des produits et service
	Gestion économique et financier appliquée à l'informatique
	Gestion et control des couts informatiques et de télécom
	Définition/suivi de budgets et de comptes d'exposition
Conduite De Projet	Direction de projet
	Gestion du changement
	Gestion des conflits
	Vision des impacts organisationnels de projet
	Ingénierie d'accompagnement de projet
	Mener des projets avec des acteurs multiples
	Mise en place des indicateurs de suivi de projet
	Méthodes et techniques de conduite de projet
	Elaboration des scénarios de gestion /conduite projet
Culture D'entreprise	Connaissance de l'environnement de l'entreprise
	Connaissance du milieu de travail
	Connaissance de l'organisation /fonctionnement de l'entreprise
	Connaissance des métiers et activités de l'entreprise
	Connaissance des clients de l'entreprise /utilisateurs SI et leurs besoins
	Connaissance des systèmes d'information d'entreprise
Gestion des Fournisseurs / Achats	Connaissance pour mener une offre commerciale
	Mise en place de dispositifs contractuels
	Gestion des stocks de matériel et approvisionnement
	Gestion des contrats de prestataires

Juridique	Connaissance générale en droit
	Connaissance juridique en droit des contrats
	Réglementation de l'informatique et des télécoms
	Législation sur la sécurité
	Réglementation du personnel
Langues	Langue anglais
	Langue français
	Langue Allmen
Organisation	Conception de solution organisationnelles
	Elaboration des procédures du fonctionnement / utilisation du SI
	Techniques de l'organisation
	Planification
	Organisation du soutien aux utilisateurs
	Organisation de flux
	Vision des impacts organisationnels d'un projet
Processus Métiers	Modélisation et analyse de processus
	Pratique des outils métiers et connaissance des conditions de leur utilisation
	Procédures, normes et standards PM
Bureautique	Pratique de la communication téléphonique et Fax
	Messageries électronique
	Informatique bureautique
	Capacité rédactionnelle
Management	Gestion sociale
	Méthodes de management d'équipe
	Technique de gestion de situations conflictuelles
Electronique	Connaissance/manipulation des circuits électroniques
	Electricité
	Connaissance du fonctionnement du composant électronique
Pédagogie	Critères et techniques d'évaluation
	Connaissance des importants éléments psycho pédagogiques
	Techniques d'enseignement
Mathématique	Algèbre
	Analyse
	Logique mathématique
	Probabilités
	Statistiques

1.2. Les sous thèmes de « THEME SPECIFIQUE »

Algorithmique & Programmation	Algorithmique	Analyse algorithmes	
		Modélisation Problèmes	Algorithmes linéaires
		Algorithmes parallèles/distribués	
	Structures De Données	Types de données simples	
		Structure séquentielles	
		Structure réursive	
		Structure hiérarchique	
		Fichier	
		Graphe	
	Langage Programmation Haut Niveau	Langage événementielle	
		Langage fonctionnelle	Lisp
			Haskell
			Scheme
		Langage logique	Prolog
			Clips
		Langage orienté objet	C++
			JAVA
			Delphi
			Visual Basic
			Smaltolk
Langage impératif		C	
		Pascal	
	Basic		
	Fortran		
	Perl		
Architecture Machine	Analyse performances machine		
	Composants Matériels	Connaissance composant matérielle	
		Connectique	
		Entretien installations techniques	
		Systèmes de câblage	
	Architecture fondamentale de l'ordinateur		
	Interfaçage matériel / logiciel		
	La microprogrammation		
	Types architecture machine		
Intégration de matériel			
Compilation	Théorie des langages		
	Principales de la compilation		
	Principales de l'interprétation		

Gestion d'information	Bases De Données	Méthodes de gestion de données			
		Conception de BDDS	Modèles conception BDDS	Modèle objet	
				Modèle relationnel	
			Technique conception BDDS		
		Développement de BDDS	Outils de développement BDDS		
			Langages de BDDS	SQL	
				PLSQL	
			MySQL		
		Administration de BDDS	Création BDD		
			Manipulation BDD		
			Maintenance BDD		
			Sécurité BDD		
		Type SGBD	SGBD Relationnel	Access	
			SGDB Objet	Oracle	
		Intégration de BDDS			
	Optimisation de BDDS				
	BDDS avancées	BDDS contrainte			
		BDDS distribuées			
		BDDS géographiques			
		BDDS multimédia			
		BDDS XML			
		BDDS parallèles			
	Système d'information	Environnement du SI			
		Modélisation du SI			
		Conception de SI	Processus de conception du SI		
			Méthodes de conception des SI	Méthodes cartésiennes	SADT
					REMORA
				Méthodes systémiques	MERISE
			Méthodes OO	UML	
				OMT	
		Développement du SI	Processus de développement des SIs		
			Techniques de développement des SIs		
Normes et outils de développement des SIs					
Intégration de SIS		Intégration de différents SIs			
		Intégration de nouvelles technologies dans le SIs			
Test / validation SI					
Installation/ maintenance SI		Techniques d'Installation et maintenance du SI			
		Processus d'Installation et maintenance du SI			
Gouvernance SI		Analyse des performances du SI			
	Pilotages des activités du SI				
	Gestion économique et financière du SI				
	Raisonnement statistique applique au SIs				
Sécurité SI					
Sis avancés	Sis basés web				
	Sis coopératifs				
	sis géographiques				
	Sis multimédia				

Réseaux et télécoms	Architecture réseau	Infrastructure de transport des informations
		Les topologies réseau
		Les types de réseaux (LAN,WAN)
	Outils réseau	
	Télécommunication	Théorie de Méthodes
		Traitement du signal et télétraitement
		Architecture de communication
		Protocoles de communication
		Contrôle de communication
	Codage / compression des informations	
	Installation / maintenance réseaux	
	Administration des réseaux	
Les services réseaux		
La sécurité réseaux		
Interface Homme Machine	Composants de l'interface	
	Fonction de l'interface	
	Ergonomie de l'interface	
	Programmation des interfaces	
	Outils de construction d'interfaces	
	Evaluation et configuration d'interfaces	
Exploitation production	Exploitation système d'information	
	Méthodes /outils /normes d'exploitation	
	Système/outils de production	
	techniques/outils de production informatique	
	Techniques Install/ Maintenance des systèmes	Techniques I/M des systèmes d'exploitation
		Techniques I/M des systèmes d'information
		Techniques I/M des systèmes de production
Optimisation des capacités		

Intelligence artificielle	Ingénierie des connaissances	Acquisition des connaissances				
		Représentation des connaissances				
		Les systèmes à base de connaissances				
		Les systèmes experts	Composants du système expert			
			Fonctionnement du système expert			
			Application pratique en système expert			
		Le raisonnement IA	Gestion des connaissances			
			Les types de raisonnement en IA			
		La logique floue				
		Reconnaissance des formes et systèmes complexes	La vision artificielle			
			L'optimisation combinatoire			
			Les systèmes complexes	Approches conception SC		
				Mécanismes génération SC	Réseaux de neurone	
					Systèmes de classification	
					Programmation génétique	
	Via artificielle / robotique					
	Les systèmes multi – agents	Architecture d'un SMA				
		Les types d'agents dans les SMAs				
		Coopération SMAs				
		Négociation SMAs				
		Palification SMAs				
		Langue programmation agent				
		Application pratique en SMAs				
	SYSTEME	Système d'exploitation	Architecture SE	Architecture système centralisé		
				Architecture système distribué		
			Fonctions du SE	Gestion de la mémoire		
				Allocation du processeur		
Gestion des fichiers						
Gestion des E/S						
Gestion des interruptions						
Gestion des processus						
Programmation système						
Configuration des systèmes d'exploitation						
Installation / maintenance des SEs			Analyse des performances du			
			Installation des SEs			
			Maintenance des SEs centralisés			
			Maintenance des SEs distribués			
Techniques d'installation /maintenance des Ses						
Intégration des systèmes d'exploitation						
Sécurité du système d'exploitation						
Systèmes d'exploitation utilisés		UNIX				
		MS DOS				
		LINUX				
		Windows Xp				
		Windows NT				
		Windows 2000 Serveur				
Système avancés		Systèmes parallèles				
		Systèmes temps réel				
		Systèmes concurrent distribués				

Génie Logiciel	Besoins Logiciel	Spécification des besoins	
		Validation des besoins	
		Gestion des besoins	
		Outils modélisation/gestion des besoins	
	Conception logiciel	Notions de base de la conception logiciel	
		Issues clés de la conception Logiciel	La concurrence/interaction
			Control des événements
			Distribution des composants
			Prise en charge des exceptions /erreur
		Architecture et structure logiciel	Structures architecturales/points vue
			Patrons de conception
		Evaluation de la conception logiciel	
		Description structurale et comportementale	
		Outils de conception logiciel	
		Méthodes / de conception logiciel	Conception orientée fonction
			Conception centrée structure données
	Conception orienté objet		
	Conception basée composants		
	Réalisation logiciel	Notions de bases de la réalisation logiciel	
		Techniques de programmation (héritée)	
		Intégration de logiciels	
		Sécurité d'applications	
		Paramétrage d'applications	
		Outils de développements logiciels	
		Techniques de développement (prototypage, rad, objet...)	
		Application pratique en développement d'application	
	Test /validation De logiciel	Notions de bases du test logiciel	
		Techniques de test	
		Niveaux de test	
		Mesures reliée au test	
		Processus de test outils de test logiciel	
		Outils de test logiciel	
	Maintenance du logiciel	Notions de base de la maintenance logiciel	
		Evaluation/ estimation du cout de la maintenance logiciel	
		Processus de maintenance logiciel	
		Techniques de maintenance logiciel	
	Gestion configuration logiciel	Outils maintenance logiciel	
		Identification de configuration logiciel	
		Contrôle de la configuration logiciel	
		Audit configuration logiciel	
Distribution et gestion des versions du logiciel			
Gestion d'ingénierie logiciel	Outils de gestion de configuration logiciel		
	Définition de la portée du logiciel		
	Planification du projet du logiciel		
	Mise en place du projet logiciel		
	Evaluation du logiciel		
	Clôture du projet logiciel		
	Mesure de l'ingénierie logiciel		
Outils de gestion d'ingénierie logiciel			

Génie Logiciel	Processus D'ingénierie Logiciel	Définition du processus IL	Modèles de cycle vie logiciel	
			Processus de cycle de vie logiciel	
			Notations pour définition du processus	
			Adaptation du processus IL	
			Automatisation	
		Evaluation du Processus	Modèles d'évaluation du processus IL	
			Méthodes évaluation processus IL	
		Implémentation du processus Ilet changement		
		Mesure du produit er du processus		
	Outils du processus d'ingénierie logiciel (case...)			
	Méthodes D'ingénierie Logiciel	Méthodes Heuristiques	Méthodes structurées	
			Méthodes orientées données	
			Méthodes orientées objet	
		Méthodes Formelles	Langages de spécification	
			Raffinement	
			Vérification/preuve des propriétés	
		Méthodes De Prototypage	Les styles	
			Cible de prototypage	
			Techniques d'évaluation	
	Qualité Du Logiciel	Concepts de base de la qualité logiciel		
		Outils de qualité logiciel		
Processus de Gestion de la qualité logiciel		Assurance de la qualité logiciel		
		Vérification et validation logiciel		
		Revue et audit de la qualité logiciel		
Considération Pratiques De la qualité logiciel		Besoins de la qualité d'application		
		Caractérisation de défaut		
		Techniques gestion qualité logiciel		
Mesure qualité logiciel				
Traitement sens et image	Les données multimédia	Les types de données multimédia		
		Représentation des données multimédia		
		Manipulation des données multimédia		
		Compression / stockage des données multimédia		
	Traitement d'image	Les systèmes graphiques		
		La modélisation géométrique		
		Les techniques de base de l'infographie		
		Les outils de traitement d'image		
		La réalité virtuelle		
	Traitement du sens	Codification du sens		
		Techniques de traitement du sens		
		Traitement de la parole		

Technologie Web	Web Classique	moteur de recherche		
		Développement d'application web	HTML	
			PHP	
			ASP	
			Java script	
	VB script			
	La sécurité web			
	Web Sémantique	Représentation Des Connaissance En WS	Formalismes de représentation Du WS	Frames
				Graphes conceptuels
				Réseaux sémantiques
				Logique de description
			Les ontologies	Eléments d'ontologie
				Construction D'ontologie
		Intégration d'ontologies		
Maintenance D'ontologies				
Langages du WS		XML		
		XML schéma		
		RDF		
		RDFS		
	DAML+OIL			
	OWL			

2. Les métiers du domaine « Informatique et Télécommunication » et leurs familles

<i>Familles Métiers</i>	<i>Métiers</i>
Conseil en SI Et maîtrise D'ouvrage	Consultant en SI, Architecte SI, Chef projet maîtrise d'ouvrage, Responsable Du SI, Responsable de projet.
Support/ Assistance aux utilisateurs	Assistant Fonctionnel, Technicien support SVP, Chargé d'affaires internes.
Production/ Exploitation	Technicien D'exploitation, Technicien poste de travail, Technicien réseaux/télécoms, Administrateur d'outils /systèmes/réseaux et télécom, Administrateur BDD Intégrateur d'exploitation, Pilote D'exploitation.
Etude/ Développement /intégration	Chef projet maîtrise d'œuvre, Développeur (Analyste programmeur), Intégrateur d'applications, Parametreur d'ERP.
Assistance Technique interne	Expert système D'exploitation, Expert réseaux /télécoms, Architecte Technique, Responsable sécurité SI, Expert méthodes Et outils/ Qualité/sécurité, Expert de technologie Internet/intranet et multimédia.
Administration /Gestion	Responsable du Management, Responsable D'une entité Informatique, Responsable Service administratif et finance, Responsable Exploitation Informatique, Responsable Télécoms
Enseignement /Recherche	Enseignant En post Graduation, Enseignant En Graduation, Enseignant École, Poste Scientifique.

3. Les diplôme domaine « Informatique et Télécommunication » et leurs familles

Familles Diplômes	Diplômes
Baccalauréat	Bachelier.
DEUA	DEUA en Système d'information. DEUA en Réseaux/ Télécoms.
Ingénieur	Ingénieur d'état SPD, Ingénieur d'état SIA, Ingénieur d'état en IA, Ingénieur en Informatique Industrielle.
Licence	Licence Académique, Licence Professionnelle GL, Licence Professionnelle SI, Licence Professionnelle STIC.
Master	Master Option SI, Master Option IA.
Magister	Magister Option Réseaux, Magister Option GL, Magister Option Système, Magister Option SIA, Magister Option Information&Computation.
Doctorat	Docteur En IA, Docteur En système, Docteur En SI, Docteur En Réseaux.

4. les types de contrat : les types d'emploi sont :

- Intérim.
- Stage.
- Titulaire.

5. les types d'emploi : les types d'emploi sont :

- Temps pleins.
- Temps partiel.
- Journalier

Remarque : pour le contexte organisationnel le nombre minimal et maximal de personnel dans un groupe ou service selon le type de ces derniers est détaillé dans le tableau suivant :

	Le nombre minimal personnel	Le nombre maximal personnel
Grand service		
Service moyen		
Petit service		
Grand group		
Group moyen		
Petit group		

Concernent le contexte Economique, le chiffre d'affaire minimal et maximal pour chaque type d'entreprise est détaillé dans le tableau suivants :

	chiffre d'affaire minimal	chiffre d'affaire maximal
Grande entreprise	18,620 milliards de dollars	
Entreprise moyenne		
Petite entreprise		

Un extrait du code OWL de l'ontologie ER

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://www.owl-ontologies.com/unnamed.owl#"
  xml:base="http://www.owl-ontologies.com/unnamed.owl">
  <owl:Ontology rdf:about="" />
  <owl:Class rdf:ID="EfficacitéPersonnelle">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Aptitude" />
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="AssistanceEtService">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Aptitude" />
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="CultureEntreprise">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="ThèmeGénéral" />
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Animation">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#ThèmeGénéral" />
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="RéférentielDiplomes">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="réfèreDiplôme" />
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing" />
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:cardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="liéA" />
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

```

```

    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="ressource">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Habilité"/>
        <owl:Class rdf:ID="Savoir"/>
        <owl:Class rdf:ID="SavoirFaire"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="Thème">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#ThèmeGénéral"/>
        <owl:Class rdf:ID="ThèmeSpécifique"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="Salaire">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Exigence"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Compétence">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="aContexte"/>
      </owl:onProperty>
      <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >2</owl:minCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Aptitude"/>
        <owl:Class rdf:ID="CompétenceScientifiqueTechnique"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>

```

```
</owl:Class>
</owl:equivalentClass>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:ObjectProperty rdf:ID="aNiveau"/>
    </owl:onProperty>
    <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >1</owl:minCardinality>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Bureautique">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeGénéral"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="RéseauxEtTélécom">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="ServiceMilitaire">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="QualificationPersonnelle"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="SituationFamill">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#QualificationPersonnelle"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Savoir">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#ressource"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:about="#Habilité"/>
          </owl:complementOf>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <owl:Class>
    <owl:complementOf>
      <owl:Class rdf:about="#SavoirFaire"/>
    </owl:complementOf>
  </owl:Class>
</owl:Class>
```

```

    </owl:Class>
  </owl:intersectionOf>
</owl:Class>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="#ressource"/>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:ObjectProperty rdf:ID="aThème"/>
    </owl:onProperty>
    <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="GestionFournisseursAchats">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeGénéral"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="TypeEmploi">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Exigence"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Rotation">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Exigence"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#QualificationPersonnelle">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Age"/>
        <owl:Class rdf:ID="Nationalité"/>
        <owl:Class rdf:ID="PermisConduire"/>
        <owl:Class rdf:about="#ServiceMilitaire"/>
        <owl:Class rdf:ID="Sexe"/>
        <owl:Class rdf:about="#SituationFamill"/>
        <owl:Class rdf:ID="VileRésidence"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="IntelligenceArtificielle">
  <rdfs:subClassOf>

```

```

<owl:Class rdf:about="#ThèmeSpécifique"/>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#CompétenceScientifiqueTechnique">
<rdfs:subClassOf>
<owl:Class>
<owl:intersectionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Compétence"/>
<owl:Class>
<owl:complementOf>
<owl:Class rdf:about="#Aptitude"/>
</owl:complementOf>
</owl:Class>
</owl:intersectionOf>
</owl:Class>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="#Compétence"/>
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty>
<owl:ObjectProperty rdf:ID="mobiliseRessou"/>
</owl:onProperty>
<owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int ">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Contexte">
<owl:equivalentClass>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:ID="Economique"/>
<owl:Class rdf:ID="Organisationnel"/>
</owl:unionOf>
</owl:Class>
</owl:equivalentClass>
</owl:Class>
<owl:Class rdf:about="#SavoirFaire">
<rdfs:subClassOf>
<owl:Restriction>
<owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:cardinality>
<owl:onProperty>
<owl:ObjectProperty rdf:about="#aThème"/>
</owl:onProperty>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="#ressource"/>

```

```

<rdfs:subClassOf>
  <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#ressource"/>
      <owl:Class>
        <owl:complementOf>
          <owl:Class rdf:about="#Habilité"/>
        </owl:complementOf>
      </owl:Class>
      <owl:Class>
        <owl:complementOf rdf:resource="#Savoir"/>
      </owl:Class>
    </owl:intersectionOf>
  </owl:Class>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Langues">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeGénéral"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Diplome">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="mobilise"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="aFamilDiplôme"/>
      </owl:onProperty>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:about="#Nationalité">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:ID="AuditSécurité">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeGénéral"/>
  </rdfs:subClassOf>

```

```
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="GénieLogiciel">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Famildiplome"/>
<owl:Class rdf:ID="Organisation">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeGénéral"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="ArchitectureMachine">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#PermisConduire">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:ID="Système">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="GestionDInformation">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#VileRésidence">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:ID="Juridique">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeGénéral"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="InterfaceHommeMachine">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="TraitementSensEtImage">
  <rdfs:subClassOf>
```

```

    <owl:Class rdf:about="#ThèmeSpécifique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#ThèmeGénéral">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Thème"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:about="#ThèmeSpécifique"/>
          </owl:complementOf>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#Thème"/>
</owl:Class>
<owl:Class rdf:ID="ActionEtRéalisation">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Aptitude"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#ThèmeSpécifique">
  <rdfs:subClassOf rdf:resource="#Thème"/>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Thème"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#ThèmeGénéral"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="RéférentielMétiers">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:cardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#liéA"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>

```

```

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
    <owl:onProperty>
      <owl:ObjectProperty rdf:ID="réfèreMétier"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Organisationnel">
  <rdfs:subClassOf rdf:resource="#Contexte"/>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:about="#Economique"/>
          </owl:complementOf>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Sexe">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:about="#Exigence">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Contrat"/>
        <owl:Class rdf:ID="Déplacement"/>
        <owl:Class rdf:about="#Rotation"/>
        <owl:Class rdf:about="#Salaire"/>
        <owl:Class rdf:about="#TypeEmploi"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:about="#Contrat">
  <rdfs:subClassOf rdf:resource="#Exigence"/>
</owl:Class>
<owl:Class rdf:ID="Métier">
  <rdfs:subClassOf>
    <owl:Restriction>

```

```

<owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
<owl:onProperty>
  <owl:ObjectProperty rdf:ID="aFamilMétier"/>
</owl:onProperty>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:ID="Pédagogie">
  <rdfs:subClassOf rdf:resource="#ThèmeGénéral"/>
</owl:Class>
<owl:Class rdf:ID="Domaine"/>
<owl:Class rdf:about="#Economie">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#Organisationnel"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#Contexte"/>
</owl:Class>
<owl:Class rdf:ID="ExploitationProduction">
  <rdfs:subClassOf rdf:resource="#ThèmeSpécifique"/>
</owl:Class>
<owl:Class rdf:ID="Electronique">
  <rdfs:subClassOf rdf:resource="#ThèmeGénéral"/>
</owl:Class>
<owl:Class rdf:about="#Age">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:about="#Aptitude">
  <rdfs:subClassOf rdf:resource="#Compétence"/>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#CompétenceScientifiqueTechnique"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>

```

```

<owl:equivalentClass>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#ActionEtRéalisation"/>
      <owl:Class rdf:ID="AptitudManagérial"/>
      <owl:Class rdf:about="#EfficacitéPersonnelle"/>
      <owl:Class rdf:about="#AssistanceEtService"/>
    </owl:unionOf>
  </owl:Class>
</owl:equivalentClass>
</owl:Class>
<owl:Class rdf:about="#AptitudManagérial">
  <rdfs:subClassOf rdf:resource="#Aptitude"/>
</owl:Class>
<owl:Class rdf:ID="AlgoProgrammation">
  <rdfs:subClassOf rdf:resource="#ThèmeSpécifique"/>
</owl:Class>
<owl:Class rdf:ID="TechnologiesWeb">
  <rdfs:subClassOf rdf:resource="#ThèmeSpécifique"/>
</owl:Class>
<owl:Class rdf:ID="Compilation">
  <rdfs:subClassOf rdf:resource="#ThèmeSpécifique"/>
</owl:Class>
<owl:Class rdf:about="#Déplacement">
  <rdfs:subClassOf rdf:resource="#Exigence"/>
</owl:Class>
<owl:Class rdf:ID="Magement">
  <rdfs:subClassOf rdf:resource="#ThèmeGénéral"/>
</owl:Class>
<owl:Class rdf:about="#Habilité">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#ressource"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#Savoir"/>
        </owl:Class>
        <owl:Class>
          <owl:complementOf rdf:resource="#SavoirFaire"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#ressource"/>
</owl:Class>
<owl:Class rdf:ID="Budget">

```

```

<rdfs:subClassOf rdf:resource="#ThèmeGénéral"/>
</owl:Class>
<owl:Class rdf:ID="Niveau"/>
<owl:Class rdf:ID="FamillMétier"/>
<owl:Class rdf:ID="ConduiteDeProjets">
  <rdfs:subClassOf rdf:resource="#ThèmeGénéral"/>
</owl:Class>
<owl:Class rdf:ID="Document"/>
<owl:ObjectProperty rdf:about="#aFamillDiplôme">
  <rdfs:range rdf:resource="#FamillDiplome"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="famillDiplômeDe"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#Diplome"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aContexte">
  <rdfs:range rdf:resource="#Contexte"/>
  <rdfs:domain rdf:resource="#Compétence"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="concerne"/>
  </owl:inverseOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreMétier">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="réfèreParRM"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#RéférentielMétiers"/>
  <rdfs:range rdf:resource="#Métier"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreDiplôme">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="réfèreParRD"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#RéférentielDiplomes"/>
  <rdfs:range rdf:resource="#Diplome"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aFamillMétier">
  <rdfs:range rdf:resource="#FamillMétier"/>
  <rdfs:domain rdf:resource="#Métier"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="famillMétierDe"/>
  </owl:inverseOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#mobiliseRessou">
  <rdfs:domain rdf:resource="#CompétenceScientifiqueTechnique"/>
  <owl:inverseOf>

```

```
<owl:ObjectProperty rdf:ID="ressouEstMobilisePar"/>
</owl:inverseOf>
<rdfs:range rdf:resource="#ressource"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreParRM">
  <rdfs:domain rdf:resource="#Métier"/>
  <owl:inverseOf rdf:resource="#réfèreMétier"/>
  <rdfs:range rdf:resource="#RéférentielMétiers"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#familDiplômeDe">
  <rdfs:domain rdf:resource="#FamilDiplome"/>
  <rdfs:range rdf:resource="#Diplome"/>
  <owl:inverseOf rdf:resource="#aFamilDiplôme"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aThème">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#SavoirFaire"/>
        <owl:Class rdf:about="#Savoir"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="#Thème"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#concerne">
  <owl:inverseOf rdf:resource="#aContexte"/>
  <rdfs:range rdf:resource="#Compétence"/>
  <rdfs:domain rdf:resource="#Contexte"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreParRD">
  <rdfs:domain rdf:resource="#Diplome"/>
  <rdfs:range rdf:resource="#RéférentielDiplomes"/>
  <owl:inverseOf rdf:resource="#réfèreDiplôme"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#familMétierDe">
  <rdfs:domain rdf:resource="#FamilMétier"/>
  <owl:inverseOf rdf:resource="#aFamilMétier"/>
  <rdfs:range rdf:resource="#Métier"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aNiveau">
  <rdfs:range rdf:resource="#Niveau"/>
  <rdfs:domain rdf:resource="#Compétence"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#ressouEstMobilisePar">
  <rdfs:range rdf:resource="#CompétenceScientifiqueTechnique"/>
```

```

<owl:inverseOf rdf:resource="#mobiliseRessou"/>
<rdfs:domain rdf:resource="#ressource"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#mobilise">
<rdfs:range rdf:resource="#Compétence"/>
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Diplome"/>
<owl:Class rdf:about="#Métier"/>
</owl:unionOf>
</owl:Class>
</rdfs:domain>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#liéA">
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#RéférentielDiplomes"/>
<owl:Class rdf:about="#RéférentielMétiers"/>
</owl:unionOf>
</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="#Domaine"/>
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="maxAge">
<rdfs:domain rdf:resource="#Age"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="identification">
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#RéférentielDiplomes"/>
<owl:Class rdf:about="#RéférentielMétiers"/>
</owl:unionOf>
</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="expFrançais">
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#RéférentielDiplomes"/>
<owl:Class rdf:about="#RéférentielMétiers"/>

```

```

    <owl:Class rdf:about="#Compétence"/>
    <owl:Class rdf:about="#Diplome"/>
    <owl:Class rdf:about="#Métier"/>
  </owl:unionOf>
</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="sexe">
  <rdfs:domain rdf:resource="#Sexe"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="accepté">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Déplacement"/>
        <owl:Class rdf:about="#Rotation"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="nationalité">
  <rdfs:domain rdf:resource="#Nationalité"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="type">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contrat"/>
        <owl:Class rdf:about="#TypeEmploi"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="age">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdfs:domain rdf:resource="#Age"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="marié">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#SituationFamill"/>
</owl:DatatypeProperty>

```

```

<owl:DatatypeProperty rdf:ID="nom">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Diplome"/>
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class rdf:about="#Métier"/>
        <owl:Class rdf:about="#ressource"/>
        <owl:Class rdf:about="#Thème"/>
        <owl:Class rdf:about="#RéférentielDiplomes"/>
        <owl:Class rdf:about="#RéférentielMétiers"/>
        <owl:Class rdf:about="#FamilMétier"/>
        <owl:Class rdf:about="#FamilDiplome"/>
        <owl:Class rdf:about="#Domaine"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="dégagé">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#ServiceMilitaire"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="auteur">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#RéférentielDiplomes"/>
        <owl:Class rdf:about="#RéférentielMétiers"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="valeurMin">
  <rdfs:domain rdf:resource="#Contexte"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="montant">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdfs:domain rdf:resource="#Salaire"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="catégorie">
  <rdfs:domain rdf:resource="#PermisConduire"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>

```

```

</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="poids">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Exigence"/>
        <owl:Class rdf:about="#QualificationPersonnelle"/>
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class rdf:about="#Contexte"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="villeR">
  <rdfs:domain rdf:resource="#VileRésidence"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="désignation">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Niveau"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="UriDocument">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Document"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="maxDif">
  <rdfs:domain rdf:resource="#Salaire"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="valeur">
  <rdfs:domain rdf:resource="#Niveau"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="maxDa">
  <rdfs:domain rdf:resource="#VileRésidence"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="valeurMax">
  <rdfs:domain rdf:resource="#Contexte"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="typDocument">
  <rdfs:domain rdf:resource="#Document"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

```

```
<owl:DatatypeProperty rdf:ID="expAnglais">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class rdf:about="#Diplome"/>
        <owl:Class rdf:about="#Métier"/>
        <owl:Class rdf:about="#FamilMétier"/>
        <owl:Class rdf:about="#FamilDiplome"/>
        <owl:Class rdf:about="#Domaine"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</owl:DatatypeProperty>
</rdf:RDF>
```

<!-- Created with Protege (with OWL Plugin 2.1, Build 284) <http://protege.stanford.edu> -->

Un extraie du fichier d'annotation de l'offre1

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://www.owl-ontologies.com/unnamed.owl#"
  xml:base="http://www.owl-ontologies.com/unnamed.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="EfficacitéPersonnelle">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Aptitude"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="AssistanceEtService">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Aptitude"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="RéférentielDiplomes">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="réfèreDiplôme"/>
        </owl:onProperty>
        <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="liéA"/>
        </owl:onProperty>
        <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Thème">
    <owl:equivalentClass>
      <owl:Class>
```

```

<owl:unionOf rdf:parseType="Collection">
  <owl:Class rdf:ID="ThèmeGénéral"/>
  <owl:Class rdf:ID="ThèmeSpécifique"/>
</owl:unionOf>
</owl:Class>
</owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="ressource">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Habilité"/>
        <owl:Class rdf:ID="Savoir"/>
        <owl:Class rdf:ID="SavoirFaire"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="Salaire">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Exigence"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Compétence">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="aNiveau"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Aptitude"/>
        <owl:Class rdf:ID="CompétenceScientifiqueTechnique"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >2</owl:minCardinality>

```

```

<owl:onProperty>
  <owl:ObjectProperty rdf:ID="aContexte"/>
</owl:onProperty>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:about="#ThèmeGénéral">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Thème"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:about="#ThèmeSpécifique"/>
          </owl:complementOf>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#Thème"/>
</owl:Class>
<owl:Class rdf:ID="ActionEtRéalisation">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Aptitude"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#ThèmeSpécifique">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Thème"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#ThèmeGénéral"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#Thème"/>
</owl:Class>
<owl:Class rdf:ID="RéférentielMétiers">
  <rdfs:subClassOf>
    <owl:Restriction>

```

```

<owl:onProperty>
  <owl:ObjectProperty rdf:ID="réfèreMétier"/>
</owl:onProperty>
<owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#liéA"/>
    </owl:onProperty>
    <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:cardinality>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Organisationnel">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Contexte"/>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:ID="Economique"/>
          </owl:complementOf>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Sexe">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="QualificationPersonnelle"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Exigence">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Contrat"/>

```

```

    <owl:Class rdf:ID="Déplacement"/>
    <owl:Class rdf:ID="Rotation"/>
    <owl:Class rdf:about="#Salaire"/>
    <owl:Class rdf:ID="TypeEmploi"/>
  </owl:unionOf>
</owl:Class>
</owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="ServiceMilitaire">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#QualificationPersonnelle"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Contrat">
  <rdfs:subClassOf rdf:resource="#Exigence"/>
</owl:Class>
<owl:Class rdf:ID="Métier">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="aFamilMétier"/>
      </owl:onProperty>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:ID="SituationFamil">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#QualificationPersonnelle"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Savoir">
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#ressource"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:about="#Habilité"/>
          </owl:complementOf>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class>

```

```
<owl:complementOf>
  <owl:Class rdf:about="#SavoirFaire"/>
</owl:complementOf>
</owl:Class>
</owl:intersectionOf>
</owl:Class>
</rdfs:subClassOf>
<rdfs:subClassOf rdf:resource="#ressource"/>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
    <owl:onProperty>
      <owl:ObjectProperty rdf:ID="aThème"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Domaine"/>
<owl:Class rdf:about="#Economique">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Contexte"/>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#Organisationnel"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Age">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#QualificationPersonnelle"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Aptitude">
  <rdfs:subClassOf rdf:resource="#Compétence"/>
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
```

```

<owl:Class rdf:about="#ActionEtRéalisation"/>
<owl:Class rdf:ID="AptitudManagérial"/>
<owl:Class rdf:about="#EfficacitéPersonnelle"/>
<owl:Class rdf:about="#AssistanceEtService"/>
</owl:unionOf>
</owl:Class>
</owl:equivalentClass>
<rdfs:subClassOf>
<owl:Class>
<owl:intersectionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Compétence"/>
<owl:Class>
<owl:complementOf>
<owl:Class rdf:about="#CompétenceScientifiqueTechnique"/>
</owl:complementOf>
</owl:Class>
</owl:intersectionOf>
</owl:Class>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#AptitudManagérial">
<rdfs:subClassOf rdf:resource="#Aptitude"/>
</owl:Class>
<owl:Class rdf:about="#TypeEmploi">
<rdfs:subClassOf rdf:resource="#Exigence"/>
</owl:Class>
<owl:Class rdf:about="#Rotation">
<rdfs:subClassOf rdf:resource="#Exigence"/>
</owl:Class>
<owl:Class rdf:about="#QualificationPersonnelle">
<owl:equivalentClass>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Age"/>
<owl:Class rdf:ID="Nationalité"/>
<owl:Class rdf:ID="PermisConduire"/>
<owl:Class rdf:about="#ServiceMilitaire"/>
<owl:Class rdf:about="#Sexe"/>
<owl:Class rdf:about="#SituationFamil"/>
<owl:Class rdf:ID="VileRésidence"/>
</owl:unionOf>
</owl:Class>
</owl:equivalentClass>

```

```

</owl:Class>
<owl:Class rdf:about="#Déplacement">
  <rdfs:subClassOf rdf:resource="#Exigence"/>
</owl:Class>
<owl:Class rdf:about="#Contexte">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Economique"/>
        <owl:Class rdf:about="#Organisationnel"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:about="#SavoirFaire">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#aThème"/>
      </owl:onProperty>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#ressource"/>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#ressource"/>
        <owl:Class>
          <owl:complementOf>
            <owl:Class rdf:about="#Habilité"/>
          </owl:complementOf>
        </owl:Class>
        <owl:Class>
          <owl:complementOf rdf:resource="#Savoir"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#CompétenceScientifiqueTechnique">
  <rdfs:subClassOf rdf:resource="#Compétence"/>
  <rdfs:subClassOf>

```

```

<owl:Restriction>
  <owl:onProperty>
    <owl:ObjectProperty rdf:ID="mobiliseRessou"/>
  </owl:onProperty>
  <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Compétence"/>
      <owl:Class>
        <owl:complementOf rdf:resource="#Aptitude"/>
      </owl:Class>
    </owl:intersectionOf>
  </owl:Class>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Habilité">
  <rdfs:subClassOf rdf:resource="#ressource"/>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#ressource"/>
        <owl:Class>
          <owl:complementOf rdf:resource="#Savoir"/>
        </owl:Class>
        <owl:Class>
          <owl:complementOf rdf:resource="#SavoirFaire"/>
        </owl:Class>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Diplome">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</owl:minCardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="mobilise"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

```

```

<rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >1</owl:cardinality>
    <owl:onProperty>
      <owl:ObjectProperty rdf:ID="aFamildiplôme"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Nationalité">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:ID="Niveau"/>
<owl:Class rdf:ID="Famildiplome"/>
<owl:Class rdf:ID="Familmétier"/>
<owl:Class rdf:about="#PermisConduire">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:about="#VileRésidence">
  <rdfs:subClassOf rdf:resource="#QualificationPersonnelle"/>
</owl:Class>
<owl:Class rdf:ID="Document"/>
<owl:ObjectProperty rdf:about="#aFamildiplôme">
  <rdfs:range rdf:resource="#Famildiplome"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="famildiplômeDe"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#Diplome"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aContexte">
  <rdfs:range rdf:resource="#Contexte"/>
  <rdfs:domain rdf:resource="#Compétence"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="concerne"/>
  </owl:inverseOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreMétier">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="réfèreParRM"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#RéférentielMétiers"/>
  <rdfs:range rdf:resource="#Métier"/>

```

```

</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreDiplôme">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="réfèreParRD"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#RéfèrentielDiplomes"/>
  <rdfs:range rdf:resource="#Diplome"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aFamilMétier">
  <rdfs:domain rdf:resource="#Métier"/>
  <rdfs:range rdf:resource="#FamilMétier"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="familMétierDe"/>
  </owl:inverseOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#mobiliseRessou">
  <rdfs:domain rdf:resource="#CompétenceScientifiqueTechnique"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="ressouEstMobilisePar"/>
  </owl:inverseOf>
  <rdfs:range rdf:resource="#ressource"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreParRM">
  <rdfs:domain rdf:resource="#Métier"/>
  <owl:inverseOf rdf:resource="#réfèreMétier"/>
  <rdfs:range rdf:resource="#RéfèrentielMétiers"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#familDiplômeDe">
  <rdfs:domain rdf:resource="#FamilDiplome"/>
  <rdfs:range rdf:resource="#Diplome"/>
  <owl:inverseOf rdf:resource="#aFamilDiplôme"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aThème">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#SavoirFaire"/>
        <owl:Class rdf:about="#Savoir"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="#Thème"/>
</owl:ObjectProperty>

```

```

<owl:ObjectProperty rdf:about="#concerne">
  <owl:inverseOf rdf:resource="#aContexte"/>
  <rdfs:range rdf:resource="#Compétence"/>
  <rdfs:domain rdf:resource="#Contexte"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#réfèreParRD">
  <rdfs:domain rdf:resource="#Diplome"/>
  <rdfs:range rdf:resource="#RéférentielDiplomes"/>
  <owl:inverseOf rdf:resource="#réfèreDiplôme"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#familMétierDe">
  <owl:inverseOf rdf:resource="#aFamilMétier"/>
  <rdfs:domain rdf:resource="#FamilMétier"/>
  <rdfs:range rdf:resource="#Métier"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#aNiveau">
  <rdfs:range rdf:resource="#Niveau"/>
  <rdfs:domain rdf:resource="#Compétence"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#ressouEstMobilisePar">
  <rdfs:range rdf:resource="#CompétenceScientifiqueTechnique"/>
  <owl:inverseOf rdf:resource="#mobiliseRessou"/>
  <rdfs:domain rdf:resource="#ressource"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#mobilise">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Diplome"/>
        <owl:Class rdf:about="#Métier"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="#Compétence"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#liéA">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#RéférentielDiplomes"/>
        <owl:Class rdf:about="#RéférentielMétiers"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="#Compétence"/>
</owl:ObjectProperty>

```

```
</rdfs:domain>
<rdfs:range rdf:resource="#Domaine"/>
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="maxAge">
  <rdfs:domain rdf:resource="#Age"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="identification">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#RéférentielDiplomes"/>
        <owl:Class rdf:about="#RéférentielMétiers"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="expFrançais">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#RéférentielDiplomes"/>
        <owl:Class rdf:about="#RéférentielMétiers"/>
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class rdf:about="#Diplome"/>
        <owl:Class rdf:about="#Métier"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="sexe">
  <rdfs:domain rdf:resource="#Sexe"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="accepté">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Déplacement"/>
        <owl:Class rdf:about="#Rotation"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

```

</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="nationalité">
<rdfs:domain rdf:resource="#Nationalité"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="type">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Contrat"/>
<owl:Class rdf:about="#TypeEmploi"/>
</owl:unionOf>
</owl:Class>
</rdfs:domain>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="age">
<rdfs:domain rdf:resource="#Age"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="marié">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdfs:domain rdf:resource="#SituationFamil"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="nom">
<rdfs:domain>
<owl:Class>
<owl:unionOf rdf:parseType="Collection">
<owl:Class rdf:about="#Diplome"/>
<owl:Class rdf:about="#Contexte"/>
<owl:Class rdf:about="#Compétence"/>
<owl:Class rdf:about="#Métier"/>
<owl:Class rdf:about="#ressource"/>
<owl:Class rdf:about="#Thème"/>
<owl:Class rdf:about="#RéférentielDiplomes"/>
<owl:Class rdf:about="#RéférentielMétiers"/>
<owl:Class rdf:about="#FamilMétier"/>
<owl:Class rdf:about="#FamilDiplome"/>
<owl:Class rdf:about="#Domaine"/>
</owl:unionOf>

```

```
</owl:Class>
</rdfs:domain>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="dégagé">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#ServiceMilitaire"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="auteur">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#RéférentielDiplomes"/>
        <owl:Class rdf:about="#RéférentielMétiers"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="valeurMin">
  <rdfs:domain rdf:resource="#Contexte"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="montant">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdfs:domain rdf:resource="#Salaire"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="catégorié">
  <rdfs:domain rdf:resource="#PermisConduire"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="poids">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Exigence"/>
        <owl:Class rdf:about="#QualificationPersonnelle"/>
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class rdf:about="#Contexte"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
```

```
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="désignation">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Niveau"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="villeR">
  <rdfs:domain rdf:resource="#VileRésidence"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="UriDocument">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Document"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="valeur">
  <rdfs:domain rdf:resource="#Niveau"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="maxDif">
  <rdfs:domain rdf:resource="#Salaire"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="maxDa">
  <rdfs:domain rdf:resource="#VileRésidence"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="valeurMax">
  <rdfs:domain rdf:resource="#Contexte"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="typDocument">
  <rdfs:domain rdf:resource="#Document"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="expAnglais">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Contexte"/>
        <owl:Class rdf:about="#Compétence"/>
        <owl:Class rdf:about="#Diplome"/>
        <owl:Class rdf:about="#Métier"/>
        <owl:Class rdf:about="#FamilMétier"/>
        <owl:Class rdf:about="#FamilDiplome"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>

```

```

    <owl:Class rdf:about="#Domaine"/>
  </owl:unionOf>
</owl:Class>
</rdfs:domain>
</owl:DatatypeProperty>
<Economie rdf:ID="eco_graEnt">
  <nom xml:lang="fr">grandeEntreprise</nom>
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int">0</poids>
  <concerne>
    <CompétenceScientifiqueTechnique rdf:ID="comp_cplus">
      <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int">50</poids>
      <nom xml:lang="fr">cPlus</nom>
      <aContexte>
        <Organisationnel rdf:ID="org_graSer">
          <concerne>
            <EfficacitéPersonnelle rdf:ID="eff_résistancePressions">
              <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >30</poids>
              <aContexte rdf:resource="#org_graSer"/>
              <nom xml:lang="fr">résistanceAuxPréssions</nom>
              <aContexte rdf:resource="#eco_graEnt"/>
              <aNiveau>
                <Niveau rdf:ID="FO">
                  <désignation xml:lang="fr">forte</désignation>
                  <valeur rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >0</valeur>
                </Niveau>
              </aNiveau>
            </EfficacitéPersonnelle>
          </concerne>
          <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >0</poids>
          <nom xml:lang="fr">grandeService</nom>
          <concerne rdf:resource="#comp_cplus"/>
        </Organisationnel>
      </aContexte>
      <aContexte rdf:resource="#eco_graEnt"/>
      <aNiveau>
        <Niveau rdf:ID="C">
          <valeur rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >0</valeur>
          <désignation xml:lang="fr">compétent</désignation>
        </Niveau>
      </aNiveau>
    </CompétenceScientifiqueTechnique>
  </concerne>
  <concerne rdf:resource="#eff_résistancePressions"/>

```

```
</Economie>
<Nationalité rdf:ID="Nationalité3">
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >5</poids>
  <nationalité xml:lang="fr">algérien</nationalité>
</Nationalité>
<TypeEmploi rdf:ID="TypeEmploi3">
  <type xml:lang="fr">tempsPlein</type>
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int">10</poids>
</TypeEmploi>
<Salaire rdf:ID="Salaire3">
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int">10</poids>
  <maxDif rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >6000</maxDif>
  <montant rdf:datatype="http://www.w3.org/2001/XMLSchema#int">30000</montant>
</Salaire>
<Age rdf:ID="Age3">
  <maxAge rdf:datatype="http://www.w3.org/2001/XMLSchema#int">3</maxAge>
  <age rdf:datatype="http://www.w3.org/2001/XMLSchema#int">28</age>
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int">5</poids>
</Age>
<VileRésidence rdf:ID="VileRésidence3">
  <villeR xml:lang="fr">constantine</villeR>
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >5</poids>
  <maxDa rdf:datatype="http://www.w3.org/2001/XMLSchema#int">150</maxDa>
</VileRésidence>
<Sexe rdf:ID="Sexe3"> <sexe xml:lang="fr">féminin</sexe>
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int">5</poids>
</Sexe>
<Contrat rdf:ID="Contrat3">
  <poids rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >10</poids>
  <type xml:lang="fr">titulaire</type>
</Contrat>
<Document rdf:ID="Document3">
  <typDocument xml:lang="fr">offred'emploi</typDocument>
  <UriDocument xml:lang="fr">C:\Documents and Settings\zouhe\Bureau\magister</UriDocument>
</Document>
</rdf:RDF>
```

Bibliographie

- [**ALLI08**] : S.Allioua " Annotation sémantique des images". Mémoire de Magister en informatique, université de Tebassa, 2008.
- [**AYAR05**] : S. El ayari. « Moteur D'indexation et De recherche: Vers une recherche Intelligent De L'information Sur Internet ». Mémoire de maîtrise en Sciences du Langage Université PARIS III, Institut de Linguistique et de Phonétique Générales et Appliquées, Juillet 2005, 101 p.
- [**AMAR07**] : F. Amardeilh. « Web Sémantique et Informatique Linguistique : Propositions méthodologiques et réalisation d'une plateforme logicielle », Thèse de doctorat en Informatique, Université Paris X – Nanterre, France 2007.
- [**ABRA04**] : A. Abran, J. W. Moore, P. Bourque, R. Dupuis et L.L. Tripp. «Guide to the Software Engineering Body of Knowledge-SWEBOK". A Project of the IEEE Computer Society Professional Practices Committee, 2004.
- <http://www.swebok.org>
- [**BLAN04**] : E. Blanchard, M. Harzallah. « Inférences en gestion des compétences ». In 15 èmes journées francophones d'ingénierie des connaissances, 2004.
- www.irit.fr/GRACQ/IMG/pdf/IC2004_prog.pdf.
- [**BAUG97**] : J.Baugh. "Rewarding Competencies in Flatter Organizations", Competency: the journal of performance through people, v. 4:3, spring, and 1997.
- [**BERN04**] : A. Bernstein, E. Kaufmann, C. Buerki, M. Klein."Object Similarity In Ontologies a Foundation for Business Intelligence Systems and High-Performance Retrieval". In: Twenty-Fifth International Conferences on Information Systems, 2004.
- [**BIZE04**] : C. Bizer, M. Mochol, D. Westphal. Recruitment XML Clearinghouse Report, Report 8, April 2004.
- www.xml-clearinghouse.de/reports/Clearinghouse-Report.pdf.
- [**BILL02**] : A. Billig, K. Sandkuhl." Match-Making based on Semantic Nets: The XML-based approach of BaSeWeb". In: Tolksdorf, R., Eckstein, R. (Edt.): Proceedings of the 1st workshop on XML-Technologien für das Semantic Web, 2002, pp. 39–51.
- [**BLAN04**] : E. Blanchard, M. Harzallah. « Inférences en gestion des compétences ». In : 15èmes journées francophones d'ingénierie des connaissances, 2004.
- www.irit.fr/GRACQ/IMG/pdf/IC2004_prog.pdf
- [**BEUM07**] : F. Beumier, « Echange d'expérience : La gestion des compétences 2007 ». www.bbest.be/excel/excel79eecompe.pdf
- [**BOUG01**] : I. Boughzala. « Démarche méthodologique de conception de système d'information coopératifs », thèse de doctorat de l' université de pierre et marie curie, décembre 2001.
- [**BOUC08**] : Zouhel Boucetta , Zizette Boufaïda et Leïla Yahiaoui « Appariement sémantique des documents à base d'ontologie pour le e-Recrutement », Colloque Sur L'optimisation et Les Système D'Information, tizi ousou, COSI Algérie 2008.
- [**BIZE05**] : C. Bizer, R. Heese, M. Mochol, R. Oldakowski, R. Tolksdorf, and R. Eckstein, "The Impact of Semantic Web technologies on Job Recruitment Process". In: International Conference workshop on computer science (WI'05), (2005).
- [**BORG96**] : S. Borgo, N. Guarino, & C. Masolo. "Stratified Ontologies: the case of physical objects. Paper presented at the ECAI96". Workshop on Ontological Engineering, Budapest. (1996).

- [BRIN04]: S. Bringay, S. Barry, J. Chartlet. "Les documents et les annotations du dossier patient hospitalier, I3 Informatique Interaction Intelligence, 2004, Vol 4, N 1.
- [BENA05] : A. Benayache. « Construction D'une Memoire Organisationnelle De Formation et Evaluation dans un Contexte E-Learning : Le Projet MEMORAE », Thèse de doctorat, Université De Technologie Compiègne, 2005 ,226 p.
- [TRIC06]: O. Corcho. "Ontology based document annotation: trends and open research Problems". In: International Journal of Metadata, Semantics and Ontologies, 1(1), Inderscience, 2006, pp. 47-57.
- [CIGRE05]: Nomenclature, « les emplois-métiers du système d'information dans les grandes entreprises », (CIGREF), 2005.
<http://www.cigref.fr/cigref/livelink.exe/Nomenclature>
- [DAGM01] : E.Dagmouch. Livre : « Cour en Economie d'Entreprise », septembre 2001.
- [DESM02] : E. Desmontils et C. Jacquin. « Annotation sur le web : notes de lecture ». Journées de l'AS-CNRS Web sémantique, 2002.
- [DENO00] : L. Denoue. «De la création à la capitalisation des annotations dans un espace personnel d'informations ». Thèse de Doctorat d'Informatique, Université de Savoie, France, 2000.
- [DUJA02] : J.M. Dujardin. « Du développement d'un logiciel de gestion des compétences à l'accompagnement de démarches sur le terrain: une expérience de partenariat avec des entreprises et organisations en Région francophone de Belgique ». [www.udi.hec.ulg.ac.be/.../documents/Personnel et gestion du developpement d un logiciel_002.pdf](http://www.udi.hec.ulg.ac.be/.../documents/Personnel_et_gestion_du_developpement_d_un_logiciel_002.pdf)
- [EURO03] : Instructions pour l'utilisation du curriculum vitae Europass
http://www.unispezia.it/doc/modulistica/CVguida_FR.pdf 2003
- [FERN97] : M. Fernandez, A. Gomez-Perez et N. Juristo. "METHONTOLOGY: from ontological art toward ontological engineering". Spring symposium series on ontological engineering. AAAI97, USA, 1996.
- [GRUN96] : M. Grundstein. "La capitalisation des connaissances de l'entreprise, une problématique de management". In actes de 5ème Rencontres du programme MCX, Complexité : la stratégie de la relance, Aix-en Provence, 4-5 juillet 1996.
- [GAND02] : F. Gandon, R. Dieng-Kuntz, O. Corby, A. Giboin. "Web Sémantique et Approche Multi-Agents pour la Gestion d'une Mémoire Organisationnelle Distribuée", IC'2002, Rouen, 28-30 mai.
- [GRUB02]: T. R Gruber. "Ontolingua: A mechanism to support portable ontologies". Technical Report KSL 91-66, Stanford University, Knowledge Systems Laboratory. Revision, 1992.
- [GRUN95]: M. Gruninger, M.S. Fox. "The logic of enterprise modelling". In J. Brown & D.O.Sullivan, Eds. Reengineering the Enterprise: 83-98. London: Chapman & Hall,1995
- [GUAR97] : N. Guarino. "Understanding, building and using ontologies". International J. Human-ComputerStudies, 46, 293-310, 1997.
- [GÓME99]: A Gómez-Pérez. "Ontological Engineering: A state of the art". Expert Update, 2(3), 33-43, 1999.
- [GRUE93] : T. Gruber. "A Translation Approach to Portable Ontology Specifications". Knowledge Acquisition, 5(2), 199-220,1993.
- [GOME98]: A. Gomez-Pérez. "Ontological Engineering", Tutorial of ECAI' 98, Budapest, Hongrie, août 1998.
- [GUER07] : H.E Guergour. « Construction d'une ontologie d'application dans le cadre de l'EAI ».Mémoire de magister université Montouri Constantine, algérie, 2007.

[HEIJ97]: G. Van Heijst, A. Schreiber et B. J. Wielinga . “Using Explicit Ontologies in KBS Development”. *International Journal of Human and Computer Studies /Knowledge Acquisition*, 46(2/3), 183-292, 1997.

[HEMA05]: M. Hemam. " Un processus de développement d'ontologie dans le cadre du Web sémantique". Mémoire de Magister en informatique, centre universitaire Larbi Ben M'hidi-Oum El Bouaghi, institut des sciences exactes, mars 2005.

[HAND01]: S. Handschuh, S. Staab and A.Maedche." CREAM - Creating relational metadata with a component-based, ontology-driven annotation framework". In *Proceedings of the Knowledge Capture Conference (KCAP'01)*, Banff, Canada, 2001, pp. 76-83.

[HAND03]: S. Handschuh, S. Staab et R. Studer. “Leveraging metadata creation for the Semantic Web with CREAM”. *KI '2003 - Advances in Artificial Intelligence*. In *Proceedings Annual German Conference on AI*, Sept. 2003.

[HABE05]: B. Habert. « Instruments et ressources électroniques pour le français », Collection « L'essentiel Français », Ophrys, Paris, 2005, 169 p.

[HAND05]: S. Handschuh. « Creating Ontology-based Metadata by Annotation for the Semantic Web », Thèse de doctorat, Université de Karlsruhe, 2005, 225 p.

[HAND02]: S. Handschuh, S. Staab et F. Ciravegna."S-CREAM – Semiautomatic CREATION of Metadata" .*The 13th International Conference on Knowledge Engineering and Management (EKAW 2002)*, ed Gomez-Perez, A., Springer Verlag, 2002.

[HAAR03]: V. Haarslev et R. Moeller. " Racer : A core inference engine for the Semantic Web". In *2nd International Workshop on Evaluation of Ontology-based Tools (EON-2003)*, Island, 2003.

[HAAR01]: V. Haarslev and R. Moller. “Racer user’s guide and reference manual” version 1.6. Technical report , University of Hamburg, Computer Science Department, 2001.

[IKED99]: M. Ikeda, Y. Hayashi, J. Lai, W. Chen, J. Bourdeau, K. Seta, , and R. Mizoguchi. “An ontology, more than a shared vocabulary”. In *Proceedings of the AIED '99 Workshop on Ontologies for Educational Systems*, Le Mans, Francia, 1999.

[JOBG07] : **JOBGATE** Source de Talents.
http://www.jobgate.fr/pdf/plaquette_Mai2007.pdf.

[KALY03]: A. Kalyanpur, J. Hendler, B. Parsia and J. Golbeck. “SMORE – Semantic Markup, Ontology, and RDF Editor”, 2003,
<http://www.mindswap.org/papers/SMORE.pdf>.

[KAHA01]: J. Kahan, M-J Koivunen., E. Prud'Hommeaux et R. Swick. “Annotea: An Open RDF Infrastructure for Shared Web Annotations”. In *Proceedings 10 th International World Wide Web Conference (WWW 2001)* Hong Kong.

[KOIV2005]: M-R. Koivunen. “Annotea and Semantic Web Supported Collaboration, Invited talk at Workshop on User Aspects of the Semantic Web” (UserSWeb). At *European Semantic Web Conference (ESWC 2005)* Heraklion, Greece, 29 May 2005.

[KOIV01]: M. R. Koivunen, , and R. Swick. “Metadata Based Annotation Infrastructure offers Flexibility and Extensibility for Collaborative Applications and Beyond”. *Proceedings of K-CAP 2001, Workshop on Knowledge Markup and Semantic Annotation*. Victoria, Canada, 21 October, 2001.

[KNUB04]: H. Knublauch, M. A. Musen et A. L. Rector . “Editing Description Logic Ontologies with the Protégé OWL Plugin”. *International Workshop on Description Logics - DL2004*, A technical discussion for logicians, Canada, 2004.

[LOCH99]: Loche Terminologie et intelligence artificielle (actes du colloque de Nantes, 10-11 mai 1999).

- [LUON07] : P. H Luong. « Gestion de L'évolution D'un Web Sémantique D'entreprise », Thèse de doctorat, l'Ecole des Mines de Paris, France, 2007.
- [LMD04] : Reforme LMD de l'enseignement Supérieur, Université de Constantine, Faculté des Sciences de l'ingénieur, Département Informatique, 2004.
- [MEST07] : A Mestiri. « Vers Une Approche Web Sémantique dans Les Applications de Gestion de Conférences », Mémoire de maîtrise en informatique pour l'obtention du grade de maîtres sciences (M.Sc.), Université Laval, Québec ,2007.
- [MORI02] : E. Morin, E. Desmontils, C. Jacquin. " Indexation sémantique de documents sur le Web: application aux ressources humaines". Journées de l'AS-CNRS Web sémantique, IRIN, Octobre 2002.
- [MAE02]: A. Maedche, V. Zacharias. " Clustering Ontology-based Metadata in the Semantic Web: Principles of Data Mining and Knowledge Discovery". In: 6th European Conference, PKDD, Helsinki, Finland , 2002.
- [MARS98]: C.Catherine. "Toward an ecology of hypertext annotation". Actes de ACM conference on Hypertext and Hypermedia: Links, Objects, Time and Space, Juin 1998, Pittsburgh (USA). ACM Press, pp. 40-49.
- [MKAD04] : A. Mkardmi. « Recherche collaborative d'informations : repenser l'architecture des SRIs à l'ère numérique », Thèse de doctorat, Université paris 8, 2004, 293 p.
- [MIFE06] : A. MIFE. Les sites internet pour l'emploi 2006.
<http://bijbourg.free.fr/site%20emploi.pdf>
- [NOY00]: N. Noy, R. W. Ferguson and M. A. Musen. "The knowledge model of Protégé2000: combining interoperability and flexibility". In Proceedings of the International Conference on Knowledge Engineering and Knowledge Management (EKAW'00), 2000.
- [NICO03]: S. Nicolas. "Sesei : une filtre sémantique pour les moteurs de recherche conventionnels par comparaison de structures de connaissance extraites depuis des textes en langage naturel. Master's thesis, Département d'informatique et de génie logiciel, Université Laval. (2003).
- [ONET]: Occupationnel information network O*NET.
<http://online.onetcenter.org/>.
- [PRIE 04] : Y.Prie and S.Garlatti. « Méta-données et annotations dans le Web sémantique », Hors série de la Revue Information -Interaction - Intelligence (I3), 4(1), Cépaduès, Toulouse, 2004, pp. 45-68.
- [PSYC04] : V. Psyché. « Proposition d'une méthode d'ingénierie ontologique pour les EIAH : application aux systèmes auteurs Aspects théoriques, modèles et expérimentations. Une proposition de recherche doctorale MAI 2004.
<http://www.ecogesam.ac-aix-marseille.fr/revue/N2tmp/KM/>
- [PATE03]: P. F. Patel-Schneider, P. Hayes et I. Horrocks. «OWL web ontology language semantics and abstract syntax". W3C Working Draft, 2003.
<http://www.w3.org/TR/2003/WD-owl-semantics-20030331/>.
- [PATL03] : P. F. Patel-Schneider, I. Horrocks et F. V. Harmelen. «From SHIQ and RDF to OWL: The making of a web ontology language". 2003.
<http://www.cs.man.ac.uk/%7Ehorrocks/Publications/download/2003/HoPH03a.pdf>
- [RECT,04] : A. Rector, M. Horridge, H. Knublauch, R. Stevens et C. Wroe." A Practical Guide to Building OWL Ontologies Using the Protégé-OWL Plugin and CO-ODE Tools". Edition 1.0, université de Manchester en coopération avec l'université de Stanford, aout 2004.
<http://www.co-ode.org/resources/tutorials/ProtegeOWLTutorial.pdf>
- [SIMO00]: C. Simone. "Unifying or reconciling when constructing Organizational Memory» In extended version of ECAI' 2000 workshop, Università di Torino, 2000.

- [**SIDH05**] : S. Sidhom, C. Robert, A. David. « Analyse automatique de textes comme point de départ d'un processus d'annotation », revue e-TI 2005.
- [**SCHM06**]: A. Schmidt, C. Kunzmann. "Professional Learning Ontology and Competencies". In: the OntoContent workshop at the On the Move Federated Conferences, Montpellier (2006).
- [**SCHM06a**]: A.Schmidt,C.Kunzmann. "Professional Learning Ontology and Competencies". The project Learning in Process and subsequent research activities in the Information Process Engineering research group of FZI Research Center for Information Technologies (2006)
http://professional-learning.eu/competence_ontology
- [**SOWA00**]: J. F. Sowa. "Knowledge Representation: Logical, Philosophical, and Computational Foundations". Brooks/Cole Publishing Co, London, 2000 edition, 2000.
- [**SHIM04**] : Y Shimimana, Outils d'Ingénierie Ontologique: étendre les ontologies aux multiples points de vue pour résoudre les conflits 2004.
- [**STAA01**] – S. Staab, A. Maedche, S. Handschuh. An Annotation Framework for the Semantic Web,in Proceedings of the 1st International Workshop on MultiMedia Annotation, Tokyo, Japon, 2001.
- [**STAA01b**] – S. Staab, A. Maedche, S. Handschuh. Creating Metadata for the Semantic Web: An Annotation Framework and the Human Factor, Technical Report, Institut AIFB, Université de Karlsruhe, Allemagne, 2001, 25 p.
- [**TRIC02**]: F. Trichet. "Annotating WithCigref". Un prototype d'annotation développé sous protégé-2000, 2002.
- [**TRIC06**]: F.Trichet, V. Radevski. "Competency-based Systems dedicated to e-Recruitment". Proceedings of the 4th International Conference on Computer Scienceand Information Technology (CSIT'2006, Amman).Volume 3, pp 185-197. ISBN: 9957-8592-0-X, 2006.<http://csit2006.asu.edu.jo>
- [**TIXI01**]: B Tixiter. « La problématique de la gestion des connaissances », Le cas d'une entreprise de développement informatique bancaire, Rapport de Recherche No 01.9 Septembre 2001.
- [**TISS99**] : R. C. Tisseyre. "Knowledge Management, théorie et pratique de la gestion des connaissances", Eds.Hermès,1999.
<http://www.cnam.fr/lipsor/dso/articles/fiche/rctisseyrekmtheorpratique.doc>
- [**TENI07**] : S. Tenier, A. Napoli, X. Polanco, Y. « Toussaint, A sémantique de pages web ».
- [**TECH00**] : NOCompétences, « Dictionnaire sectoriel de compétences : Exploitants de réseaux – Télécommunications», 2000.
www.technocompetences.qc.ca
- [**TOUS07**] : Y. Toussaint, S. Tenier. « Annotation sémantique par classification », 2007.
- [**USCH95**]: M. Uschold et M. King . "Towards a Methodology for Building Ontologies" .Paper presented at the Workshop on Basic Ontological Issues in Knowledge Sharing, 1995.
- [**USCH96**] : M. Uschold et M. Grüninger. Ontologies: Principles, Methos and Applications. Journals of KnowledgeEngineering Review, 11(2), 1996.
- [**WEIB97**]: S. Weibel et C. Lagoze. "An element set to support resource discovery" .In: International Journal on Digital Libraries, 1997, 1: 176-186, 1997.
- [**W3C04**]: W3C (2004) "OWL Web Ontology Language Overview:
<http://www.w3.org/TR/2004/REC-owl-features-20040210/>".
- [**YAH06**] : L. Yahiaoui, Z. Boufaïda et Y. Prié. «Automatisation du recrutement dans le cadre du web sémantique».IC 2006, 17 journée francophones d'ingénierie des connaissances, Nantes du 28.
- [**YAH06a**] : L. Yahiaoui, " Annotation sémantique de documents application au

e-recrutement". Mémoire de Magister en informatique, université montouri Constantine, juin 2006.

[ZHON02]: J. Zhong, H. Zhu, J. Li, and Y. Yu. « Conceptual Graph Matching for Semantic Search ». In: Proceedings of the 2002 International Conference on Computational Science, 2002.