

A Semantic Web Services for Medical Analysis using the OWL-S Language

Zaidi Fayçal

Mathematics and Computer Science Department
Tebessa University –Tebessa - ALGERIA

Touahria Mohamed

Computer Science Department
Ferhat Abbas University – Setif - ALGERIA

ABSTRACT

The merger of the Semantic Web with the Web Service giving rise to Semantic Web Service, this new technology allows software agents to discover, select, call, compose, invoke, and execute automatically a Web services without the intervention of human beings. In this article, we will apply this new technology in the medical field, specifically in medical analysis, by the construction of two ontologies: a domain ontology "Medical analysis domain ontology", following the method METHONTOLOGY and using OWL language and PROÉGÉ tool, and an ontology of web services medical "Medical analysis Web services ontology", using OWL-S language, OWL-S EDITOR for PROTÉGÉ tool and the Integrated Development Environment "NetBeans IDE". The ontology of Web services uses the concepts of domain ontology to define and describe the IOPEs (inputs/outputs/Preconditions/Effects) of Web service. These two ontologies are used by a health care system; it includes all those involved in the process of care, human actors (doctors, patients, lab technicians...), software actors (medical web services, medical applications...).

Keywords

Web service, semantic Web service, ontology, OWL-S, WSMO, medical analysis.

1. INTRODUCTION

The application of Semantic Web [1] to the Web services [2] has given rise to semantic Web services [3]. This new technology allows software agents to discover, select, call, compose, invoke, and execute automatically a Web services without the intervention of human beings [4], it combines the technology of Web service (WSDL, SOAP, UDDI) with Semantic Web technologies (RDF, RDFS, OWL) as shown in Figure 1.

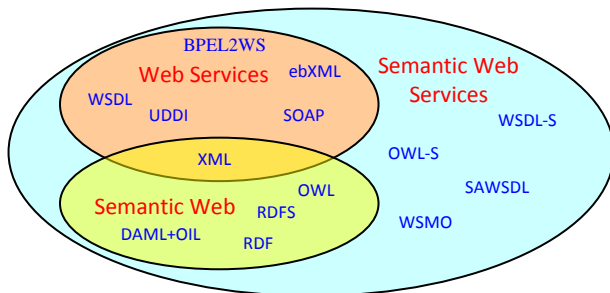


Fig 1: semantic Web service technology

On account of the increase of knowledge and web services in the medical analysis field, it seems necessary to resort to semantic web service technologies, by the construction of medical resources ontological and semantic description of web services of medical analysis in order to increase the quality of care [5]. The aim of this paper is to answer the following questions:

What are the approaches of semantic description of the Web services?

And which is among these approaches the most adequate, and which covers adequately the semantics of the Web services?

How to apply this new technology of Semantic Web service on the medical field, specifically in the field of medical analysis?

To this aim, this work is organized as follows: Section 2 provides an overview of the languages of semantic description of Web services OWL-S and WSMO. Section 3 presents a conceptual comparison between these two languages. Section 4 discusses the construction of two ontologies for semantic Web services in the medical analysis field. Section 5 presents the related work in this area. Finally, Section 6 is devoted to the conclusion and the future work.

2. LANGUAGES OF SEMANTIC WEB SERVICES

Several languages have been defined for describing semantic web services, these languages are based on techniques of ontological engineering [6]. Among these languages, two languages are the richest, standardized and generic: OWL-S (Web Ontology Language for Web Services) And WSMO (Web Service Modeling Ontology).

2.1 Web ontology language for Web services

The Web Ontology Language for Web Services (OWL-S) [7] is a subset of language "Web Ontology Language" (OWL) [8], dedicated to the semantic description of Web services. It is compatible with syntactic description formats such as WSDL.

The OWL-S ontology contains four sub-ontologies:

ServiceProfile (What a service does).

ServiceProcess or ServiceModel (How the service works).

ServiceGrounding (How the service is implemented).

A fourth ontology (Service) contains the Service concept which links together a ServiceProfile, a ServiceModel and a ServiceGrounding concept.

The following figure illustrates the ontologies OWL-S: a resource provides a service; this service presents a ServiceProfile; is described by Service Model and supports a ServiceGrounding.

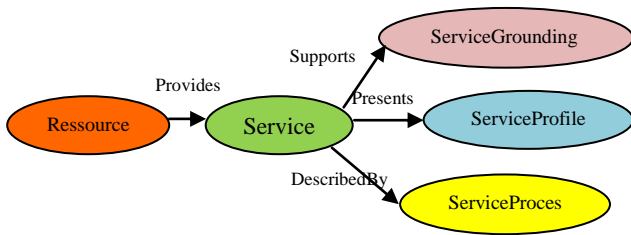


Fig 2: Ontology OWL-S

The service profile: Describes the profile of Web service (service name, service provider, description text,...), and a description of service functionality (inputs, outputs, preconditions and effects). This profile ontology is useful for the discovery and selection of Web service.

The ServiceModel or ServiceProcess: Represents the behavior of a service. Web services can be modeled in this ontology as a process (atomic, simple or composite process). This ProcessModel ontology is useful for service execution.

The ServiceGrounding: Used to determine how to access the service published (the data exchanged, communication protocols). The Grounding ontology is useful for service invocation, it specifies the communication protocols used and the specific service elements such as ports to use, it makes the link between the description of OWL-S and WSDL specification. (Figure 3)

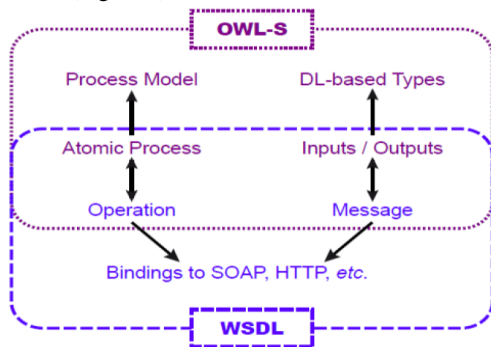


Fig 3: Relationship between OWL-S and WSDL

2.2 Web service modeling ontology

Web Service Modelling Ontology (WSMO) [9], is a conceptual model for Semantic Web services. It comprises of an ontology of core elements for Semantic Web services, based on the Web Service Modelling Framework (WSMF) and has an execution environment (WSMX: Web Service Execution Environment).

The WSMO is organized around four fundamentals components: Web services, Goals, Ontologies and Mediators, as shown in the figure below:

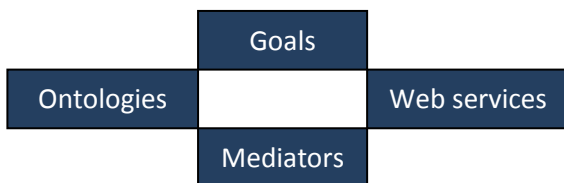


Fig 4: The four elements of the WSMO approach

-Ontologies: define the terminology used by other WSMO elements, in terms of concepts, relations, functions, instances and axioms.

-Goals: describe aspects related to user desires with respect to the requested functionality.

-Web Services: defines the functionalities offered by the service.

-Mediators: it is a link between different WSMO elements to enable interoperability between heterogeneous components.

WSML language (Web Service Modeling Language) is used to describe formally all the elements of WSMO.

3. COMPARISON AND DISCUSSION

The two approaches of semantic description of the Web services (OWL-S and WSMO) follow the same objective: describe and manage the semantic associated to the Web services, because this semantics remains absent in the Web service layers.

Nine criteria have been chosen to synthesize the characteristics of these approaches [10]:

-Level of abstraction: that allows specifying the level of abstraction (conceptual, technical) associated with the approach considered.

-Standardization: represents the degree of standardization enabled by the approach.

-Semantic description: describes the manner by which services are semantically described by the approach studied.

-Mediation: the interoperability level between heterogeneous components.

-Maturity: the maturity of the approach in relation to the problem of semantic description of services, and the percentage of use of the approach in practice.

In addition, four criteria related to: the service discovery; interaction with the service; languages and logics used.

The main characteristics of the two approaches presented previously, are summarized in Table 1.

Table 1. Comparison between WSMO and OWL-S

N° :	Analysis Criteria	OWL-S	WSMO
01	Level of abstraction	Conceptual (based on Profile, and Process)	Conceptual (based on Goals, Web services and ontologies)
02	Standardization (W3C)	very strong (based on RDF ¹ , RDFS ² and OWL)	low (based on WSML)
03	Semantic description	OWL-S Ontology	WSMO Ontology
04	Mediation	Under development	Strong (MEDIATORS elements)
05	maturity of the approach	Strong	Medium
06	Web service discovery (detection of appropriate)	ServiceProfile	Goals +Web Services (Web service capability)

¹ www.w3.org/RDF/

² www.w3.org/TR/rdf-schema/

	service)		
07	Interaction with the Web service	ServiceProcess	Interface Classe (Choreography)
08	Languages used	OWL	WSML
09	Logic used	Description logic	-Description logic -Predicate logic -Logic programming

Considering all the evidence presented in this section, we think that the OWL-S is the approach with the most mature, generics and standardization. This is the fundamental reason which led us to base our works on this approach (OWL-S).

4. DESIGN AND IMPLEMENTATION

In this paper, we shall construct two ontologies, an ontology in the field of medical analysis, through the Protégé tool³ that uses the OWL language, this ontology called "analysis medical domain ontology". And another ontology of web services of medical analysis called "medical analysis web service ontology" which allows semantic annotation of Web services for medical analysis by using the OWL-S language. These two ontologies are used by a health network (Figure 5), the healthcare network includes all those involved in the process of healthcare, human actors (doctors, patients, laboratory technicians ...) and software actors (medical web services, medical applications ...).

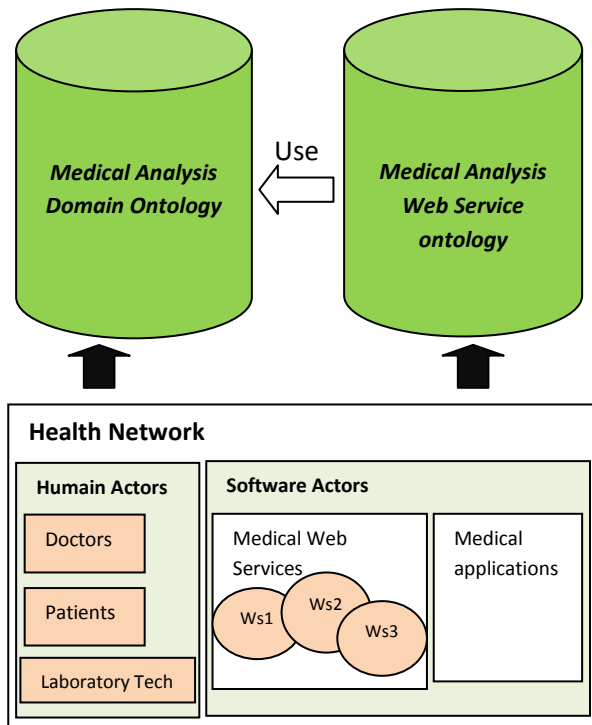


Fig 5: Network healthcare and medical ontologies

4.1 Medical Analysis Domain ontology

We shall construct a domain ontology [11] in the field of medical analysis, this ontology contains concepts related to medical analysis (serum sodium, creatinine, medical score, weight, age, size, medicine ...) and other concepts related to the Web services such as inputs, outputs, preconditions, effects ... etc. To accomplish this task, we use OWL language and Protégé tool, and following a process of building a domain ontology knowledge from raw ontology arriving in an operational ontology, this process is inspired by the method METHONTOLOGY [12], this method involves five phases (Requirements Specification, Conceptualization, Formalization, Implementation and Verification).

Requirements Specification: in this phase, we construct an informal document for requirement specification written in natural language, which allows describing: The sources of information collected, the goal of this ontology and the users of this ontology.

-Conceptualization: Many tasks performed in this phase such as: Construction of the glossary of terms, the construction of the dictionary of concepts, describe binary relations between concepts, specify the attributes, axioms and instances of concepts.

-Formalization: in this phase we formalize the results concepts of conceptualization phase in a language understandable by machine; our choice is focused on the description logic based on Shoin (D) syntax.

-Implementation: in this phase, we implement the ontology by software tool, to enable machine via this ontology to manipulate domain knowledge. The Protégé tool has been chosen, this choice is justified because Protégé is an easy and simple editor to build ontologies. Protégé is also a Java library that can be extended to create present applications of knowledge bases.

Verification and Evaluation: In this phase we use the Pellet inference engine to test and evaluate our ontology.

Our ontology contains 300 concepts, 14 relationships, 20 attributes and a few instances and axioms in the field of medical analysis as shown in Figure 6, with the possibility of extending it to all time by adding new concepts. This ontology is very useful for users and health experts (doctors, students, laboratory technicians, patients ...) to create, view and maintain their knowledge of this area and have a common and standardized medical language.

Figure 6 shows part of the taxonomy of the source code ontology.

³ <http://protege.stanford.edu/>



Fig 6: Medical Analysis Domain ontology

4.2 Medical Analysis Web Services Ontology

We shall construct an ontology of web services of medical analysis called "Medical Analysis Web Service Ontology", which allows semantic annotation of Web services related

medical analysis. We chose the language "OWL-S", the "OWL-S Editor for Protégé⁴" tool and the integrated development environment "NetBeans IDE⁵" to implement the web services. The ontology of Web services uses the concepts of domain ontology to describe and define the concepts specific to Web services such as IOPEs (inputs / outputs / Preconditions / Effects), the Client, the Server, the Service Provider, Service category ... etc.

We following a process of building Web services ontology contains four phases (Requirements Specification, Conceptualization, Formalization, Verification and execution).

Requirements Specification: in this phase, we specify in informal language the sources of services collected, the goal of this ontology and the users of this ontology.

Conceptualization: Many tasks performed in this step such as:

- Collection of web services.
- Define a text description for each web service.
- Define the synonyms, the category, the provider (Contact Information) for each web service.
- Identify the Inputs, outputs, preconditions and effects of Web service.
- Identify the type of web service processes (atomic, simple, or composite).
- If the web service is composite, what are the sub web services with their scheduling (Sequence)?

-Implementation: In this phase, we implement the Web service by the Integrated Development Environment "NetBeans IDE", and we will create the instances OWL-S (ServiceProfile, ServiceProcess, ServiceGrounding) of each Web service by using the "OWL-S EDITOR for PROTÉGÉ" tool, and make a mapping between ServiceGrounding and the WSDL file of Web service generated by NetBeans IDE.

Verification and execution: in this phase we verify the consistency of the ontology to ensure the good functioning of all web services. And finally run the web services.

Our ontology contains more than 74 web services in the field of medical tests (Figure 7), and is very useful for software agents to discover, select, call, compose, invoke, and execute automatically Web services of medical analysis.

4.3 Example scenario: Assessing renal function (évaluer fonction rénale)

Renal function, in nephrology, is an indication of the state of the kidney and its role in renal physiology. Glomerular filtration rate describes the flow rate of filtered fluid through the kidney. Creatinine clearance rate (CrCl) is the volume of blood plasma that is cleared of creatinine per unit time and is a useful measure for approximating the Glomerular filtration rate. Creatinine clearance exceeds Glomerular filtration rate due to creatinine secretion, which can be blocked by cimetidine.

In practice, it is recommended to estimate Creatinine clearance from serum creatinine, age, sex and weight of the patient (Figure 8) using the Cockcroft and Gault formula.

⁴ <http://owlseditor.semwebcentral.org/>

⁵ www.netbeans.org/

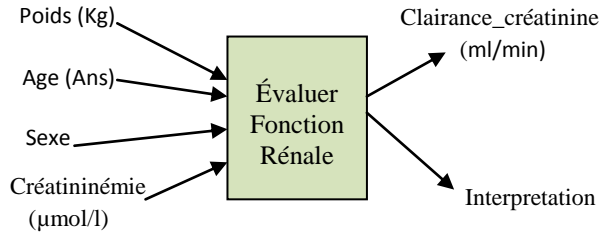


Fig 8: Inputs and Outputs of service

Creatinine clearance by Cockcroft-Gault formula :
For males = $[1.23 \times \text{Poids (kg)} \times (140 - \text{age})]$ divided by Créatininémie ($\mu\text{mol/l}$) (ml/min).
For females = $[1.04 \times \text{Poids (kg)} \times (140 - \text{age})]$ divided by Créatininémie ($\mu\text{mol/l}$) (ml/min).

Terms of use: Créatininémie between 6 and 70.

Table 2. Stages of Chronic Kidney Disease

Stage	Definition	Creatinine Clearance (ml/min)
1	Slight kidney damage with normal or increased filtration	More than 90
2	Mild decrease in kidney function	60-89
3	Moderate decrease in kidney function	30-59
4	Severe decrease in kidney function	15-29
5	Kidney failure	Less than 15 (or dialysis)

4.3.1 Presentation of a clinical case

Ms. AMINA 50 years. It measures 1.61 m and weighs 46 kg and has an HIV infection.

The serum creatinine is normal (11 mg/l or 96 $\mu\text{mol/l}$).

His antiretroviral treatment includes:

Viread 300 mg/day,

Rétrovir 200 mg 3 times per day,

Epivir 150 mg 2 times per day,

4.3.2 Analysis of clinical case

The Creatinine seems "normal" but Ms. AMINA has kidney failure. His creatinine clearance calculated using the Cockcroft & Gault formula is:

$CC \text{ or } CLCR = \{1.04 \times 46 \times (140 - 50) / 96\} = 44.9 \text{ ml/min}$

AMINA lost more than half of their kidney function. She has a kidney "moderate" (stage 3). What to do?:

1 - Nephrology Consultation to diagnose (the) cause (s) of the degradation of renal function + prescription a treatment to correct renal failure.

2-Check that the dosages of drugs prescribed to Ms. AMINA is adapted to their level of renal function (45 ml / min).

4.3.3 Implement the Web service (generate the WSDL file and the URL)

Before adding semantic to Web service, we will implement the Web service by a programming language such as java.

The following figure displays a part of the Web service: Evaluer Fonction Rénale (calculate the CC) encoded in the java syntax using the integrated development environment "NetBeans IDE 6.8".

```

package server;

import javax.jws.WebMethod;
import javax.jws.WebParam;
import javax.jws.WebService;

/**
 *
 * @author ZAIDI Fayçal
 */
@WebService()
public class EvaluerFonctionRenale {

    /**
     * Web service operation
     */
    @WebMethod(operationName = "EvaluerFonctionRenaleOp")
    public Double EvaluerFonctionRenaleOp (@WebParam(name = "Creatininemie")
    int Creatininemie, @WebParam(name = "Poids")
    int Poids, @WebParam(name = "Age")
    int Age, @WebParam(name = "Sexe")
    String Sexe) {
        //TODO write your implementation code here:
        if (Sexe.equals("homme")){
            double CC = 1.23 * Poids * (140 - Age)/Creatininemie;
            return CC;
        }
        else { // Sexe = femme
            double CC = 1.03 * Poids * (140 - Age)/Creatininemie;
            return CC;
        }
    }
}
  
```

Fig 9: The web service in NetBeans IDE 6.8 environment

The IOPEs in this example are:

The Inputs are: poids (Weight, 46 kg), âge (age, 50 ans), créatinine (serum creatinine, 96 $\mu\text{mol/l}$), sexe (sex, female).

The Outputs are: Creatinine Clearance (44,9 ml/min), Interpretation (Moderate decrease in kidney function).

The precondition *Valid_Creatininemie* is : serum creatinine between 6 and 70 mg/L.

The effect *Medical_Decision* is :

- Nephrology Consultation to diagnose (the) cause (s) of the degradation of renal function + establishment of a specific treatment to correct renal failure.
- Check that the dosages of drugs prescribed to Ms. AMINA is adapted to their level of renal function (45 ml / min).

Then we will generate the WSDL and the URL of the service for use with OWL-S.

4.3.4 Semantic description of the Web service

For the semantic description of web service with the OWL-S, we must construct:

- An instance for the service class.
- An instance for the service *Profile* class.
- An instance for the service *Process* class.
- An instance for the service *Grounding* class.
- Mapping between OWL-S Parameter and the implementation of WSDL file generated by “Net Beans IDE 6.8”.

As shown in the figure below:

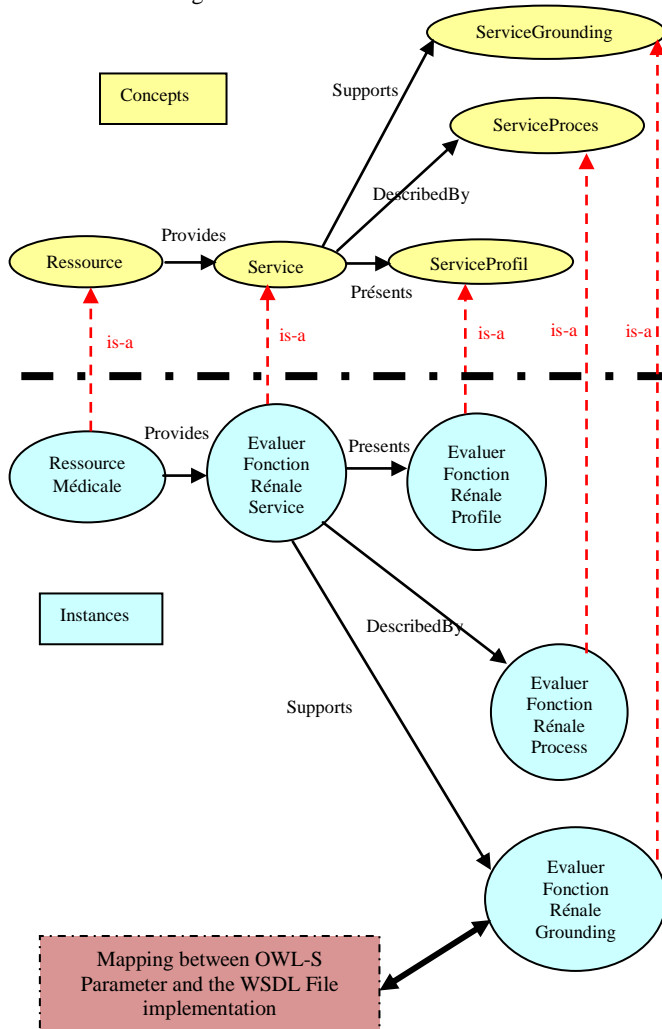


Fig 10: Concepts and instances of Web service

The semantic description of this service by using the tool OWL-S Editor for Protégé is shown in the following figure:

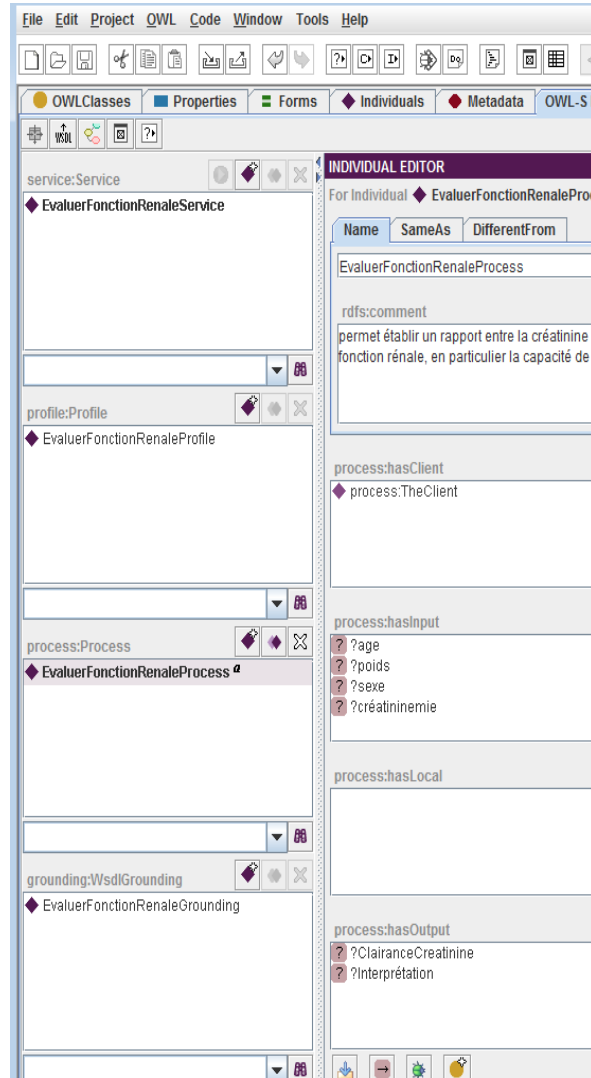


Fig 11: Semantic description of service “EvaluerFonctionRenale” under OWL-S Editor

The OWL code for this service (service instance, Profile, Process, Grounding) using OWL-S are described in Listing 1, 2, 3 and 4.

Instance service

The following listing displays a part of the ontology (Instance Service) encoded in the OWL RDF syntax:

```

1 <service:Service rdf:ID="EvaluerFonctionRénale">
2 <service:presents
  rdf:resource="&EFR_profile;#EFRénale_Profile"/>
3 <service:describedBy
  rdf:resource="&EFR_process;#EFRénale_Process"/>
4 <service:supports
  rdf:resource="&EFR_grounding;#EFRénale_Grounding"/>
</service:Service>

```

Listing 1: Instance service EvaluerFonctionRénale

In Listing 1.Lines \$:

- \$1 name of service is: EvaluerFonctionRénale .
- \$2 reference to service profile : #EFRénale_Profile.
- \$3 reference to service Model (Process) : #EFRénale_Process.
- \$4 reference to service Grounding : #EFRénale_Grounding.

Service profile

The following listing displays a part of the ontology (Instance ServiceProfile) encoded in the OWL RDF syntax:

```
1<profile:Profile rdf:ID="EFRénale_PROFILE">
2<service:presentedBy
rdf:resource="&EFR_service;#EvaluerFonctionRénale"/>
3<profile:has_process
rdf:resource="&EFR_process;#EFRénale_Process"/>
4<profile:serviceName>EvaluerFonctionRénale
</profile:serviceName>
5<profile:textDescription xml:lang="fr">
Évaluer la Clairance de la créatinine (mL/min) en fonction de la Créatininémie
( $\mu\text{mol/L}$ ), du Poids (P en kg), de l'Âge (A en ans) et du Sexe. La Créatinine est un
produit de dégradation de la Créatine produite par le foie et stockée dans le
muscle, filtrée par le rein. La Clairance de la créatinine est actuellement un des
meilleurs marqueurs de la fonction rénale.
</profile:textDescription>
6<profile:contactInformation>
Name_actor, phone, Fax, Email, physicalAddress
</profile:contactInformation>
7<profile:qualityRating>SomeRating</profile:qualityRating>
8<profile:serviceCategory>
<addParam:UNSPSC rdf:ID="UNSPSC-category">
</addParam:UNSPSC>
</profile:serviceCategory>
9<profile:hasInput rdf:resource="&EFR_process;#Poids"/>
10<profile:hasInput rdf:resource="&EFR_process;#Age"/>
11<profile:hasInput rdf:resource="&EFR_process;#Sexe"/>
12<profile:hasInput
rdf:resource="&EFR_process;#Créatininémie"/>
13<profile:hasOutput
rdf:resource="&EFR_process;#Clairance_Créatinine"/>
14<profile:hasOutput
rdf:resource="&EFR_process;#Interpretation"/>
15<profile:hasPrecondition
rdf:resource="&EFR_process;#Valide_Créatininémie"/>
16<profile:hasEffect
rdf:resource="&EFR_process;#Décision_Médicale"/>
</profile:Profile>
```

Listing 2: Service profile EFRénale_PROFILE

In Listing 2.
lines \$:

- \$1 name of class Profile is *EFRénale_PROFILE*.
- \$2 the Profile class is described by the service # EvaluerFonctionRénale which is a reference to a part of OWL-S file.
- \$3 the Profile classe is a process model : #EFRénale_Process.
- \$4 the name of service is *EvaluerFonctionRénale*.
- \$5 the Profile class is accompanied by a text description of the service.
- \$6 the Profile class is accompanied by information on the service provider (name, phone, fax, email, physical address ...).
- \$7 a percentage of the vote on the web service that allows the estimated quality of the service by the service consumer.
- \$8 service category according to UNSPSC (United Nations Standard Products and Services Code) is a coding system to classify both products and services for use in electronic commerce systems.
- \$9 the service has an input parameter #Poids is a reference to the Process Model.
- \$13 the service has an output parameter #Clairance_Créatinine which is a reference to the Process Model.

- \$15 the service has a precondition. #Valide_Créatininémie which is a reference to the Process Model.
- \$16 the service has an effect. #Décision_Médicale which is a reference to the Process Model.

Service Model (Process)

The following listing displays a part of the ontology (Instance ServiceModel) encoded in the OWL RDF syntax:

```
1<process:ProcessModel rdf:ID="EFRénale_Process">
2<service:describes
rdf:resource="&EFR_service;#EvaluerFonctionRénale"/>
</process:ProcessModel>
3<process:AtomicProcess rdf:ID=" EvaluerFonctionRénale ">
<process:hasInput rdf:resource="&Poids"/>
<process:hasInput rdf:resource="&Age"/>
<process:hasInput rdf:resource="&Sexe"/>
<process:hasInput rdf:resource="&Créatininémie"/>
<process:hasOutput rdf:resource="&Clairance_Créatinine "/>
<process:hasOutput rdf:resource="&Interpretation"/>
<process:hasPrecondition
rdf:resource="&Valide_Créatininémie"/>
<process:hasEffect rdf:resource="&Décision_Médicale"/>
</process:AtomicProcess>
4<process:Input rdf:ID="Poids">
<process:parameterType
rdf:resource="&OntoMédicale.owl;#Poids"/>
</process:Input>
5<process:Input rdf:ID="Age">
<process:parameterType rdf:resource="&
OntoMédicale.owl;#Age"/>
</process:Input>
6<process:Input rdf:ID="Sexe">
<process:parameterType rdf:resource="&
OntoMédicale.owl;#Sexe"/>
</process:Input>
7<process:Input rdf:ID="Créatininémie">
<process:parameterType rdf:resource="&
OntoMédicale.owl;#Créatininémie"/>
</process:Input>
8<process:UnConditionalOutput rdf:ID="
Clairance_Créatinine ">
<process:parameterType rdf:resource="&
OntoMédicale;#Clairance_Créatinine"/>
</process:UnConditionalOutput>
9<process:UnConditionalOutput rdf:ID="Interpretation ">
<process:parameterType rdf:resource="&
OntoMédicale.owl;#Interpretation"/>
</process:UnConditionalOutput>
10<process:UnConditionalPrecondition
rdf:ID="Valide_Créatininémie">
<process:cePrecondition rdf:resource="&
OntoMédicale.owl;#Valide_Créatininémie"/>
</process:UnConditionalEffect>
11<process:UnConditionalEffect
rdf:ID="Décision_Médicale">
<process:ceEffect rdf:resource="&
OntoMédicale.owl;#Décision_Médicale"/>
</process:UnConditionalEffect>
```

Listing 3: Service model EFRénale_Process

In Listing 3.

lines \$:

- \$1 name of class ProcessModel is EFRénale_Process.
- \$2 the service is described by # EvaluerFonctionRénale.
- \$3 the service EvaluerFonctionRénale has an atomic process with a set of IOPEs (inputs, outputs, preconditions, effects) which are a references.
- \$4 the input parameter #poids is described in OntoMédicale.owl ontology.
- \$9 the output parameter #Interpretation is described in OntoMédicale.owl ontology.
- \$10 the precondition parameter #Valide_Créatininémie is described in OntoMédicale.owl ontology.
- \$11the effect parameter #Décision_Médicale is described in OntoMédicale.owl ontology.

Service Grounding

The following listing displays a part of the ontology (Instance ServiceGrounding) encoded in the OWL RDF syntax:

```
1<grounding:wSDLGrounding rdf:ID="
EFRénaLe_Grounding">
2<service:supportedBy rdf:resource="#EFR_service;#
EvalueurFonctionRénaLe"/>
</grounding:wSDLGrounding>
3<grounding:owlsProcess
rdf:resource="#EFR_process;#EFRénaLe_Process"/>
4<grounding:wSDLOperation
rdf:resource="#EvalueurFonctionRénaLe_operation"/>
5<grounding:wSDLInputs rdf:parseType="Collection">
---
</grounding:wSDLInputs>
6<grounding:wSDLOutputs rdf:parseType="Collection">
---
</grounding:wSDLOutputs>
7<grounding:portType>
<xsd:anyURI
rdf:value="#EvalueurFonctionRénaLeGroundingWSDL;#Evalue
urFonctionRénaLe_PortType"/>
</grounding:portType>
8<grounding:operation>
<xsd:anyURIrdf:value="#EvalueurFonctionRénaLeGroundingW
SDL; #EvalueurFonctionRénaLe_operation">
</grounding:operation>
```

Listing 4: Service grounding EFRénaLe_Grounding

In Listing 4.

Ligne\$:

\$1 a class Grounding is declared whose name is EFRénaLe_Grounding.

\$2 the service is supported by #EvalueurFonctionRénaLe.

\$3 Grounding for the atomic process #EFRénaLe_Process.

\$4 Reference to the WSDL file operation.

\$7 locates the portType used by the Web service.

\$8 locates the operation used by the Web service.

5. RELATED WORK

Two complementary projects that are investigating the use of Semantic Web Services in the healthcare sector: COCOON Glue⁶ and ARTEMIS⁷. COCOON Glue is a WSMO compliant discovery engine that aims at developing an efficient system for the management of semantically described Web Services and their discovery (teleconsultation meeting services, medical advice services or teaching services...). ARTEMIS is a semantic Web service based Peer to Peer Infrastructure for the interoperability of medical information systems. ARTEMIS has a peer-to-peer architecture in order to facilitate the discovery of healthcare organisations, Web Services and patient records.

6. CONCLUSION AND FUTURE WORK

Today, the Semantic Web Services are promising way to better exploit the Web services by automating the various tasks related to the life cycle of a service. It seems that they are essential to enable effective use of Semantic Web Services in different areas such as the medical field.

We have developed two ontologies, "medical analysis domain ontology" and "medical analysis web service ontology". The first ontology contains 300 concepts, and it is very useful for users and domain experts (doctors, students, laboratory technicians, patients ...) to create, view and maintain their knowledge of this area and have a common and standardized medical language, the second ontology contains 74 Web services, and it is used to guide software agents to discover, select, call, compose, invoke, and execute automatically a Web services without the intervention of human beings.

As future work we suggest to:

- Use the domain ontology to research medical analysis information.
- Use the Medical Analysis Web service ontology to guide the machine to discover, select, and compose automatically Web services.
- Expand the domain ontology by adding new concepts, and to make it a general ontology of medical analysis.
- Extend the Web services ontology, with the addition of new medical analysis Web services.

7. REFERENCES

- [1] T. Berners-Lee, J. Hendler and O. Lassila, (2001). "The Semantic Web". *Scientific American*, 284(5):34–43.
- [2] E. Newcomer, "Understanding Web Services XML, WSDL, SOAP, and UDDI", Pearson Education 3rd Edition August 2004, CANADA.
- [3] R. Studer, S. Grimm, A. et A. Abecker, "Semantic Web Services Concepts, Technologies, and Applications", Springer, Germany, 2007.
- [4] HN. Talantikite, D. Aissani, "Semantic annotations for web services discovery and composition", *Computer Standards & Interfaces, Volume 31, Issue 6*, Mira University of Béjaia Algeria, November 2009, Pages 1108-1117.
- [5] F. Zaidi, M. Touahria, "Utilisation des technologies de services Web sémantiques dans le domaine médical", In ICIST 2011: *Proceedings of the Information Systems and Technologies*, pp.585–594, TEBESSA, ALGERIA, 2011.
- [6] GP. Asunción, FL. Mariano, O. Corcho, "Ontological Engineering: With Examples from the Areas of Knowledge Management, E-commerce and the Semantic Web". Springer, 2004.
- [7] D. L. Martin, M. Paolucci, S. A. McIlraith, M. H. Burstein, D. V. McDermott, D. L. McGuinness, B. Parsia, T. R. Payne, M. Sabou, M. Solanki, N. Srinivasan, et K. P. Sycara, "Bringing Semantics to Web Services : The OWL-S Approach", *Lecture Notes in Computer Science*, Springer, 2004, pp. 26–42.
- [8] Horrocks, P. F. Patel-Schneider et F. Van Harmelen, 2003. "From shiq and rdf to owl : The making of a web ontology language". *Journal of Web Semantics* (1), 726.
- [9] H. Lausen, A. Polleres, et D. Roman, "Web service modeling ontology (WSMO)", Technical report, W3C, 2005. <http://www.w3.org/Submission/WSMO/>.
- [10] S. Izza, "Intégration des systèmes d'information industriels, une approche flexible basée sur les services sémantiques", PhD Thesis, ENSM Saint-Etienne, November 2006.
- [11] M. Sabou, C. Wroe, C. Goble, H. Stuckenschmidt, "Learning domain ontologies for semantic Web service descriptions", *Web Semantics: Science, Services and Agents on the World Wide Web*, Volume 3, Issue 4, December 2005, Pages 340-365
- [12] Mariano FERNANDEZ, Asuncion GOMEZ-PEREZ et Natalia JURISTO. "Methontology: from ontological art towards ontological engineering". In *Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering*, pages 33–40, Stanford, USA, March 1997.

⁶ <http://www.cocoon-health.com>

⁷ <http://www.srdc.metu.edu.tr/webpage/projects/artemis/>