Bringing Context to Intentional Services

Salma Najar
Centre de Recherche en Informatique-Université Paris1
90, rue de Tolbiac 75013 Paris - France
Citypassenger
1, avenue de l'atlantique 91976 Courtaboeuf – France
Salma.Najar@malix.univ-paris1.fr

Manuele Kirsch-Pinheiro, Carine Souveyet
Centre de Recherche en Informatique-Université Paris1
90, rue de Tolbiac 75013 Paris - France
Manuele.Kirsch-Pinheiro@univ-paris1.fr,
Carine.Souveyet@univ-paris1.fr

***Abstract*—In service-orientation, the notion of service is used in different views. On the one hand, several approaches have been proposing services that are able to adapt themselves according to the context in which they are used. On the other hand, some researches have been proposing to consider user's goals when proposing business services. We believe that these two views are complementary. A goal is only meaningful when considering the context in which it emerges, and conversely, context description is only meaningful when associated with a user goal. In order to take profit of both views, we propose to extend the OWL-S service description by including on it both the specification of context associated with the service and the goal that characterize it**.

Keywords-OWL-S; SOA; intentional service; context aware service.

I. INTRODUCTION

Service-Oriented Architecture (SOA) is a computing paradigm lying on the notion of service as fundamental element for developing software applications [16]. Its key feature is the notion of services, which stands to independent entities, with well-defined interfaces that can be invoked in a standard way, without requiring the client to have knowledge about how the service actually performs its tasks [5].

SOA can be viewed through multiple lenses, from the IT perspective up to business leaders [27]. The notion of service is used on different abstraction levels. Technically, it refers to a large variety of technologies (Web Services, ESB [21], OSGI [15], etc.). On a business level, services are proposed as a way to respond to high-level user requirements.

On the one hand, we can observe a tendency to context-awareness and adaptation on services. Several authors [10][24][25][26] have been proposing services that are able to adapt themselves according to the context in which they are used. These services are usually called context-aware services [10]. Their importance is growing with the development of pervasive and mobile technologies. Context-aware services focus on service adaptation considering the circumstances in which it is requested. However, considerations such as why context is important and what is its impact to the user's needs remain underestimated.

On the other hand, research has pointed out the importance of considering user's requirements on service orientation. Several works [7][13][16][19] proposed to take into account user's goals when proposing business services. According to these works, a service is supposed to satisfy a given user's intention. However, even when considering high level services, as business services, one should consider variability related to context on service execution. Several authors have been considering the influence of context information on business process [20] [22]. This influence remains whenever such processes are implemented through business services. Such services still have to cope with the context in which they are called.

Therefore, we have two separated views of service orientation. First, we have an extremely technical view, which focuses on technical issues needed to execute and adapt service in highly dynamic environments. In the opposite, we have a high level view, which focuses on user's requirements. The latter considers why a service is needed, without necessarily considering how it is executed, neither in which circumstances it is performed. More than the execution context, this high level view ignores the context in which user's goals emerges, while technical view passes over user's goals behind observed context information.

We believe that these two views are complementary and should not be isolated from each other. Fully potential of service orientation will not be reached if we do not consider both points of view: *goal-based services* and *context-aware services*. For us, a goal is only meaningful when considering it in a given context and a context description is only meaningful when associated with a user goal. However, this goal is not a simple coincidence; it emerges because he is under a given context.

In this paper, we propose a semantic description of services that encompass the description of the goals service can satisfy and the context in which this goal is meaningful.

This paper is organized as follows: Section II presents an overview on related work. Section III introduces the notion of goal and its representation, while Section IV presents the notion of intentional service. In Section V, we discuss the notion of context and its representation. In Section VI, we propose a semantic descriptor for intentional and context-aware services. And finally, we conclude in Section VII.

II. RELATED WORKS

A service can be seen as an independent and easily composed application that can be described, discovered and invoked by other applications and humans. In the last decade, the notion of service has evolved, from simple Web services to semantic Web services [12]. Indeed, we could observe an important tendency for semantically describing

 ISBN: 978-1-61208-152-6

services, in order to handle potentially ambiguous service descriptions [12]. Such semantic description is based on richer representation languages, mainly OWL-S [11], which provides a comprehensive specification of a service.

A semantic description is one of the building blocks of context-aware services. *Context-aware services* can be defined as services which description is associated with contextual (notably non-functional) properties. Several authors [24][26] have been proposing context-aware services, whose importance is growing with the development of pervasive and mobile technologies. An illustration of this phenomenon is given by [24], who propose improving service modeling, based on OWL-S, with context information (user information, service information and environment information). Suraci *et al.* [24] focus on adapting service composition to the user's requirements concerning context (device capabilities, user's location, etc.). These authors consider that user should be able to specify contextual requirements corresponding to the service he is looking for (availability, location, etc.), as well as to the context provided by the environment (wireless connection, etc.).

Other authors, such as [26], also advocate for representing context requirements when describing context-aware services. Toninelli *et al.* [26] consider that, in pervasive scenarios, users require context-aware services that are tailored to their needs, current position, execution environments, etc. Therefore, service modeling should be improved, including contextual information. Such a semantic modeling contributes not only to handle problems related to service interoperability, but also to consider different aspects of the environment in which the service is executed.

A different point of view is given by [1][7][13], which highlight the importance of considering user's requirements on service orientation. According to them, a service is supposed to satisfy a given user's intention, formalized as a user goal, which becomes central to service definition.

Among these works, [7] and [19] propose a service oriented architecture based on an intentional perspective. Such architecture proposes the notion of *intentional service*, which represents a service focusing on the intention service allows to satisfy rather than the functionality it performs. Besides, Mirbel e*t al.* [13] propose goal-based service discovery mechanisms. They propose a semantic approach guided by the user's intentions, in which user's requests are expressed using semantic Web technologies

None of these works considers the notion of context, contrary to [1], which proposes a goal-based dynamic service discovery and composition framework that uses context information. Nevertheless, context information is used only for filtering the input of the user's request.

All these works represent two different views of service orientation: (i) one view proposing a context-aware based approach, which focuses on the adaptation of services according to the context information; and (ii) a second view focusing on a goal-based approach, proposing high level services, which focuses on user's goals. The first view focuses on service composition on a highly dynamic environment, without considering why service is needed. The latter considers this question without considering context in which this need emerges.

Questions such as "*why a service is useful in a given context*?" or "*in which circumstances a service need raises*?" remain unexplored. For us, *a goal is only meaningful when considering it in a given context and a context description is only meaningful when associated with a goal*. In order to explore both views, we have first to represent them in a semantic way. Thus, we propose a semantic description of services that encompass notions of context and intention.

III. UNDERSTANDING USER GOALS

Several researches in service engineering [4][7][17] focus on the adoption of goal-based approaches from requirements engineering to identify user's requirements and intentions. This vision is the base for several goal-based approaches [1][23], which propose to take into account user's goals when proposing business services.

The term goal has several different meanings. According to [6], a goal is an "optative" statement expressing a state that is expected to be reached or maintained. The intention represents the goal that we want to achieve without saying how to perform it [7]. Bonino *et al.* [1] defines an intention as a goal to be achieved by performing a process presented as a sequence of intentions and strategies to the target intention. Even if they differ, all these definitions let us consider an intention as a *user's requirement representing the goal that a user wants to be satisfied by a service without saying how to perform it.*

To ensure a powerful intention matching, the intention is formulated according to a template [7][19], represented in Fig. 1. In this template, a goal is expressed by a verb, a target and a set of optional parameters, which play specific roles with respect to the verb.

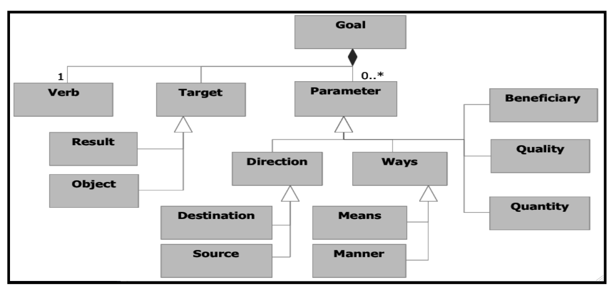

Fig 1. Goal template based on [7]

IV. INTENTIONAL SERVICE AT THE GLANCE

Recently, several authors have considered a direct participation of end user on service specification. Brnsted *et al.* [2] illustrate this tendency by observing several approaches allowing end users to actively interact with service composition specification. However, these authors do not consider whether terminology used by these tools correspond to the user's current vocabulary. The question that emerges here is the following: are these users technical people, who are familiarized with service-oriented technology, or are these users business actors who are totally unaware of technical considerations?

To bridge the gap between high-level business services, comprehensible by the business actors, and low-level software services, understandable by technical people, an intentional description is proposed [7] [19]. This user-centric perspective forms the so-called *Intentional Service Oriented Architecture (ISOA)*. ISOA represents services at a high level of abstraction, referring to the intention they can fulfil rather than the function they perform. Such services, named *intentional services*, are expressed in terms of intentions and strategies to achieve them.

A. *Defining intentional services*

An intentional service is a service captured at a high-level, in business comprehensible terms and described in an intentional perspective. The intentional service model (ISM) [7] [19] associates to each service an intention it can satisfy. It is composed of 4 facets, represented in Fig. 2, namely the *service interface*, the *service behaviour*, the *service composition* and the *QoS*.

The *service interface* represents the service that permits the fulfilment of an intention, given an initial situation and terminating in a final situation. The *service behaviour* specifies the pre and post conditions that represent the sets of initial states required by the service for the goal achievement, and the set of final states resulting from goal achievement. The *service composition* represents the possibility of composing more complex goals by combining lower abstraction level goals. Next section gives more precisions about service composition. Finally, the *QoS* introduces the non-functional dimension of service. It represents the quality requirements associated with intentional services.

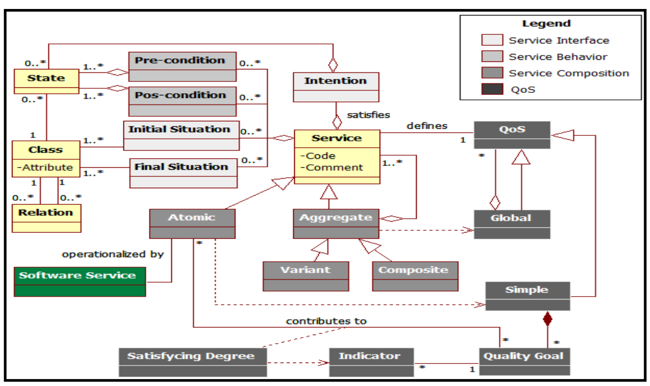

Fig 2. Intentional Service Model (ISM) [19]

B. *Composing intentional services*

The intentional service model emphasises variability on the satisfaction of its corresponding intention. It allows the variability through the service composition. In the ISM model, an intentional service can be *aggregate* or *atomic*. Aggregate services represent high-level intentions that can be decomposed in lower level one, helping business people to better express their strategic/tactical intentions.

Intentional composition admits two kind of aggregate services: a *composite* and a *variant*. While composite services reflect the precedence or succession relationship between two intentions, variant service correspond to the different manner to achieve an intention. This need for variability is justified by the need to introduce flexibility in intention achievement.

According to [19], atomic services are related to operationalized intentions and can be fulfilled by SOA functional services. Atomic intentional services are then operationalized by software services. In contrast, aggregate services have high-level intentions that need to be decomposed in lower level ones till atomic intentional services are found.

Nevertheless, this vision does not consider the evolution of service technology, which can stand now for small pieces of software encapsulating reusable functionalities, as well as for large legacy systems, whose complex process are hidden by technologies such as Web Services or ESB [21]. By considering that only atomic services can be operationalized by software service, ISOA architecture limits the reuse of such legacy systems under an intentional approach.

In this paper, we consider that both atomic and aggregate intentional services can be operationalized by software service, which can be also atomic or composite. As a consequence, both technical and intentional compositions are possible independently, allowing more powerful constructions. Section VI describes how both can coexist in the proposed service semantic description.

V. DESCRIBING CONTEXT INFORMATION

In the last decade, an important change has been performed on the way we work and on the way technology support us. We pass from a quite static model, in which people use to interact with business process only during their "work time" in well-defined circumstances (in their offices, with their desktop computers) to "mobile worker" model. With the evolution of mobile technologies, and notably smartphones, this static model does not fit anymore.

As a consequence of this evolution, information systems should now consider not only the tasks a user can (or must) perform, but also the context in which such user finds himself when performing an action. Context information corresponds to a very wide notion. It is usually defined as any information that can be used to characterize the situation of an entity (a person, place, or object considered as relevant to the interaction between a user and an application) [3]. The notion of context is central to context-aware services that use it for adaptation purposes. Context information can stand for a plethora of information, from user's location, device resources [18], up to user's agenda and other high level information [8]. Nevertheless, in order to perform such adaptation processes, context should be modelled appropriately. The way context information is used depends on what it is observed and how it is represented. The context-adaptation capabilities depend on the context model [14].

Different kinds of formalism for context representation have been proposed. Nevertheless, an important tendency can be observe on most recent works: the use of ontology for context modelling [14]. According to [14], different reasons motivate the use of ontologies, among them their capability of enabling knowledge sharing in a non-ambiguous manner and its reasoning possibilities. This tendency follows the evolution of context-aware services, which adhere, in their majority, to a semantic description of such services. In this paper, we also adhere to this tendency, adopting an ontology-based context modelling based on [18].

VI. PUTTING EVERYTHING TOGETHER: DESCRIBING CONTEXT-AWARE INTENTIONAL SERVICES

The latest research in service oriented computing recommends the use of the OWL-S for semantically describe services [24]. Even if OWL-S is tailored for Web services, it is rich and general enough to describe any service [24]. OWL-S [11] defines web service capabilities in three parts representing interrelated sub-ontologies named service profile, process model and grounding. The *service profile* expresses what the service does. It gives a high-level description of a service, for purposes of advertising, constructing service requests and matchmaking. The *process model* answers to the question: how is it used? It represents the service's behaviours as a process and describes how it works. Finally, the *grounding* maps the constructs of the process model onto detailed specification of message formats, protocols and so forth (often WSDL).

OWL-S represents a flexible and extensible language, as demonstrated by works such as [9][24]. Similar to these works, we propose to extend service description in OWL-S by including information concerning both context and goal that characterize a service.

A. *Describing service intentions in OWL-S*

According to an intentional perspective, a user requires a service because he has a goal that the service is supposed to satisfy. Hence, the importance of considering user's goals emerges on service orientation. Such goal is formalized as an intention, which becomes central to service definition.

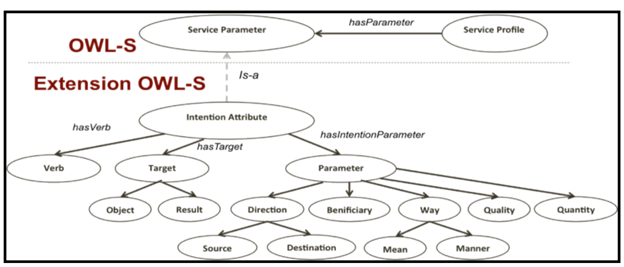

Fig 3. Service intention Description in OWL-S profile

OWL-S profile provides features to expose characteristics of a service through service parameter [11]. We propose to extend service profile by adding a new parameter named *intention* attribute, which describes the intention associated with the service (see Fig. 3). This parameter is described using the template *"Verb, Target, Parameters"* (see Section III). Thus, the intention attribute of a service assures to the service an intentional interpretation.

Fig. 3 details elements we added to OWL-S profile description. First of these elements is the *intention* attribute itself, which is an extension from OWL-S service parameter. A service intention has three properties: a *verb*, a *target* and a *parameter*. A *verb* characterizes the user's intention. Possible verbs can be organized in a verb ontology that recognizes significant verbs for a given community. A *target* indicates either a *result* from the satisfaction of the goal, or an *object* that exists before the achievement of the goal. Finally, a *parameter* represents additional information needed by the *verb*.

```
9     <profile:serviceParameter>
10        <eiprofile:Intention rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/Intention.owl#concept.intention">
11            <profile:sParameter
12                <eiprofile:Verb rdf:ID="http:// www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/Intention.owl#concept.intention.verb.pay">
13                    Pay
14                </eiprofile:Verb>
15                <eiprofile:Target rdf:ID=vhttp:// www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/Intention.owl#concept.intention.target">
16                    <eiprofile:Object rdf:ID="http:// www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-
                      S/Intention.owl#concept.intention.target.object.booking">
17                        Booking
18                    </eiprofile :Object>
19                </eiprofile:Target>
20                <eiprofile:Parameter rdf:ID="http:// www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/Intention.owl# concept.intention.parameter">
21                    <eiprofile:Way rdf:ID="http:// www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-
                      S/Intention.owl#concept.intention.parameter.way">
22                        <eiprofile:Mean rdf:ID="http:// www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-
                          S/Intention.owl#concept.intention.parameter.way.mean.credit-card">
23                            Credit Card
24                        </eiprofile :Mean>
25                    </eiprofile:Way >
26                </eiprofile:Parameter>
27            </profile:sParameter>
28        <eiprofile:Intention>
29    </profile:serviceParameter>
```

Fig 4. Example of describing service intention in OWL-S

Fig. 4 illustrates this extension through an example of service profile that includes the intention attribute. This example presents a *booking service,* which satisfies the intention *booking payment by credit card* (lines 10-28). The lines 12-14 describe the verb *pay*, which describes the intention. The target of the intention is represented by the object *booking,* described in the lines 15-19. This intention has, as a *parameter,* the *mean* of the intention represented by the *credit card* in this example (lines 22-24).

B. *Describing contextual information in OWL-S*

A goal that a user wants to satisfy is not a coincidence; it emerges because the user is in a given location or under a given context. In our opinion, a goal is only meaningful when considering it in a given context and a context description is only meaningful when associated with a goal. According to this, we propose to extend the service profile to allow service provider to define context information that characterize an intentional service.

For instance, let us consider a *parachute jump booking service* that enables users to browse, search for, and reserve a parachute jump in different situations and according to the user preference. This service can be particularly designed considering client devices with high screen resolution and flash technologies to show video and tutorials about the parachuting; a second implementation of the same service can be designed considering, for example, a particular user profile (e.g., adult users). Such contextual information can be considered as part of the service description, since it indicates situations to which the service is better suited. According to [9], context information cannot be statically stored on the service profile due to its dynamic nature. Context properties related to service execution can evolve (e.g., server load may affect properties of services running on it), whereas service profile is supposed to be a static description of the service.

Thus, in order to handle dynamic context information on static service description, we adopt the approach [8] to enrich OWL-S service profile with a *context* attribute, which represents a URL pointing to context description file. Since context information is dynamic and cannot be statically stored on the service profile description, we opt to describe context element in external file to allow service provider to easily update such context information related to the service description itself. The context description of a service describes, from the one side, the situation status of the requested service (environment in which the service is executed), and from the other side, the contextual condition (requirement) to execute the service.

C. *Composing intentions*

Intention and context attributes described above intent to expose both aspects of a service notably for discovery purpose. Thanks to the OWL-S extension we propose, a service can be discovered either by intention it can satisfy, or by the context associated with this intention. In addition to these aspects, a third aspect should be exposed: the service variability. Such variability is expressed, in the intentional perspective, by the composition of intentional services indicating the decomposition of the service goal on lower level goals. Thus, while technical composition of a service, described in OWL-S process model, represents software components that are combined to supply service operations, intentional composition represents not only lower level goals necessary to satisfy service goal, but also different possibilities for satisfying this goal. Technical composition supplies technical elements necessary for service execution, while intentional composition provides an understanding, from final user's point of view, of the service and the diverse forms of satisfying service goal. Thus, we propose to extend OWL-S process model by including the specification of an intentional service process (see Fig. 5).

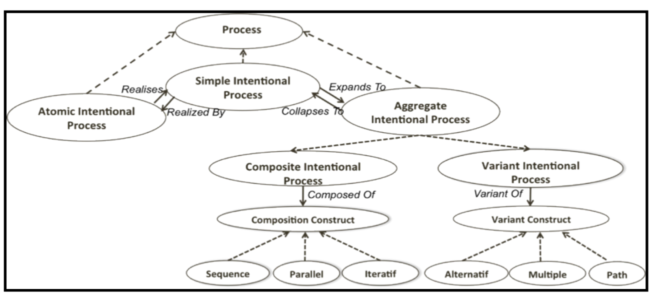

Fig 5. Composing intentions in OWL-S process Model

Fig. 5 presents the extension we propose for the process model. This extension considers two kinds of process: the atomic intentional process and the aggregate intentional process. It considers also a simple intentional process, which is used to provide an abstracted view that can be atomic or aggregate. A simple intentional process is realized by an atomic intentional process and expands into an aggregate intentional process. An aggregate intentional process can be either a composite intentional process or a variant intentional process.

The composite intentional processes reflect the precedence/succession relationship between their intentions. Such relationships are specified using composition constructs such as *Sequence*, *Parallel* and *Iterative*. The composition represents a *sequence* in which there is a sequential order between component processes, or a *parallel* in which components can run in parallel. The *iterative* construct is used when the satisfaction of a goal may require iterative execution of a given set of actions.

The variability is represented by the variant intentional process, which uses constructs such as *multiple*, *alternative* and *path*. The *multiple* construct offers a non-exclusive choice in the realization of the goal. It groups multiple simple processes, among them, at least one will be chosen. The *alternative* construct represents a process with an alternative choice that regroups several simple processes that are mutually exclusive. It builds a new process of the same level of abstraction but of higher granularity. And finally, the *path* construct offers a choice in how to achieve the goal of the aggregate process by offering composite processes that are mutually exclusive.

```
<eiprocess:CompositeIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/MakeParachuteJumpBooking">
    <eiprocess: CompositeIntenionalProcessID> MakeParachuteJumpBooking </eiprocess: CompositeIntenionalProcessID>
    <eiprocess: CompositeIntenionalProcessName> Make a parachute jump booking </eiprocess: CompositeIntenionalProcessName>
    <eiprocess: ComposedOf>
        <eiprocess: Sequence>
            <process: Components rdf:parseType="Collection">
            <eiprocess: VariableIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/PayParachuteJumpBooking">
            <eiprocess: AtomicIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/ReserveDateParachuteJumpBooking">
            <eiprocess: AtomicIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/TakeInsurance">
            </process: Components>
        </eiprocess: Sequence >
    </eiprocess: ComposedOf>
</eiprocess: CompositeIntenionalProcess >
```

Fig 6. Example of OWL-S Intentional composition: Composite Intentional Process

For instance, let us consider the example a service named $S_{\textit{Confirm parachute jump booking}}$, which intents making a confirmed parachute jump booking. It is described as a variant service that represents a *path* between the composite service $S_{\textit{Get a rewarded parachute jump booking}}$ and the service $S_{\textit{Make parachute jump booking}}$ (see Fig. 6). This latest is composed of a sequence of the variant service $S_{\textit{Pay parachute jump booking,}}$ the atomic services $S_{\textit{Reserve the date of the parachute jump}}$ and service $S_{\textit{take an insurance}}$ (see Fig. 7).

```
<eiprocess: VariableIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/confirmParachuteJumpBooking">
    <eiprocess: VariableIntenionalProcessID> ConfirmParachuteJumpBooking   </eiprocess: VariableIntenionalProcessID>
    <eiprocess: VariableIntenionalProcess Name> Confirmation parachute jump booking </eiprocess: VariableIntenionalProcessName>
    <eiprocess: VariantOf>
        <eiprocess:Path>
            <process:Components rdf:parseType="Collection">
                <eiprocess: CompositeIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/getRewardedParachuteJumpBooking">
                <eiprocess: VariableIntenionalProcess rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/MakeParachuteJumpBooking">
            </process: Components>
        </eiprocess: Path>
    </eiprocess:VariantOf>
</eiprocess:VariableIntenionalProcess >
```

Fig 7. Example of OWL-S Intentional composition: Variant Intentional Process

 ISBN: 978-1-61208-152-6

Thanks to the OWL-S extension proposed here, we enable the description of intentional composition, from final user's point of view. This extension exposes the variability representing different manners to satisfy user's goals. The intentional composition description allows a service discovery guided by intention, presented at a high level.

VII. Conclusions and Future Works

In this paper, we considered context-aware and goal-based service orientation as complementary approaches that should not be isolated from each other. We explain our belief that a goal is only meaningful when considering it in a given context and a context description is only meaningful when associated with other goal. We propose, consequently, to enrich OWL-S service description, by including the description of the goals service can satisfy as well as context in which this goal is meaningful, context in which service is (or can be) executed. From the one side, we propose to enrich service description with knowledge about goals and composition of goals that are meaningful for final users, who request the service. From the other side, we propose to enrich this service description with context information necessary for adapting such service. By proposing such a semantic description of service, we enable the expression of services that can adapt themselves to context of use and that represent a formulated user's requirements. By exposing both aspects of a service, we develop a context-aware goal-based service oriented framework. This framework is currently under evaluation.

References

[1] L.O. Bonino da Silva Santos, G. Guizzardi, L.F. Pires, and M. Van Sinderen, "From User Goals to Service Discovery and Composition," ER Workshops, pp. 265-274, 2009.

[2] J. Brnsted, K. Hansen, and M. Ingstrup, "Service Composition Issues in Pervasive Computing," IEEE Pervasive Computing, vol. 9 n° 1, pp. 62 -70, 2010.

[3] A. Dey, "Understanding and using context, Personal and Ubiquitous Computing," vol. 5 n°1, pp. 4-7, 2001.

[4] J. Gomez, M. Rico, and F. Garcia-Sanchez, "GODO : Goal Oriented Discovery for Semantic Web Services," WIW Workshop on WSMO Implementations, 2004.

[5] V. Issarny, M. Caporuscio, and N. Georgantas, "A Perspective on the Future of Middleware-based Software Engineering," In: Briand, L. and Wolf, A. (Eds.), Future of Software Engineering 2007 (FOSE), ICSE (Conf on Software Engineering), IEEE-CS Press.

[6] M. Jackson, "Software Requirements and Specifications: A lexicon of practice, principles and prejudices," Addison Wesley Press, 256, 1995.

[7] R.S. Kaabi and C. Souveyet, "Capturing intentional services with business process maps," RCIS, pp. 309-318, 2007.

[8] M. Kirsch-Pinheiro, J. Gensel, and H. Martin, "Representing Context for an Adaptative Awareness Mechanism," G.-J. de Vreede; L.A. Guerrero, G.M.Raventos (Eds.), LNCS 3198 - X Workshop on Groupware (CRIWG 2004), pp. 339-348, 2004.

[9] M. Kirsch-Pinheiro, Y. Vanrompay, and Y. Berbers, "Context-aware service selection using graph matching," 2nd Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC'08), ECOWS. CEUR Workshop proceedings, vol. 411, 2008.

[10] Z. Maamar, D. Benslimane, and N.C. Narendra, "What can context do for web services?," Communication of the ACM, vol. 49 n° 12, pp. 98-103, 2006.

[11] D. Martin, M. Paolucci, S. Mcilraith, M. Burstein, D. Mcdermott, D. Mcguinness, B. Parsia, T. Payne, M. Sabou, M. Solanki, N. Srinivasan, and K. Sycara, "Bringing Semantics to Web Services: The OWL-S Approach," Cardoso, J. & Sheth, A. (Eds.), SWSWPC 2004, LNCS 3387, Springer, pp. 26-42, 2004.

[12] S.A. Mcilraith, T.C. Son, and H. Zeng, "Semantic Web Services, IEEE Intelligent Systems," vol. 16, pp. 46-53, 2001.

[13] I. Mirbel and P. Crescenzo, "From end-user's requirements to Web services retrieval: a semantic and intention-driven approach," J.-H. Morin, J. Ralyte, M. Snene, "Exploring service science", First International Conference, IESS 2010, LNBIP 53, Springer, pp. 30-44, 2010.

[14] S. Najar, O. Saidani, M. Kirsch-Pinheiro, C. Souveyet, and S. Nurcan, "Semantic representation of context models: a framework for analyzing and understanding," J. M. Gomez-Perez, P. Haase, M. Tilly, and P. Warren (Eds),1st Workshop on Context, information and ontologies (CIAO 09), European Semantic Web Conference (ESWC), ACM, pp. 1-10, 2009.

[15] OSGi Alliance, http://www.osgi.org/ (Jan 2011).

[16] M.P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann, "Service-Oriented Computing: A Research Roadmap," Int. J. Cooperative Inf. Syst. vol 17 n° 2, pp. 223-255, 2008.

[17] L. Penserini, A. Perini, and A. Susi, "High Variability Design for Software Agents: Extending Tropos," ACM Transaction on autonomous and Adaptative Systems, vol.2 n°4, 2007.

[18] R. Reichle, M. Wagner, M. Khan, K. Geihs, L. Lorenzo, M. Valla, C. Fra, N. Paspallis, and G.A. Papadopoulos, "A Comprehensive Context Modeling Framework for Pervasive Computing Systems," In 8th IFIP Conf on Distributed Applications and Interoperable Systems (DAIS), Springer.

[19] C. Rolland, M. Kirsch-Pinheiro, C. Souveyet, "An Intentional Approach to Service Engineering," IEEE Transactions on Service Computing, vol.3 n°4, pp. 292-305, 2010.

[20] M. Rosemann, J. Recker, and C. Flender, "Contextualization of Business Processes," Int. J. Business Process Integration and Management, vol. 1 n°1/2/3, 2007.

[21] W. Roshen, "SOA-Based enterprise integration: a step-by-step guide to services-based application integration," McGraw Hill, 2009.

[22] O. Saidani and S. Nurcan, "Towards Context Aware Business Process Modeling," 8th Workshop on Business Process Modeling, Development, and Support (BPMDS'07), CAiSE'07, 2007.

[23] M. Stollberg and B. Norton, "A Refined Goal Model for Semantic Web Services," IEEE, Second International Conference on Internet and Web Applications and Services (ICIW'07), pp. 17, 2007.

[24] V. Suraci, S. Mignanti, and A. Aiuto, "Context-aware Semantic Service Discovery," 16th IST Mobile and Wireless Communications Summit, pp. 1-5, 2007.

[25] N. Taylor, P. Robertson, B. Farshchian, K. Doolin, I. Roussaki, L. Marshall, R. Mullins, S. Druesedow, and K. Dolinar, "Pervasive Computing in Daidalos, Pervasive Computing," vol. 10 n° 1, pp. 74 - 81, 2011.

[26] A. Toninelli, A. Corradi, and R. Montanari, "Semantic-based discovery to support mobile context-aware service access," Computer Communications, vol.31 n° 5, pp. 935-949, 2008.

[27] R. Welke, R. Hirschheim, and A. Schwarz, "Service-oriented architecture maturity," IEEE Computer, vol. 44 n° 2, pp. 61-67, 2011.

 ISBN: 978-1-61208-152-6