A Comparitive study: FLOWS and PSL model in Selecting the Ontologies for Dynamic Web service Selection in Semantic web Environment

[1]Michael Raj T.F , [2]Rajesh.K
Asst.Prof., Department of CSE,SASTRA University,Tanjure,INDIA-612001

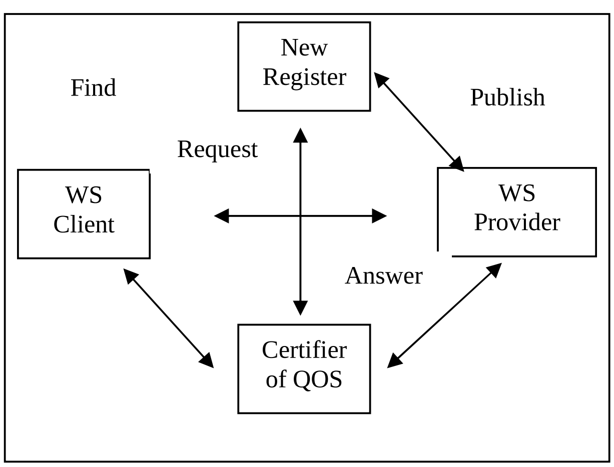
***Abstract*— Dynamic web service selection is a problem in semantic web environment because of the selection of ontologies. Web services(WSs) will be the building blocks of for developing the next generation applications using the service oriented architecture(SOA). Semantic web service ontologies are more important in defining the web service identification and selection. Web service provision is the process of assigning particular services to the continent tasks of business processes. Keyword-based search has been popularized by Web search services. However, due to the problems associated with polysemy and synonym, users are often unable to get the exact information they are looking for. In this paper we analyze the environment and identifying and comparing the ontology models and proposing best model for the web service selection.**

Keywords-Semantic web, ontologies, Flows, PSL, web service

I. INTRODUCTION

There are increasing amount of Web services being made available in the Internet, and an efficient Web services composition would help to integrate different algorithm together to provide a variety of services. There is widely this kind of the composition of Web services in e-commerce. The Semantic Web plays an important role in making the Web more relevant [1]. The data and rules are systematically described so that they can be shared and used by distributed agents. The main components implementing this Web vision include techniques such as XML for adding arbitrary structures to documents; RDF, to express meaning by simple statements about things having properties with values; and ontology, to formally describe concepts and their relationships.

A typical ontology is an explicit specification of a conceptualization [2], but the semantic Web still faces major problems in a context. The semantic Web services are located at the cross roads of two major research areas of the net technology: the Semantic Web and Web services. The aim of semantic Web services is to create a semantic Web service whose properties, capabilities, interfaces and effects are unambiguously described and used by machines [3]. Semantics used will allow the automation features needed for effective collaboration between companies, namely; description and publication services, discovery services, selection of services, composition of services, and provision and administration services. In [4], the author proposes a scheme based on the quality of service. This is a model set that can coexist with UDDI registries. It consists of four roles, namely; web services provider, web services client, the certifier of quality of service, and the new register.



This paper is organized as follows, Section-II semantic web service ontology, section-III Related work, section-IV Research challenges, section-V concludes the paper and section –VI references.

II - SEMANTIC WEB SERVICE ONTOLOGY

The Semantic Web Services Ontology (SWSO). This includes a description of the conceptual model underlying the ontology, and a description of a first-order logic (FOL) axiomatization which defines the model-theoretic semantics of the ontology. The axiomatization is called "SWSO-FOL" or equivalently, FLOWS -- the First-order Logic Ontology for Web Services -- and is expressed using SWSL-FOL.

The Semantic Web Services Ontology (SWSO) ontology is expressed in two forms: FLOWS, the First-order Logic Ontology for Web Services; and ROWS, the Rules Ontology for Web Services, produced by a systematic translation of FLOWS axioms into the SWSL-Rules language.

A. FLOWS

The goal of FLOWS is to enable reasoning about the semantics underlying Web services, and how they interact with each other and with the "real world". FLOWS does not strive for a complete representation of web services, but rather for an abstract model that is faithful to the semantic aspects of service behavior. FLOWS has been specified in SWSL-FOL, the first-order logic language.

This includes a description of the conceptual model underlying the ontology, and a description of a first-order logic (FOL) axiomatization which defines the model-theoretic semantics of the ontology. The axiomatization is called "SWSO-FOL" or equivalently, FLOWS -- the First-order Logic Ontology for Web Services -- and is expressed using SWSL-FOL.

FLOWS is intended to enable reasoning about essential aspects of Web service behavior, for a variety of different purposes and contexts. Some targeted purposes are to support descriptions of Web services that enable automated discovery, composition, and verification, and creation of declarative descriptions of a Web service, that can be mapped to executable specifications.

FLOWS captures the salient, functional elements of various models of Web services found in the literature and in industrial standards. Models focused on specifying semantic Web services including the OWL-S model [7] of atomic processes, their inputs, outputs, preconditions and effects, and the associated notion of impacts "on the world" and testing conditions "about the world"; and provides the ability to model, albeit in an abstract, semantically motivated manner, several key Web services standards, including WSDL, BPEL, and WS-Choreography. A primary difference between FLOWS and OWL-S is the expressive power of the underlying language.

B. PSL

The purpose of PSL-Core is to axiomatize a set of intuitive semantic primitives that is adequate for describing the fundamental concepts of manufacturing processes. Consequently, this characterization of basic processes makes few assumptions about their nature beyond what is needed for describing those processes, and the Core is therefore rather weak in terms of logical expressiveness.

The basic ontological commitments of PSL-Core are based on the following intuitions:

Intuition 1:

There are four kinds of entities required for reasoning about processes -- activities, activity occurrences, timepoints, and objects.

Intuition 2:

Activities may have multiple occurrences, or there may exist activities that do not occur at all.

Intuition 3:

Timepoints are linearly ordered, forwards into the future, and backwards into the past.

Intuition 4:

Activity occurrences and objects are associated with unique timepoints that mark the begin and end of the occurrence or object.

Organization of PSL

• PSL is a modular, extensible ontology capturing concepts required for process specification
• There are currently 300 concepts across 50 extensions of a common core theory (PSL-Core), each with a set of axioms written using the Knowledge Interchange Format.
• Two kinds of extensions:
    • Core theories
    • Definitional extensions

III - RELATED WORK

The FLOWS process model is created as a family of extensions of PSL-OuterCore, The fundamental extension of PSL-OuterCore for services is called "FLOWS-Core". As will be seen, this extension is quite minimalist, and provides an abstract representation only for (web) services, their impact "on the world", and the transmission of messages between them.

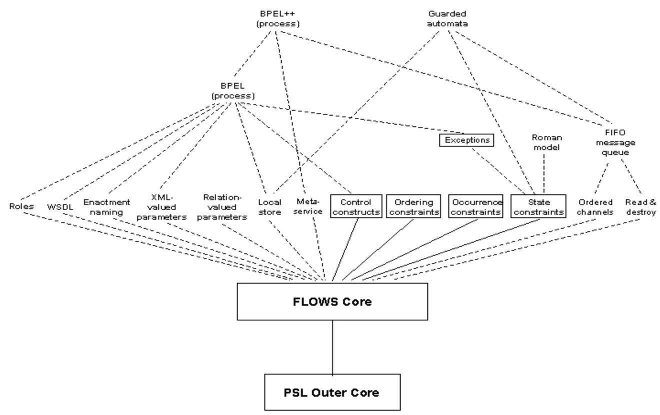


Figure 3.1

Figure 3.1 shows various families of process modeling constructs, including PSL-OuterCore, FLOWS-Core, and several others. These include models from standards, such as the process model portion of BPEL (and a refinement of BPEL that incorporates typical assumptions made when using BPEL in practice), and models from the research literature, such as guarded automata. A line between a family F of constructs and a family F' above F indicates that it is natural to view F' as a family that includes most or all of the constructs in F. The OWL-S process model is not shown explicitly in Figure 3.1, because it has been integral in the design of the FLOWS-Core and the Control Constructs extension. Of course, the different families of modeling constructs shown in the figure should not be viewed as comprehensive.

In some cases, it may be useful to create PSL extensions of FLOWS-Core, to formally specify the properties of certain families of constructs shown in Figure 3.1. Indeed, as part of the Process Model of the FLOWS ontology presented below several extensions are specified; these are indicated in the figure by the rectangles above FLOWS-Core. We note that FLOWS-Core can serve as a mathematical foundation for representing several other aspects and models from the Web services and SOA standards and literature, and it seems likely that additional PSL extensions of FLOWS-Core will be created in the future.

FLOWS is an abstract model that is faithful to the semantic aspects of service behavior. The FLOWS models uses the predicates and terms that can change value over time. It provides the structure for representing messages between services. It focus on semantic construct of a message rather than packaged into an XML-based message. FLOWS captures the salient, functional elements of various models of Web services found in the literature and in industrial standards.

A key premise of FLOWS is that an appropriate foundation for formally describing Web services can be built as a family of PSL extensions. PSL -- the Process Specification Language. In a typical usage of FLOWS, an application domain is created by combining the FLOWS axioms with additional logical sentences to form a (first order logic) theory. Speaking loosely, each sentence in such a theory can be viewed as a constraint or restriction on the models

The FLOWS Process Model Ontology Modules

AtomicProcess

composedOf

message

channel

Control Constraints

Sequence

Unordered

Choice

IfThenElse

Iterate

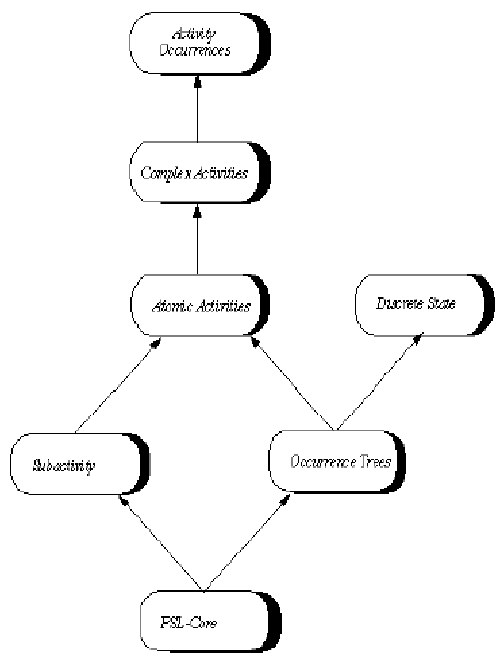
RepeatUntil

Ordering Constraints

Occurrence Constraints

State Constraints

Exception Constraints



B Formal Properties of PSL

• The meaning of terms in the ontology is characterized by models for first-order logic.
• The PSL Ontology has a first-order axiomatization of the class of models.

• Classes in the ontology arise from classification of the models with respect to invariants (properties of the models preserved by isomorphism).
• Process descriptions are specified by definable types for elements in the models.

C. Select candidate ontologies

The selection step did not have to cope with the issue of discovering potential reuse candidates, as the set of reusable ontologies was limited to the Ontolingua repository. However, this step covers a detailed report on the evaluation procedure which unsuccessfully attempted to apply existing reusability-assessment approaches. This process step resulted in the selection of a single ontology subjectively perceived to be useful for the application context.

D. Customize and integrate relevant ontologies

Due to the poor application relevance results obtained in the previous step, the integration was restricted to extracting particular fragments of the selected ontology, which were subsequently embedded in the application system.

E. QoS description and evaluation

W3C published in 2003, which defined the quality of service. put a lot of effort on building QoS ontology. Using a third party to measure the QoS is easy to implement, but exists geneogenous shortage. In this paper, we use QoS Ontologies to describe services' and requests' QoS constraints, adopt user feedback mechanism to get the dynamic QoS parameters[8].

IV – RESEARCH CHALLENGES

- FLOWS-Core does not provide any explicit constructs for the structuring of processing inside a Web service.
- Some minimal constructs are available in PSL OuterCore, including primarily the soo_precedes predicate, which indicates that one atomic activity occurrence in a complex activity occurrence precedes in time another atomic activity occurrence.
- This is intentional, as there are several models for the internal processing in the standards and literature (e.g., BPEL, OWL-S Process Model, Roman model, guarded automata model), and many other alternatives besides (e.g., rooted in Petri nets, in process algebras, in workflow models, in telecommunications).
- When using FLOWS-Core, it is often useful to model humans.
- It is typical to assume that humans cannot directly perform atomic processes for testing or directly impacting domain-specific fluents, but must rather achieve that by invoking standard (web) services.

- We note that FLOWS-Core can be used to faithfully represent and study service process models that do not include all of the notions just listed. For example, OWL-S focuses on world-changing atomic processes and largely ignores the variations that can arise in connection with message passing.
- A given application domain can be represented in FLOWS-Core in such a way that the message passing is essentially transparent, so that it provides a faithful simulation of an OWL-S version of the domain.

V- CONCLUSION

This paper has introduced a comparative study for the selection mechanism of the semantic web ontology models based on the usability of the semantic composite Web service, with minimum coverage thinking of the FLOWS and PSL model rules of which can compute the semantic matching degree between demand and services. Therefore, it can provide customers with more suitable Web services in the condition of having no Web services completely or perfectly match demand. In the future, we will build an intelligent self-adaptive abstract model for work flow by dynamic way, and expand effective plan methods and future web services and SOA models, it is possible to look at data flow and information sharing at an abstract level based on what a service "knows", or to specialize this to models that involve imperative variable assignment.

VI-REFERENCES

[1] Berners-Lee T., Hendler J. and Lassila O, “The Semantic Web," *Scientific American*, pp.35-43,2001

[2] Gruber T, ”Towards Principles for the Design of Ontology Used for Knowledge Sharing,” *IEEE Transactions on Software Engineering*,

[3] Ganden F. Sadeh N.,"Semantic Web Technologies to Reconcile Privacy and Context A wareness," vol.1, no.3, pp.241-260.,avril 2004.
[4] Shuping R. ,"A model for web services discovery with QoS," ACM ,vol. 4 , issue 1, pp. 1 – 10, 2003
[5] QoS for Web Services: Requirements and Possible Approaches,http://www.w3c.or.kr/kr-office/TR/2003/ws-qos/
[6] Z. Xu, P. Martin, W. Powley, F. Zulkernine, "Reputation-Enhanced QoS-based Web Services Discovery", *In Proceedings of the fifth International Conference on Web Services*, 2007.
[7] "OWL-S: Semantic Markup for Web Services",
[8] A QoS-oriented Optimization Model for Web Service Group, 2009 Eight IEEE/ACIS International Conference on Computer and Information Science