International Journal of u- and e- Service, Science and Technology
Vol.8, No. 10 (2015), pp.405-414
http://dx.doi.org/10.14257/ijunesst.2015.8.10.39

A Study on the Computation Model for the Product Service System Development

Zhang Fei and Xu Yang

Institute of Mechanical Engineering, China Jiliang University, Hangzhou 310027, China
E-mail: zhfei@cjlu.edu.cn, enihsyx@163.com

Abstract

Today the manufacturing industries must transform their focus from the traditional physical products to the service embedded products, thus the industrial product service system design is very essential. The system should satisfy the demands of flexibility, open, and re-configurability of enterprises' resources, model for service resource based on ontology was referred. The hierarchy-relation and properties of resource were described, and the property-oriented metadata structure of resource was constructed according to the resource description framework (RDF). And furthermore based on the knowledge ontology, the intrinsic of resource and the hierarchy relation of resource were described. Finally in protégé environment, the construction method of service resource bases was illustrated with an example.

Keywords: *service embedded product, ontology, computation model, RDF*

1. Introduction

Manufacturing industries are currently faced a transformation from provision of physical products to servicing. That movement is related to the following three factors. First is the development of information communication technology, which enables the construction of service platforms through digitalization and networking. Accordingly, networked home electronics have been emerging and transition to a service economy thereby is accelerated. Second, in vertical value chains, value adding phases are now shifting from assembling processes to upstream and downstream processes. Upstream processes such as product development and design process or downstream process such as maintenance and after-sales service are producing higher profitability. The idea, proposed by Stan Shih, the founder of Acer Inc. is designated as a smiling curve. The third factor is the change of consumer demand from the act of product possession to act of experience through services [1-2]. Especially since the 1990s, it is said that consumers' expenses on services are steadily increasing.

In response to those strong trends, manufactures must consider a new fusion framework incorporating products and services. An offering from such design is often called industrial product service systems (PSS). PSS offerings consist of combinations of physical products, like service is embedded in the physical products, services and systems that have been integrated and optimized from a lifecycle perspective in relation to the customer value.

According to the United Nations, a PSS is a competitive system of product, services, supporting networks and infrastructure. It is made up of services such as product maintenance and parts recycling that focus on fulfilling customer needs in a manner that is competitive, life cycle oriented and environmentally friendly [3]. Several authors have defined, described and reviewed the PSS concept by focusing on research and industrial perspectives. A widely accepted approach to the design and delivery of a PSS identifies three main categories of PSS value proposition: product, use or result orientation [4-6]. In

ISSN: 2005-4246 IJUNESST

the product-oriented approach, a company offers the sale of products and also opens channels for customers to access additional services such as upgrades, maintenance, professional advice and consultancy. For the use-oriented approach, a company maintains the rights to a product for use in a service environment made accessible to customers via services such as product renting. Result-oriented approaches involve a company delivering contents of a service to customers, independent of product choice. Schemes such as paying-per-unit service, delivery functional results, activity management and outsourcing, all fall under this category [7-8].

A PSS involves considering factors relating to the life cycle of products, closely linking products and services, and establishing links with customers and other manufacturers to aid product/service delivery. Once these factors have been considered, the next step should involve the modeling of business requirements for the development of an effective and efficient PSS. It is very important to develop a PSS platform to satisfy the demand of flexibility, open, and re-configurability of company service resources. When developing the system, all the service resources during the entire supply chain (upstream process and downstream process) should be considered. Usually the service resources distribute into several companies or enterprises.

2. Model of Service Resources for PSS Development

In PSS, Service resources include application resources, component resources, network resources and equipment resources and so on. The elements of these resources can construct the complex service by the basic application service units and the integration theory. And the service resources must be described by proper approaches to be better validity and efficient. Because the resources come from different service suppliers, the integration of resources is dynamic and every unit can be integrated into a body temporarily according the resource layout and topology. However, the mechanism of organization and the interior structure of units are different in PSS, and the single modeling method from one hierarchy is hard to describe the whole service resource. So a multi-hierarchy integration model of resource system and new methods are needed to establish a computation model with unified structure for this kind of complex system.

The model of service resource is an important standard to judge the development ability of service supplier, and it will be a part of the process on developing the complication system. There is an inheritable and associative relationship among service resource, sub-missions and gross mission. So the information model of service resource has three layers such as the mission layer, the unit layer and the resource layer as shown as Figure 1.

In mission layer, system will select the optimum service member according to a certain principle. The mission assignment agent decomposes the gross mission into several independent sub-missions. The root mission can be presented as a series of sub-mission nodes around the gross mission. A series of service unit nodes are formed in the process of distributed business, and they can be composed into an entity which can complete a certain development mission of system.

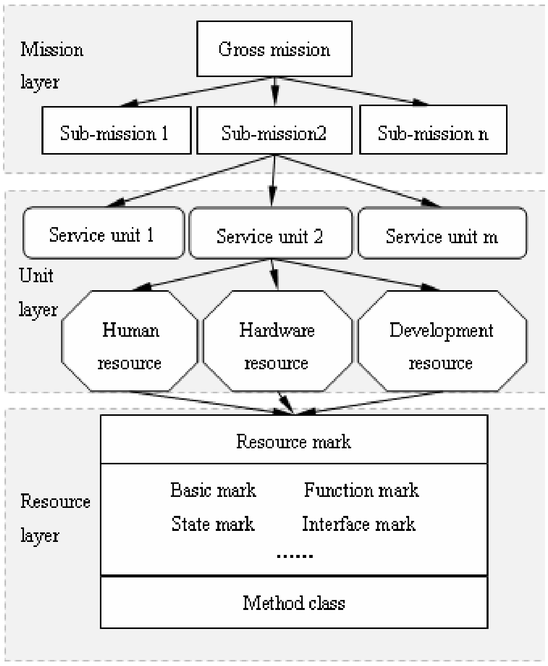

Figure 1. Three Layers Model of Service Resource for PSS Development

In unit layer, the service members with different functions have the same basic development resources such as human resource, hardware facility resource, technology resource and so on. Especially the development and management are the most important resource. In this layer, the analysis of resource model is the object-oriented model of technology resource. Overall, the resources on unit layer are the abstract of development ability, and make the produce resource integrate with the enterprise unit tightly.

In resource layer, the property and the operation method can be described by object-oriented method. The object-oriented method is to abstract the information of complex entity as objects in reality. An object has some properties to describe the form, composition and character of the object. The property is an object. Also it has other objects as its properties. So the similar objects can be gathered into a class. An object has some methods to describe the action of object. The object's status can be changed by the methods of class. The description of object and the operation are encapsulated into a body and the class can communicate with outside only by message.

3. Description of Service Resource for PSS Development

In order to implement the communication among the different layers, the resource metadata must be constructed. Recently, the common method of constructing the resource metadata with different structure is based on XML, but XML is lack of the ability to describe the relationships of metadata and resource. So the resource description frame (RDF) is referred to solve this problem. RDF is the standard of metadata model. RDF uses a standard and general way to normalize the semantic data based on XML. It is an infrastructure that enables the encoding, exchange, and reuse of structured metadata. This infrastructure enables metadata interoperability through the design of mechanisms that support common conventions of semantics, syntax, and structure. RDF does not stipulate semantics for each resource description community, but rather provides the ability for these communities to define metadata elements as needed. RDF uses XML (Extensible Markup Language) as a common syntax for the exchange and processing of metadata. The XML syntax is a subset of the international text processing standard SGML

(Standard Generalized Markup Language) specifically intended for use on the Web. The XML syntax provides vendor independence, user extensibility, validation, human readability, and the ability to represent complex structures. By exploiting the features of XML, RDF imposes structure that provides for the unambiguous expression of semantics and, as such, enables consistent encoding, exchange, and machine-processing of standardized metadata [9-10].

RDF supports the use of conventions that will facilitate modular interoperability among separate metadata element sets. These conventions include standard mechanisms for representing semantics that are grounded in a simple, yet powerful, data model discussed below. RDF additionally provides a means for publishing both human-readable and machine process-able vocabularies. Vocabularies are the set of properties, or metadata elements, defined by resource description communities. The ability to standardize the declaration of vocabularies is anticipated to encourage the reuse and extension of semantics among disparate information communities. RDF is designed to support this type of semantic modularity by creating an infrastructure that supports the combination of distributed attribute registries. Thus, a central registry is not required. This permits communities to declare vocabularies which may be reused, extended and/or refined to address application or domain specific descriptive requirements.

The goals of RDF are broad, and the potential opportunities are enormous. This introduction to RDF begins by discussing the background context of the RDF initiative and relates it to other metadata activities.

RDF data model is similar with the entity-relation graph. But RDF data model have not a mechanism to explain the properties and relation definition between the properties and other resources. So the mission of RDF is referred to solve the problem. To define the metadata expediently, RDF will have a class system like the object-orient program and model system. A class set is called a pattern. A class can be organized into a hierarchy. The extended performance is provided by improving the sub-class. Thus when a new pattern with little difference from the existed pattern is needed to construct, we just add and modify it according to existed pattern. RDF will support the reuse of metadata definition through sharing the mode.

An example of RDF is constructed as shown as Figure 2. The relationship among class, sub-class and resource is expressed. If a resource belongs to a class, there is a definition or connotative property of the resource is *rdf:type*, and its value is the definition of the resource including the class and it can be marked as *t*. If a class is the child of another class, there is an arc *rdfs:subclass* which is marked as *S*.

The correlation resource information can be described by the class layer on RDF mode and XML. For example, there is a service resource to query other Internet service. A part of codes is as shown as Figure 3.

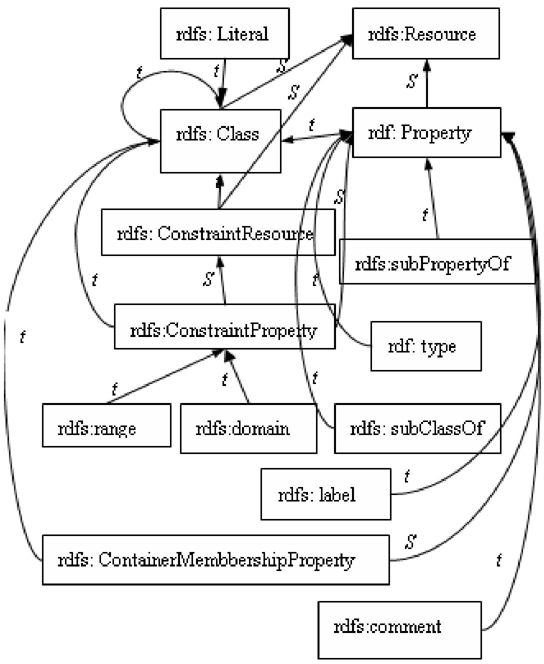

Figure 2. Description of RDF Model

```
<rdf: RDF xml:lang="en"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs= "http://www.w3.org/2000/01/rdf-shema#">
<rdfs:Class rdf:ID="SearchQuery">
<rdfs:subClassOf rdf:resource="http://
http://www.w3.org/2000/01/rdf-schema#Resource"/>
</rdfs:Class>
<rdfs:Class rdf:ID="SearchResult">
<rdfs:subClassOf rdf:resource="http://
http://www.w3.org/2000/01/rdf-schema#Resource"/>
</rdfs:Class>
...
<rdf:Property ID="queryService">
<rdfs:domain rdf:resource="#SearchQuery"/>
<rdfs:range rdf:resource="#SearchResult"/>
</rdf: Property>
...
</rdf:RDF>
```

Figure 3. Part Codes

Firstly the service resource is named as *SearchQuery* class. Search service is a sub-class of the internet service, and it has some result properties. These words express the relation between function and property through class fashion. RDF/XML-based metadata description resource has three main characteristics such as controllability, extended-ability and changeability. RDF adopts basic resource-property-value format and class layer mode, thus this solve the problem of too many resource metadata. Also XML makes the data be exchanged and extended easily.

4. Service Resources Base based on Ontology

The aforementioned service resource model and RDF-based resource metadata expression provide a method for description. A service resource base is needed to be constructed for searching and utilizing the resource rapidly. Today the relation database is used to store the metadata and content, but this method needs huge database structure, and needs to maintain these data. The whole work is very great. So according to the correlative theory and method of knowledge engineering and ontology, the method of knowledge-based service resource ontology is referred [11-13].

(1) Ontology

Originally, the term ontology has its roots in philosophy. As a discipline of philosophy, ontology denotes that the science of what is, of the kinds and structures of objects, properties events, processes, and relations in every area of reality. Starting in the late 1980s and early 1990s, ontology gained increasing awareness in Computer Science and Artificial Intelligence (AI). AI requires formal representations of real world phenomena in order to reason about these phenomena. In a literal sense, AI research borrowed the term ontology from philosophy and equipped it with a computational meaning. As a result, AI coined the term formal ontology or computational ontology. Ontology is a formal, explicit specification of a shared conceptualization of a domain of interest. Conceptualization depicts an abstract representation of some real-world phenomenon by having determined its relevant concepts, relationships, axioms, and constraints. Further, explicit denotes the explicit definition of the type of concepts, relationships, axioms, and the constraints holding on their use. Formal indicates that the ontology should be readable and interpretable by machines, thus formal excludes the use of natural language. Finally, shared conceptualization requires the ontology to capture consensual knowledge that is not private to an individual person but accepted by a larger group of individuals.

(2) Ontology model and RDF

Ontology model provides the common and sharing knowledge about one problem domain, and this kind of knowledge can communicate and understand with the distributed and isomer computers. Ontology model has two main characters. First is knowledge description. The extension and content of description based on knowledge are extended, so more exact understanding can be formed. Second is common expression rule and structure. Rule and structure are the communication precondition of ontology model in different systems. Also they are the success key to make the abundance description content fused into the strict rule framework.

In the given domain, ontology forms the core of the system expressed by any knowledge. Ontology assures the consistency, accuracy, invariability, reuse and sharing of knowledge understanding and exerting. RDF/XML is the most suited mode to describe the resource knowledge.

(3) Ontology of service resource

OIL is chosen to be the standard of constructing the ontology model of the service resource after studying on *AL-Log*, *CARIN*, *CLASSIC*, *DLR*, *F-Logic*, *Loom*, *OIL* and *OKBC*. *OIL* is referred as a kind of ontology model language by the researchers of Amsterdam University of Holland and Manchester University of Britain. It is based on *XOL* namely *OKBC-lite* knowledge model based on *XML*. *OIL* includes ontology container, class, slot, restriction of slot, value of slot and so on.

The facet method of application service taxonomy has the following class hierarchy in software component as shown as Figure 4.

The structure of Ontology has six domains. They are component domain, implementation description domain, information domain, environment domain, style domain and edition domain. Every domain has many slots. The former four domains describe the content of a component, and the later two domains describe the relationship among components.

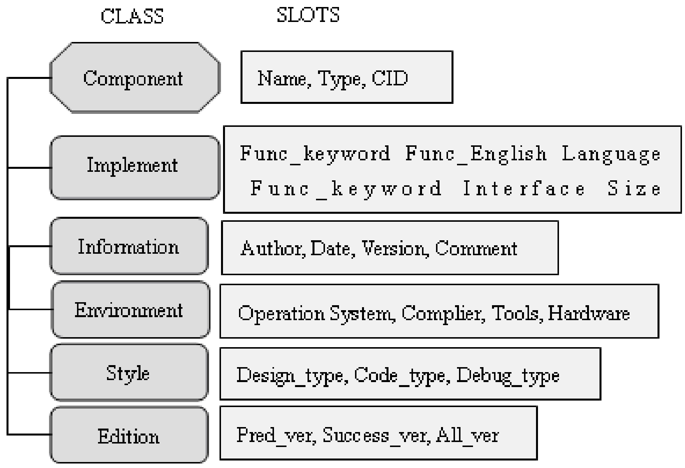

Figure 4. Class Hierarchy Structure of Ontology

(3) Ontology construction of component resource

According to ontology, the ontology-base oriented component resource is developed in Protégé environment. Protégé is based on Java, is extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development. The original goal of Protégé was to reduce the knowledge-acquisition bottleneck by minimizing the role of the knowledge engineer in constructing knowledge bases.

In order to do this, the knowledge-acquisition proceeds in well-defined stages and that knowledge acquired in one stage could be used to generate and customize knowledge-acquisition tools for subsequent stages. Thus, the original version of the Protégé software was an application that took advantage of structured information to simplify the knowledge-acquisition process. It formalized the ontology that constrained its knowledge bases and knowledge-acquisition tools. Protégé was designed to leverage the *CLIPS* expert-system shell and its knowledge representation system, and it took an important first step by providing a precise language for its ontology, based on standard frame language concepts of classes, instances, slots, and facets. Multiple inheritances among classes were supported [14]. The class and slot structure of resource base are designed as shown as Figure 5.

When constructing the class and slot structure of component resource, commonality and agility of this structure must be considered in order to describe various components easily. Implementation class is the key class, because it includes some important information such as function of component, interface and so on. Interface class is designed to be a slot of Implementation, so that it can contain complex interface information. After class and slot is designed, the component of this structure can be derived as shown as Figure 6.

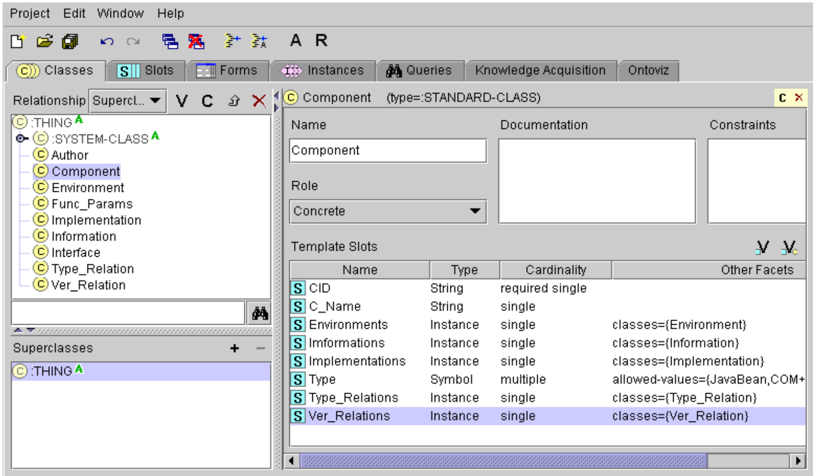

Figure 5. Structure Based-ontology of Component Resources in Protégé Environment

Because ontology -based construction method can express the relation of complex objects as class hierarchy, we can construct all resource ontology bases in service alliance, such as mission resource, sub-mission description, component resource, code resource, application resource and so on. And all bases information is based on RDF/XML.

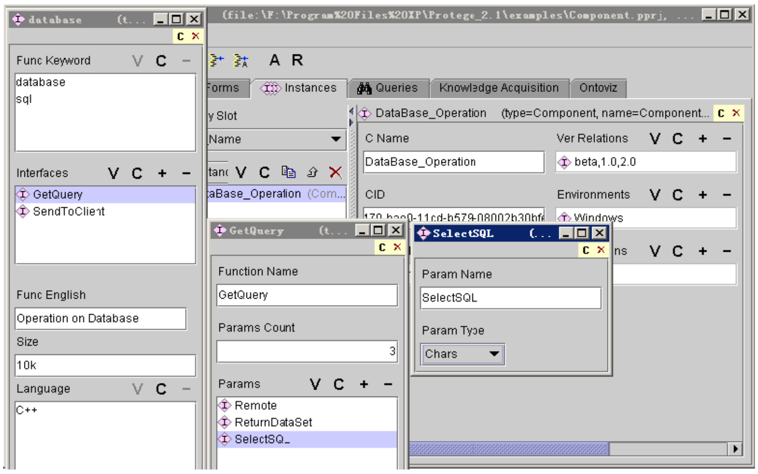

Figure 6. Component Resources Ontology in Protégé Environment

After constructing the resource ontology base, it is very convenient to search all needed resource through resource property. And by developing the Java API, the integration of service resource system in protégé environment can be realized. Finally the upper mentioned method is used to develop the prototype product and service system for one brand bicycle's company. The company can choose and optimize the consumer, modular design product, establish product structure, build and maintain modular product ontology, and develop the product collaborated with the suppliers, etc as shown as Figure 7.

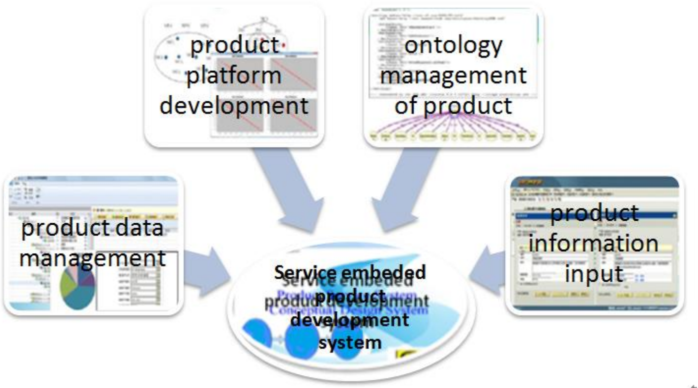

Figure 7. The Prototype based on the Ontology

5. Conclusion

The properties of service resource in PSS development and the metadata construction based on RDF are analyzed in this paper. The ontology model language and the relate methods are applied. The ontology base model oriented component soft resource is developed in protégé environment. And it can be conclude that the resource based on ontology has three kinds of characteristics. First one is commonality. The constructed class is the abstract properties of resource, the defined class and its hierarchy-relation can derive all resource with the same class. And they are described in RDF/XML. The second is flexibility. Because the class hierarchy based-ontology and the property of slot are scalable, the resource example based on class can scale its properties, and the property of resource is the compound property of another resource. The last is integration. The resource can be retrieved by metadata, thus the resource can be localized when system resource is reconfigured, and the share and integration of resource can be fulfilled by the relation interfaces.

Acknowledgements

This paper is supported by the National Natural Science Foundation, China (Grant No.51305417).

References

[1] N. Nishino, S. Wang and N. Tsuji, "Categorization and mechanism of platform-type product-service systems in manufacturing", J. Manufacturing Technology, vol. 1, no. 61, **(2012)**, pp. 391-394.
[2] L. Hao, Q. Guoning, J. Yangjian and G. Xinjian, "Service oriented product modular design method and its prospects", J. China Mechanical Engineering, vol. 12, no. 24, **(2013)**, pp. 1687-1694.
[3] Y. J. Geum and Y. Park, "Designing the sustainable product- service integration: a product-service blue print approach", J.EI. Journal of Cleaner Production, vol. 5, no. 19, **(2011)**, pp. 1601-1614.
[4] J. Jie, L. Yan and X. Yan, "Product service systems innovative design based on TRIZ final ideal solutions and function stimulation", J. Computer integrated manufacturing systems, vol. 2, no. 19, **(2013)**, pp. 225-234.
[5] C. Xuesong, "Design-driven innovation mechanisms", J. Journal of Industrial Engineering and Engineering Management, vol. 4, no. 25, **(2011)**, pp. 191-196.
[6] C. Durugbo, A. Tiwari and J. R. Alcock, "A review of information flow diagrammatic models for product-service systems", J. Advance manufacturing technology, vol. 9, no. 52, **(2011)**, pp. 1193-1208.
[7] T. C. Kuo, "Mass customization and personalization software development: a case study eco-design product service system", J. Intelligent Manufacturing, vol. 5, no. 24, **(2013)**, pp. 1019-1031.
[8] T. Sakao and M. Lindahl, "A value based evaluation method for Product/Service system using design information", J. Manufacturing Technology, vol. 1, no. 61, **(2012)**, pp. 51-54.
[9] "An introduction to the resource description framework", http://www.dlib.org/.
[10] A. Scheuermann and J. Leukel, "Supply chain management ontology from an ontology engineering perspective", J. Computers in industry, vol. 6, no. 65, **(2014)**, pp. 913-923.

[11] R. Studer, R. Benjamins and D. Fensel, "Knowledge engineering: principles and methods", J. Data & Knowledge Engineering, vol. 2, no. 25, **(1998)**, pp. 161–197.
[12] Z. Fei, "Research on theory and method of manufacture collaborative service network", D. Zhejiang University, **(2007)**.
[13] Z. Fei and Z. Jian, "Research on database access technology based on Spring and Hibernate", J. Computer engineering and design, vol. 9, no. 30, **(2009)**, pp. 1668-1670.
[14] "Protégé Project", http://protege.stanford.edu/.

Author

Zhang Fei, she is a lecture in the institute of Mechanical Engineering, China Jiliang University, China. She received her Ph.D. in Mechanical Engineering from Zhejiang University, China. Her research interests are in the fields of efficiency computation of system, technology of mobile agent, product and service modular and dynamic modeling and simulation.

