Measurement Of Correlation For Integration Of Dynamic Semantic Portlets In Portals

Khaine Swe Wynn
Faculty of Information and Communication Technology
University of Technology (Yatanarpon Cyber City), Myanmar

Win Htay
Computer Univeristy, Thahtone, Myanmar

Abstract

Portal is a means of integrating information through a portlet, which is a presentation component. Nowadays, there is not yet a final and decisive answer to integrate the semantic portlets dynamically. Semantic portlets are integrated through ontology-based interoperation which is about the semantic structures in the background. To get the desired portal page, the semantic portlets are selected and bounded dynamically with one another according to user's needs. In integrating the semantic portlets, the paper proposes to measure cohesion and coupling approach among dynamic semantic portlets that how strongly related the semantic portlets in portal page are and which degree the unrelated portlets are forced to relate among them. The paper also proposes to implement the portal page which manages semantic portlets according to the measurement of correlation.

1. Introduction

A portal is a web based application that commonly provides personalization, single sign on, content aggregation from different sources and hosts. Portlets are the kernel technology of the portal model and Web components with Web contents and applications functions of enterprises. Portlets can get access to different information sources and applications.

Portlets are applications within a portal in much the same way as servlets are applications within a Web Server. They are similar with windows applications in a user desktop in the sense that a portlet renders markup fragments that are surrounded by a decoration containing controls. The content generated by a portlet is called a fragment. A fragment is a piece of markup such as HTML, XHTML and so on adhering to certain rules and can be aggregated with other fragments to form a complete document. A single portal web page can syndicate portlets' fragments.

The portal page then contains a number of portlets whose fragments can be arranged into columns and rows, and minimized, maximized or arranged to suit the user needs. . Fragments from different portlets are arranged in the same portal page. This allows the content from different services being visually accumulated. Information contained in one portlet will surely be required in another, and forcing the individual user to manually copy and key in data from source to target portlets leads to frustration, lost productivity, and inevitable mistakes. When the dynamic portlets are aggregated, they are arranged with the data flow without measuring the correlation of the contents of the flow. Portlets' interoperations involve a portlet influencing the control of another portlet.

To solve this problem, the paper proposes the use of ontology based annotation for the input markup and the measurement of cohesion and coupling approach among dynamic semantic portlets that how strongly related the semantic portlets in portal page are and which degree the unrelated portlets are forced to relate among them. The contents of the semantic portlets are aggregated by measuring the data flow cohesion and coupling in correlation. Once all the markups are annotated along the ontology and the data flow among portlet interoperations are measured with cohesion and coupling, semantic integration can be achieved by mapping data.

2. Related works

To interoperate the porltets in the same portal page, many approaches have been implemented in the past.

In [5], the paper focuses on using ontology to support portlets semantic interoperability based on new features of JSR268. The proposed approach maps the portlets to concepts in ontology, and then constructs semantic data associations between such

concepts, and finally achieves portlet interoperation based on the data associations.

Diaz et al. proposed an annotation-based portlet interoperation approach [8] that supports semantic data transfer. In that system, portlets are characterized by their ontology. Portlets are extended with additional markup about the background services and deep annotation strives to interact with these underlying services rather than with HTML surface that conveys the markup. Portlet interoperation is achieved through the mapping of the ontology concepts.

The paper [4] focuses an HTML fragments based approach to achieve interoperability. In that paper, a presentation component which is named as ShadowComponent is constructed for each portlet involved in a portlet interoperation using its fragments. Then data flow process between components is defined using ECA rules. Finally, the paper proposed to drive a process by creating events to fulfill data flow between ShadowComponents. That process enables the portlet operation as the fragments of a portlet are synchronized with their corresponding component.

This paper implements the ontology based annotation scheme for integration of dynamic semantic portlets and applies the cohesion and coupling measurement for integration of portlet pages.

3. Motivation

Portlets are applications within a portal in much the same way as servlets are applications within a Web server. The difference stems from portlets being multi-step, user-facing applications. Portlet renders markup fragments that are surrounded by a decoration containing controls. The portal page contains a number of portlets whose fragments can be arranged into columns and rows, and minimized, maximized or arranged to suit the user needs.

Aggregating portlets into a portal is more than merely invoking these services or arranging their fragments together in the same portal page. Information in one portlet will be required in another protlet and forcing the individual user to manually copy. When another portlets are dynamically integrated, the portlet aggregation can't decide whether the portlets are coupled to another portlet or not. This situation hinders the correlation among the semantic portlets in dynamically integrating the semantic portelts.

To overcome these drawbacks, this paper presents ontology based annotation approach to portlet interoperation and correlation measurement among the dynamic semantic portlets in mapping the source portlets to target portlets.

4. Background Theory

4.1 Portlet Concepts

Portlets are reusable Web modules that run on a portal server and provide access to Web-based content, applications, and other resources. Companies can create their own portlets or select portlets from a catalog of third-party portlets. Portlets are intended to be assembled into a larger portal page, with multiple instances of the same portlet displaying different data for each user.

From a user's perspective, a portlet is a window on a portal site that provides a specific service or information, for example, a calendar or news feed. From an application development perspective, portlets are pluggable Web modules that are designed to run inside a portlet container of a portal server.

The portlet container provides a run time environment in which portlets are instantiated, used, and finally destroyed. Portlets rely on the portal infrastructure to access user profile information, participate in window and action events, communicate with other portlets, access remote content, lookup credentials, and to store persistent data. The Portlet API provides standard interfaces for these functions. The portlet container is not a stand-alone container like the servlet container. Instead, it is implemented as a thin layer on top of the servlet container and reuses the functionality provided by the servlet container. WebSphere Portal supports two different portlet APIs by providing two portlet containers:

The Java Portlet Specification API. This is based on javax.portlet interfaces. WebSphere Portal supports the Java Portlet Specifications 1.0 and 2.0, also known as JSR168 and JSR286.The IBM Portlet API. This is based on org.apache.jetspeed.portlet interfaces.

Each portlet on the page is responsible for providing its output in the form of markup fragments to be integrated into the portal page. The portal is responsible for providing the markup surrounding each portlet. In HTML, for example, the portal can provide markup that gives each portlet a title bar with minimize, maximize, help, and edit icons.

Portlets rest on two main standardization efforts: JSR 168 and WSRP (Web Services for Remote Portlets). JSR168 is a Java Community Process that standardizes an API for implementing local, WSRP-compatible portlets. WSRP standardizes the interfaces of the Web services a portlet producer must implement to allow another application (typically a portal) to consume its portlets. The next section presents the portlet interoperation.

4.2 Portlet Interoperation

Portlet aggregation is the combination of a set of portlets to achieve a common goal. The

aggregation provides some value to the user since portlets are simultaneously rendered in the same portal page. Portlet interoperation can be classified as application-based, datasource-based and annotation-based.

The application-based approach introduces the portlet application that allows distinct portlets to share a common piece of information to achieve portlet interoperation. However, a portal normally frames portlets from distinct portlet applications, which prevents the data from being exchanged.

The datasource-based approach focuses on the use of cuton JSP tag library or XML descriptors to enable a portlet to be a data source. The target portlet is defined in a WSDL file with a custom extension to describe the actions, which can consume data transferred from other portlets. However, the description based approach may cause compatibility problem.

An annotation based portlet interoperation approach supports semantic data transfer. Portlets are characterized by their ontology in which portlet fragments extend extend their markups with information about the supported process. Portlet interoperation is achieved through the mapping of the ontology concepts. Ontology-based interoperation can facilitate the declarative specification of the mapping between the realms of two distinct portlet providers rather than the mapping being hidden in the portlet code.

Based on these approaches, portlet interoperation is applied in many scenarios but there are limitations of the sharing scope and the standardization effort required. This paper proposes to apply annotation based portlet annotation scheme.

4.3 Ontology Concepts

An ontology[9] is a formal representation of a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to define the domain. In theory, an ontology is a "formal, explicit specification of a shared conceptualizat ion". An ontology provides a shared vocabulary, which can be used to model a domain — that is, the type of objects and/or concepts that exist, and their properties and relations.

There are domain ontology and task ontology in portlet annotation. A domain ontology (or domain-specific ontology) models a specific domain, or part of the world. It represents the particular meanings of terms as they apply to that domain. So ontologies are able to provide semantic information for portlets.First each portlet can map to a concept in domain ontology and register its semantic informat ion. And then, according to the registered semantic information, data will flow from one portlet to anther automatically.

The task ontology reflects the process as a collection of input and output OWL-S atomic process [9]. Although it has not been implemented yet, the basic ontology can now be extended to specify the order in which processes proceeds or the relationships between their parameters. As the domain classes come from distinct ontologies, the portal must solve first potential mismatches and ontology mappings between the different portlet ontologies. Mapping becomes necessary as distinct communities can have their own terms and regulations. Ontology mapping is an important issue whose implications are outside the scope of this paper. However, ontology mapping is vital to achieve portlet interoperation.

4.4 Correlation among semantic portlets

Definition 1:

Cohesion refers to the degree to which the elements of a module belong together.It is a measure of how strongly-related each piece of functionality. The degree of cohesion among portlets is d (dmin ≤ d ≤ d max).

$$Cohesion(X) = \frac{FD}{TD} \quad (1)$$

where FD is the number of flow data from one portlet to another ,
TD is the total number of data which occurs in both portlets.

$$Cohesion(B \to C) = \frac{FD\{(BD \to CD) \in (BD \cap CD)\}}{TD\{BD \cup CD\}} \quad (2)$$

where BD is the number of data in portlet B,
CD is the number of data in portlet C.

Definition 2:

Coupling is degree of dependence among the data of portlets.

$$Coupling(B \to p_i \in P(i = 1,2...,n)) = \frac{DD(BD \cap \{p_1 \cup p_2 .. \cup p_n\})}{TD} \quad (3)$$

where DD is the dependent data among portlets,
TD is the total number of data for portlet B.
Pi is the number of data in portlet i.

5. Proposed System Architecture

The proposed system will be developed by using ontology based annotation scheme to integrate the semantic portlets and by measuring the semantic portlet interoperation by using cohesion and coupling approach to be reliable and robust. For target portal pages with combination of

portlets, the proposed system will implement for defining the ontologies for portlets and the portal.

The system will produce a set of instantiations related to the portal's ontology and referring to the fragment markups of a source portlet. The proposed system will feed the markup of a target portlet from annotation kept by the portal. From source portlet to target portlet, the mapping will be generated by the system and the fact which portlets are coupled to another are measured using cohesion and coupling methods. Finally, the portal page is described by integrating the semantic portlets according to the user needs. In the client side, the system will implement the degree of relation among the semantic portlets to be reliable, efficient and robust.

For example, in Figure 2, the client portal page may dynamically change the presentation layout according to the user needs. For user A, the poral page is aggregated with three fragments (portlets) and for user B, the page is presented with two fragments. The required portlets are syndicated and registered in the portal but according to the user needs and the measurement of the portlet aggregation based on the mapping, the required portal page will dynamically aggregated and presented to the portal user.

To achieve the main purpose of the system, cohesion and coupling approach will be applied to measure the dynamic semantic portlets interoperation.

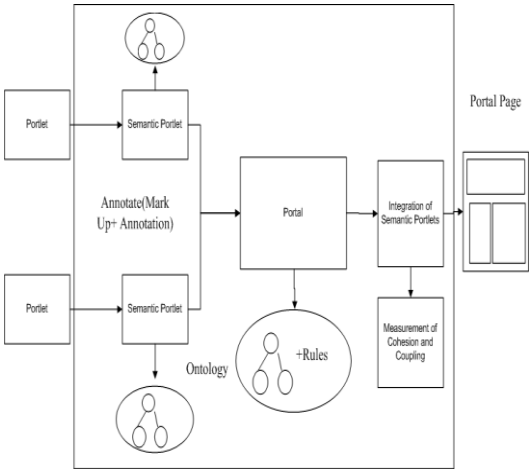

Figure 1. Overview of dynamic semantic portlet interoperation

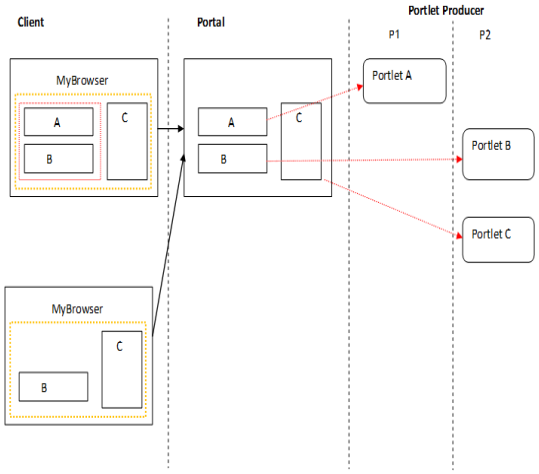

Figure 2. Portlet Producer and Portlet Consumer

6. Conclusion

The paper focuses the integration of semantic portlets in the portal page. The proposed system will use ontology based annotation scheme to integrate the background structures of portlet based on the cohesion and coupling which is generated viewing the mapping rules among the portlets. The proposed system will provide the efficient presentation layer for the portal page by integrating the front end of semantic portlets.

7. References

[1] Java Community Process. JSR 168 port let specification. 2003.http://www.jcp.org/en/jsr/detail?id=168

[2] Java Community Process. JSR 268 port let specification. 2006.http://www.jcp.org/en/jsr/detail?id=268

[3] Welcome t o prot égé. http://prot ege.st anford.edu

[4] Jingyu Song, Jun Wei, Shuchao Wan, " An HTML Fragments Based Approach for Portlet Interoperability".

[5] Liu Xianming, Li Weihua, Li Shixian, "Using Ontology to Support Portlet Semantic Interoperability", *First International Workshop on Education Technology and Computer Science*, 2009.

[6] Liu Xianming, Li Weihua, Wang Chen, "Research on the Portlet Semantic Interoperability Architecture". *World Congress On Software Engineering*, 2009. WCSE'09, pp. 453-456.

[7] Ma Aránzazu Irastorza Goñi, "Designing of Portlet-basedWeb Portals,"

[8] O. Diaz, J. Iturrioz, and A. Irastorza, "Improving portlet interoperability through deep annotation", in Proc. The 14th Int. I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[9] Zhou Qingyun, Li Shixian, "An approach to support portlet semantic interoperation by using ontology based on JSR268", Second International Conference on Information Technology and Computer Science, 2010.