International Journal of Hybrid Information Technology
Vol.8, No.4 (2015), pp. 243-250
http://dx.doi.org/10.14257/ijhit.2015.8.4.26

Research on Intelligent Retrieval System Control Base on Semantic Component: A Novel Approach

Fengling Yin and Bingquan Huo

Binzhou Polytechnic, Shandong, China
yinfengling111@126.com

Abstract

How to manage the component library is important to the component reuse. The description mechanism of components is the basic on the component retrieval, which make users better understand the component. In general, we deal with three issues in semantic search, namely, usability, scalability and retrieval performance. In this paper, we propose a novel keyword-based semantic retrieval approach. Furthermore, the tradition component library exist heterogeneity because of the different company have their own standard. So the proposed methodology in the paper is fulfilled with the component management framework in the basic of the ontological feature model based on ontology and multiple agent system .The feature model is distinguish the tradition feature model, which describe the component form three different dimensions. Also the semantic retrieval model of component and the disturbed calling of component based on the multiple agents are researched which will help to improve the component reuse. The result indicates the effectiveness of the proposed framework.

Keywords: *Intelligent Retrieval, System Control, Semantic Component*

1. Introduction

With the development of software industry, the software development process becomes more and more complex. The life cycle of software become shorter and software is updated more frequently. During the development of software, the user's demand and technologies will changed as time go on, which drive to update the production. Component reuse can improve the efficiency of software development. While the development based on component face an urgent question. The components developed by different mechanism will be heterogeneity; the description framework is different. So how to manage and use the increasingly complex components and how to communications among the component must to be solved. The current research work on the component and some standard components are mainly focus on the description of domain feature model and semantic usefulness, which is main factor of domain component reuse, especially some characteristics of the business logic component. So the component mainly focuses on the underlying function component. The description of components is also the basic of component retrieval which make user able to understand components. In [1-2], author proposed semantic component based on domain feature, depict the component from three sides: domain space, definition space and context. This method point out the relationship among business domain, definition and context, while it show a little broadly and the operation is not so strong, not describe the attribute of the component. Before furtherly research, we first introduce the basic definition of information retrieval system. Information retrieval (IR) is one of the oldest research areas in the information science. The goal of the IR discipline is to search and retrieve the most relevant documents to the information needs of the user. Therefore a good IR system should retrieve only those documents that satisfy the user needs, not a bunch of unnecessary data. The general IR process can be divided into the following components:

(1) user interface: Although it seems insignificant at first, the user interface is one of the most important aspects in IR. The design of the user interface brings the tradeoff between user-friendliness and performance. Complex and powerful interface, provide more detailed and accurate query formula, and they are tedious and time consuming to the end customers. (2) query processor: The raw query submitted by the user should be processed before searching. Usually, the query is transformed into an internal form that the system can interpret. Furthermore, several processing tasks can be involved such as stop-word elimination, stemming and other application specific tasks. These tasks should also be done in the indexing phase for consistent matching. (3) indexing: The index is an important part of the infrared system, for two reasons. First, it optimized storage conditions and improved the response time of query performance greatly inverted file structure. Basically, it stores the text of the location of every word. Second, a large number of processing tasks in the index phase are similar to the query processing stage, further improve the performance. (4) searching: In this phase, the query terms are searched against the inverted index. All the documents that contain the occurrences of the query terms are retrieved. Depending on the application, the retrieval can be done even for the partially matched documents. (5) Ranking: Retrieval of the document in the previous step gives score according to the matching between the query conditions and document quality. The document classification is based on the score, so the most relevant documents submitted to the user to retrieve the top of the list. Ranking process is highly dependent on the IR model. In the following sections, we will explain some infrared model does not support the rankings, all the retrieved document is considered to be equally important. Finally, the choice of evaluation metric is critical, since only a few of them can utilize the ranking information for the evaluation score. The Figure 1 is the basic flowchart of the IR system.

According to the domain component, the descriptions of component service are not accurately, cannot show kinds of detail features. Components can be considered as realization of specific domain, including context requires and semantic alterability caused by uncertainty environment. So this paper proposes that a component management framework based on the ontological feature model and multiple agents system. The paper is organized as follows: Section 2 presents the related work. Section 3 describes the feature model based on ontology. Section 4 describes the management framework including multiple agent system. Finally, Section 5 draws some conclusions and points to future work.

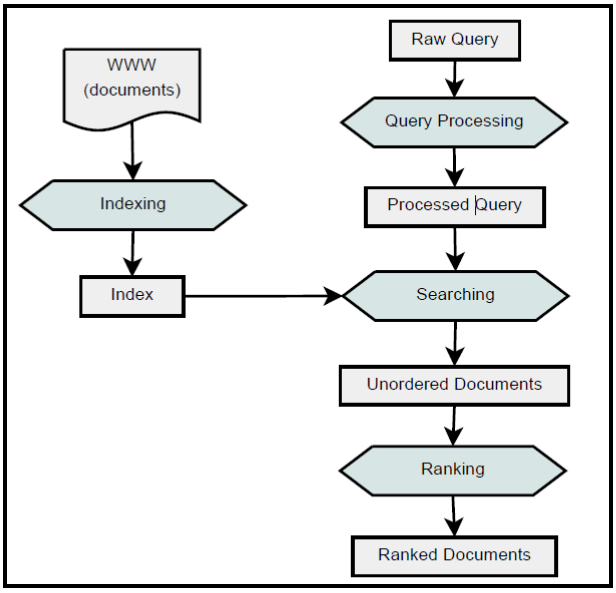

Figure 1. The General Information Retrieval Process

2. The Related Work

The research on domain engineer and product process has developed, feature modeling become a main method for domain analysis. The Component management based on feature model attracts many researchers recently. A famous development method based on feature descripted in [1], it extend the feature modeling [3, 4] to design phase. Feature modeling include the function message and non-function message of components, which describe the component overall to make user complication the component. While the ontology is as a formal specification as shared concept, that is mean that ontology is shared by people from different domain, which make the understanding from different people is the same, so the people or machine can more convenient to complication. So we choose the ontology to describe the feature modeling. In order to share the knowledge, the choice of ontology language is also important. The OWL is also being an international general standard. There are divided into three types according to the capability of the description and reasoning. OWL Full have the largest description capability, but can't assure the computing capability. OWL DL is based on the description logic; it can support the larger description capability without losing the computing capability. While OWL Lite is limited to the classification of the layers and the simple constrain, which have the weakest description capability. So OWL Full is chose to describe the feature modeling of components. Because of the characteristics of ontology, it will improve the effectiveness and accuracy of the semantic retrieval of components. Because of the adaptive, collaborative and reflection of the intelligent agent system, which can solve the communication of the distribute system. The theory and technology of intelligent agents has been studied by many projects and get widely application [5, 6, 7]. Because of the fundamental properties of an intelligent agent including autonomy, social ability, reactivity, and pro-activeness, agents could help meet the growing need for more functional, flexible, and personal computing and telecommunications systems. The agent techniques have been applied in many distributed computing fields [8], such as data mining [9], agent-based middleware [10] and information retrieval with distribute data sources [11], etc. These works brings much valuable reference to our research on distributed component repositories.

3. The Ontology-based Feature Model

3.1. The Overview of the Model

The features are the first-order entity in the demand space, which give expression to some capability or characteristic of the system. In the demand model based on the ontology, these features are expressed as concept, and divide into business action, facet and term. Business action is the main semantics part of action, which express the function of the component. Facet precisely describes the detail attributes. The Term is the value of the action attribute. In order to express the Number and Boolean value, the type of the Number and Boolean can be used to describe the action attribute. Along with the operation feature, the business object and the relationship between the businesses are also included into feature model. These features are the main part of feature model.

According to the different stages of the development processes, the component can be divided into four classes: requirement analysis component, design component, code component or function component, test component. In order to describe these four component, the description of the feature model based on the ontology are categorized into physical context message, domain message, quality message .Firstly, the physical context mainly describe the physical context message of component, for example: component name, size, type and the application domain of component. Second, Domain message mainly describe the relationship between actions, for example: business action, facets and term, which are widely used in domain feature model and defined in the paragraph 1 of chapter. At last, quality message mainly describe the performance of the component, for example: advantage and disadvantage of design component and so on.

This feature model is different to traditional feature model. The traditional feature model mainly describes the domain component and is short of the description of non-functional properties, while it is very important to help user complicate the component deeply. When there are two components that have the same function, but have different performance, in this case, the user will make a choice by the non-function properties. Also the description framework can description all kinds of components because it can describe the component from different aspects. According the above, the feature model based on the ontology as follow:

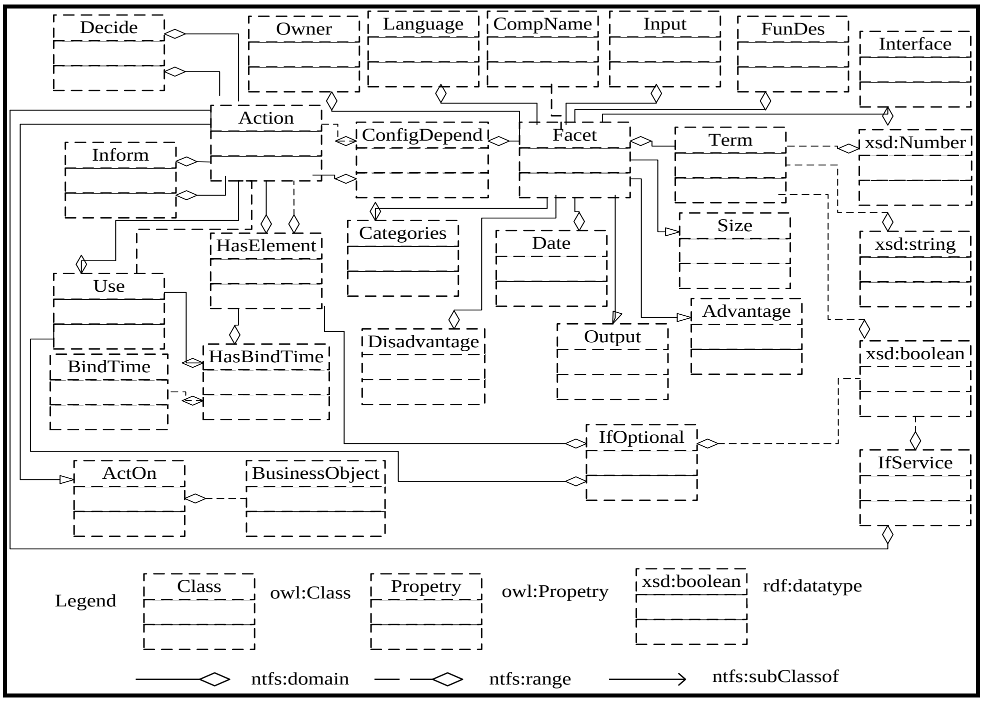

Figure 2. The Feature Model based on Ontology

3.2. The Physical Context Message

(1) Component Name: The name of component, which make user can identify the component.
(2) Language: Which language implements the component? The language is same to the developer language; the component can be used much easier.
(3) Owner: The developer of component, if possible.
(4) Date: The date of component published. We can know whether the component is the latest by the date.
(5) Categories: The categories of component. The user can retrieve the component by the categories, which will be more convenient.
(6) Input: The input Parameter of component.
(7) Output: The output of the component.
(8) Interface: The defined interface of component.
(9) Size: The physical size of component.

3.3. The Domain Message

(1) FuncDes: It describe the function semantic of component.
(2) HasElement: It describe the Decomposition of business action, father action may be related to children action.
(3) Use: It describe when you call the function calling may depend on others.
(4) IfOptional: It is the Boolean attribute value defined on the Has-Element or Use relationship, which is optional.
(5) IfServer: It is used to identify two kinds of action, if it is true, the action is services action.
(6) Inform: After calling one action will transferred the message to another action, but not depend on that function.
(7) ConfigDepend: It describe the Configuration dependencies between the component
(8) subClassof: It define the relationship between action. This relationship can be transferred to subclass relationship (rdfs:subClassOf)

3.4. The Quality Message

(1) Disadvantage: The disadvantage of component.
(2) Advantage: The advantage of component.
(3) From the figure, the feature model not only describe the function characters, but also describe non-functional properties, which includes the physical context message, quality of component, which can describe the component from different dimension, help users understand the component better. So this model can improve the accurately of retrieval, which is very important in component management system.

4. The Design of Management Framework for Component Library

The research on the component library is mainly force on the description of the component and component retrieval, is short of management mechanism, in order to improve the effective of component reuse. In this charter, we provide a management framework of component library, which will decide the extent of component reuse, which also include the semantic retrieval model of component and multiple agent system.

4.1. The Framework of Component Management

Component library is a system which mainly includes three modules, 1) Component, it deposits components. 2) The ontological feature model of component, through which users can complicate the component. 3)The management system of component, which control the component library, mainly include the adding、deleting、browsing, modifying and searching component, it control the whole component library, by these operations, can keep the component effective and improve the component reuse. The Organizational structure of component library is shown in the Figure 3. About the management system of component is help to improve the accurate component. The management of component mainly includes component adding, component deleting, and component updating, component searching. Component searching: When a user wants to reuse the component, so he will search the component in the library whether it will meet the requirement. And it will respond the feature model to the user; will make the user understand the component's function. Component adding: If you want to add a new component to the component library, firstly, the system will search whether it have the same component by matching the feature model in the library, if not, it will add successfully to library, if so, you can't be added to component library. Component deleting: If you have an older component won't be useful, so you need to delete it, but you also have administrator permission. Component updating: If you have a better component compare to the older component, so you need to delete older one and add the new one, but you also have administrator permission.

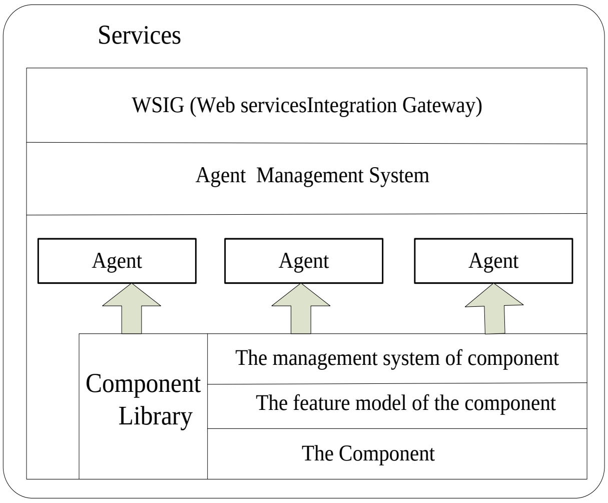
Services
WSIG (Web servicesIntegration Gateway)
Agent Management System
Agent
Agent
Agent
Component Library
The management system of component
The feature model of the component
The Component

Figure 3. The Architecture of Component Library

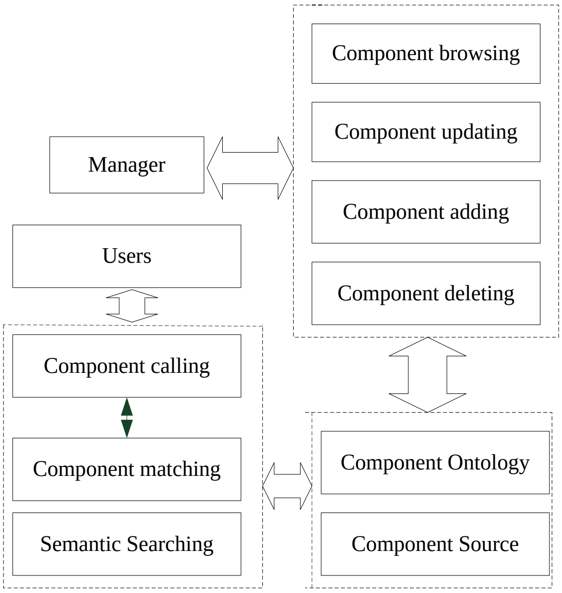

Figure 4. The Management Framework

4.2. The Intelligent Retrieval Model Based On Ontology

According to the above, the component retrieval is very important in the component library. The quality of searching will decide whether the component can be reused. The tradition retrieval model can't deal with the natural language requires of users, but user more want to search the component by natural language to increase the user experience, So we need a UI to let user input the requires, and change requires to the user required description model based on ontology, which will match with the ontological feature model of component in the component library. The Intelligent retrieval model based ontology as the following:

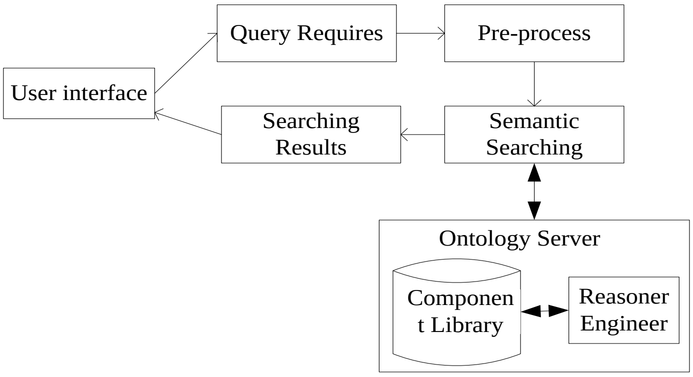

Figure 5. The Intelligent Semantic Retrieval Framework

4.3. The Disturbed Calling of Component

The traditional component libraries mainly focus on one machine, only support centralized and static calling. With the development of component library, the company in different location will develop distinguish components. In order to reuse the component better, we need to call the reuse the remote component. In order to management the remote component, we propose the multiple agent component libraries. The multiple agent component libraries follow the FIPA (The Foundation for Intelligent Physical Agents) standard. It includes three kinds of agents: 1) Agent Management System: It control to access to agent and use access points of agent, only exist a AMS on one platform, provide the life cycle services, maintain Agent ID and agent state, every agent must registration onto the AMS, in order to get the effective AID, no matter the state of agent, AMS must be active. 2) WSIG（Web Service Integration Gateway): It mainly improves effective of component reuse, this agent work as web services. To make the function agent published to traditional services, we need a new agent under the FIPA standard, which name as WSIG agent. It works as web services gateway, which mainly charge of registration of agent on the UDDI. It will map the agent action to WSDL file, which can be searched by the outer 3) F-Agent (Function agent):It is component agent. If the components are published to web service, will support distributed calling and browsing. It make user reuse the component more conveniently.

5. Conclusion and Summary

In order improve reuse to the component, we mainly make research from three aspect against the management of component library: 1) make the ontological feature model, describe the component from different aspects, provide a basic for semantic retrieval for components; 2) provide the management model of component library: The current research on the component library mainly is short of management model, To manage the component in the library, it is important to provide the management model of component library. 3) The disturbed calling of component: The traditional component library is centralized and static calling, not be convenient to calling the remote component.so we support provide multiple agent model, components are as services, make convenient to communicate with remote components.

References

[1] K C Kang, S Kim, J Lee, K Kim, E Shin and M. Huh, “FORM: A feature-oriented reuse method with domain-specific reference architectures”, Annals of Software Engineering, vol. 5, (**1998**), pp. 143−168.

[2] J. Yu and G. Y. Qing, “Domain feature space based semantic representation of component [J]”, J ournal of Software, vol. 13, no. 2, (**2002**) pp. 311 - 316.

[3] KC Kang, SG Cohen, JA Hess, WE Novak and AS Peterson, "Feature-Oriented domain analysis (FODA) feasibility study", Technical Report, CMU/SEI-90-TR-21. Pittsburgh: Software Engineering Institute, Carnegie Mellon University, (**1990**), pp. 1−52.

[4] X. Peng, W. Zhao, Y. Xue and Y. Wu, "Ontology based feature modeling and application oriented tailoring [A], The 9th International Conference on Software Reuse [C], Berlin Heidelberg: Springer 2 Verlag, (**2006**), pp. 87 - 100.

[5] O. Etzioni, N. Lesh and R. Segal, "Building softbots for UNIX," in Software Agents -- Papers from the 1994 Spring Symposium (Technical Report SS–94–03), AAAI Press, pp. 9–16.

[6] M. Roesler and D. Hawkins, "Intelligent agents"",Online, vol. 18, no. 4, (**1994**), pp. 18-32.

[7] L. Rosen, "MIT Media Lab presents the interface agents symposium: "Intelligent agents in your computer?", Information Today, vol. 10, no. 3, (**1993**), p. 10.

[8] D. L. Martin, A. J. Cheyer, and D. B. Moran, "The open agent architecture: a framework for building distributed software systems", Applied Artificial Intelligence, vol. 13, nos. 1-2, (**1999**), pp. 91-128.

[9] M. Klusch, S. Lodi, and G. Moro, "Agent-based distributed data mining: The kdec scheme", in AgentLink, vol. 2586 of LNCS, Springer, (**2003**).

[10] S. Lipperts and A. S. Park, "An agent-based middleware – a solution for terminal and user mobility," Computer Networks, vol. 31, no. 19, (**1999**), pp. 2053-2062.

[11] N.G. Shaw, A. Mian and S. B. Yadav, "A comprehensive agent-based architecture for intelligent information retrieval in a distributed heterogeneous environment", Decision Support Systems, vol. 32, no. 4, (**2002**), pp. 401-415.