Business Component Identification of Enterprise Information System：

A hierarchical clustering method

Meng Fan-Chao, Zhan Den-Chen, Xu Xiao-Fei
School of Computer Science and Technology
Harbin Institute of Technology, Harbin 150001, China
Mengfanchao74@163.com

Abstract

Business Component Identification is recognized as one of the greatest important phase in the process of Component-Based Software Development (CBSD). This paper presents an approach to identify business components from domain business model. First a domain business model of enterprise information system is proposed, and it is the input of business components identification phase. To identify business components, a hierarchical clustering technique based on graph is proposed. This method proposed differs from traditional clustering technology, and it uses the edge strength to substitute for edge weight. In the process of business components identification, we consider cohesion, coupling, granularity and number of business components. To acquire high quality business components, we give the rule of evaluation of business components, which consider mainly non-functional factors such as capacity, cost, security etc. Finally, quality management subsystem is used as example to describe the method of business component identification.

1. Introduction

Component-Based Software Development (CBSD) is a key technology to tackling reconfiguration of enterprise information system. With the rapid development and maturation of the standard of component model such as CORBA, DCOM and EJB, The Component-Based Software Development has been widely applied to development of enterprise information system [1, 2]. CBSD is different from traditional methodology of software development, it mainly emphasis on acquiring reusable components, and these components acquired are assembled to implement the functions of application system. In CBSD the process of building enterprise information system is consisting of two stages: business component Fabrication and application system assembly [3]. The business component Fabrication stage of CBSD consists of various phases: Domain Analysis and Modeling, Business Component Identification, Business Component Design and Business Component Implementation. The goals of Domain Analysis and Modeling phases are to develop Domain Business Models. The goal of Business Component Identification phases is to identify business components from Domain Business Model so that can be reused across domains. In this paper, we mainly focus on the Business Component Identification.

Component Identification is recognized as one of the greatest difficulties in the process of component Fabrication. Currently the main methods of Component Identification have clustering analysis [3, 4], matrix analysis [5, 6] and graph-based decomposing methods [7, 8, 9]. Those methods mainly discuss building OO components from UML, and they focus on much more

Proceedings of the 2005 IEEE International Conference on e-Business Engineering (ICEBE'05)
0-7695-2430-3/05 $20.00 © 2005 IEEE

technical-oriented representation of domain through classes and objects. However business components are more coarse-grained and are intended to provide a high-level business-oriented representation of domain. In this research, we mainly study on identifying business components from domain business model, whose elements have corresponding semantics. In this research, a hierarchical clustering technique based on graph using edge strength metric is proposed, which blends the advantages of clustering analysis and graph-based decomposing methods and focus on acquiring component from domain business model.

The remainder of this paper is organized as follows. First domain business model of enterprise information system are proposed in section 2. The method of business component identification for enterprise information systems is proposed in section 3. A case study demonstrating the approach is shown in section 4 and conclusion in section 5.

2. Domain business model

The goals of domain analysis phase are to understand the problem domain and built domain business model that includes function view, information view, process view and organization view [10]. Because organization view is unrelated to Component Identification, therefore we do not take into account it here. The primary model elements and various links between them are shown as Figure 1.

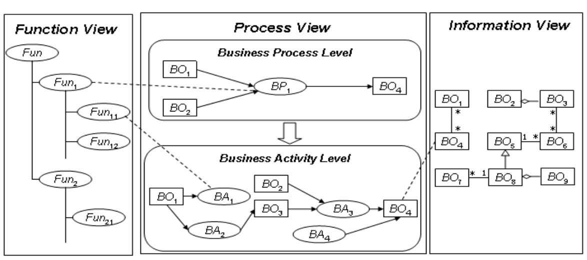


Figure1. Domain Business Model

Function view mainly expresses the partitions of business functions and the structures among functions. Functions can be defined as operating on business objects to satisfy enterprise goals. Function can be expressed at different levels of set [10]. In general, there are four levels of set: function bundle, function, sub function and basic function.

Process view mainly expresses behavior features of system. Business process can be represented at two levels of abstraction: business process level and business activity level (shown as Figure 1). At business activity level, a business process is a specific ordering of business activities across time and place, with a beginning, an end, and clearly identified inputs and outputs. These ordered business activities affect the states of business objects by creating, consuming and changing their contents.

Information view describes business objects and the relationships between them. Business objects represent entities in the business that participate in and are affected by business activities and business process. he relationships between business objects can be classified as static relationship and dynamic relationship. The static relationship represents the way various business objects are related to each other, and the dynamic relationship represents the communication of information and the transition of states between business objects.

Traditionally, a software component is defined as a self-contained piece of software with well-defined interface or set of interfaces [11]. A larger-grained component called a business component focuses on a business concept as the software implement of an autonomous business concept or business process. Business components vary from traditional software artifacts such as code segment, class and object etc. Traditional software artifacts are mostly fine-grained and technical-oriented of the domain. Business components, on the other hand, are more coarse-grained and provide a high-level business-oriented representation of the domain, and they express future components and the relations of those components.

3. Business Component Identification

The approach proposed for identifying business component uses business model as input. In business model, the main elements have business activities and business objects that are used to identify business components. This yields two sets of business components that we called business object components and business process components. Figure 2 expresses the process of business component identification. Business object component, which contains business objects and relations of those them, are basic components in enterprise information system. Business process component, which is composed of business object components and dependencies of them, are composite components. In this paper, we mainly focus on the approach of business object component identification

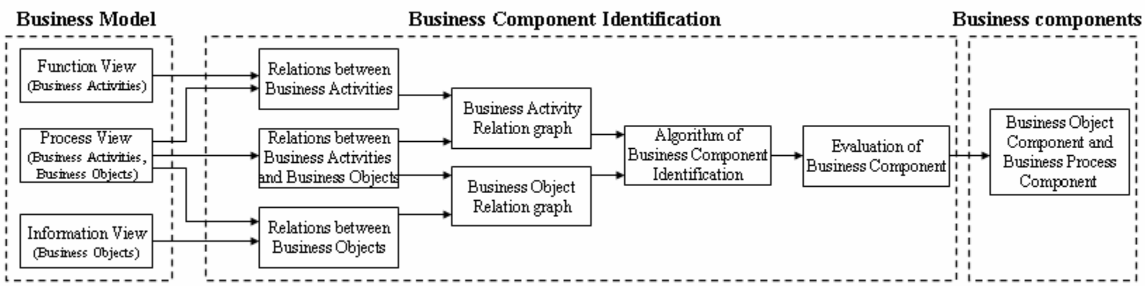


Figure2. Process of business component identification

3.1 Resemblance Degree

To identify business object component, we use the concept of resemblance degree between business objects. A resemblance degree for a given pair of business objects indicates the degree of similarity between these two business objects, depending on the relationships between them. The relationships between business objects can be classified as static relationship and dynamic relationship. Static relationships include: general, compound, aggregation and association, and dynamic relationship represent the communication of information and the transition of states between business objects through business activities.

Table 1.Object-activity relation matrix

object \ activity	BA_1	BA_2	BA_3	BA_4	BA_5
BO_1	C		U		U
BO_2	U	C	U		C
BO_3	U			U	

We use object-activity relation matrix (M_{OA}) shown in table 1 to calculate the dynamic resemblance degree between business objects. In M_{OA}, row represents business object and column represents business activity. The relationships between business object and business activity can be mainly classified as create (C) and use (U).

For example, in table 1, business activity BA_1 creates business object BO_1, and uses business object BO_2 and BO_3. Here we regard business activities as attributes of business objects, values of which can be represented as C and U, and N represents empty value. For example BO_1 possesses attribute BA_1, BA_2 and BA_5, and their values are respective C、U and U. Therefore there are six possible dynamic relationships: C-C、C-U (U-C)、U-U、C-N(N-C)、U-N(N-U)、N-N between two business objects. Assume their weights are respective w_1, w_2, w_3, w_4, w_5 and w_6, and their number are respective s_1, s_2, s_3, s_4, s_5 and s_6. Several calculating formulas of dynamic resemblance degree between business objects are shown as:

$$(1) RD_1(BO_1, BO_2) = \frac{w_1 \cdot s_1 + w_2 \cdot s_2 + w_3 \cdot s_3}{(w_1 \cdot s_1 + w_2 \cdot s_2 + w_3 \cdot s_3) + w_4 \cdot s_4 + w_5 \cdot s_5}$$

$$(w_1 + w_2 + w_3 + w_4 + w_5 = 1)$$

$$(2) RD_1(BO_1, BO_2) = \frac{(w_1 \cdot s_1 + w_2 \cdot s_2 + w_3 \cdot s_3) + w_6 \cdot s_6}{(w_1 \cdot s_1 + w_2 \cdot s_2 + w_3 \cdot s_3) + w_4 \cdot s_4 + w_5 \cdot s_5 + w_6 \cdot s_6}$$

0-7695-2430-3/05 $20.00 © 2005 IEEE

$$(w_1+w_2+w_3+w_4+w_5+w_6=1)$$

(3) $$RD_1(BO_1, BO_2) = \frac{2(w_1 \cdot s_1 + w_2 \cdot s_2 + w_3 \cdot s_3)}{2(w_1 \cdot s_1 + w_2 \cdot s_2 + w_3 \cdot s_3) + w_4 \cdot s_4 + w_5 \cdot s_5}$$

$$(w_1+w_2+w_3+w_4+w_5=1)$$

If we regard C and U as equivalence, and let $w_1= w_2= w_3=w_4=w_5=1/5$ in formula (1) and (3), and $w_1= w_2=w_3=w_4=w_5=w_6=1/6$ in formula (2), so formula (1), (2) and (3) are respective Jaccard Coefficient, Russel and Rao Coefficient and Sorenson Coefficient [12].

The Static relationships between business objects can be classified as general (g), compound (c), aggregation (a) and association (as). Assume their weights are respective $w_g, w_c, w_{a}, {}_{as}(0\leq w_g, w_c, w_a, w_{as}\leq 1)$. The static resemblance degree between business objects BO_1 and BO_2, which is represented RD_2 (BO_1, BO_2), equals to sum of weights of static relationships between them.

The resemblance degree between business object BO_1 and business object BO_2 is defined as: $RD(BO_1,BO_2)=\alpha \bullet RD_2(BO_1,BO_2)+\beta \bullet RD_2(BO_1, BO_2)$ $(\alpha+\beta=1)$. Where α and β represent respective weights of static relationship and weight of dynamic relationship.

3.2 Hierarchical clustering technique based on business object relation graph

We use business object relation graph to represent business objects and their relationships. Business object relation graph is defined as a graph $G = (V, E)$, where V is set of vertexes, and each vertex represents a business object. $E=\{(v, u, r) \mid v, u\in V; r\subseteq \{g, c, a, as, d\} \wedge r\neq \Phi\}$ is set of edges, $\forall (v, u, r)\in E$, assume $w(v, u, r)$ is weight of edge (v, u, r), henceforth $w(v, u, r)= RD(v, u)$.

The problem of business object component identification can be transferred into problem of graph decomposition. Here we propose a hierarchical clustering based on business object relation graph. We use edge strength metric to measure the strength of edges of business object relation graph. Let $e = (v, u)\in E$ be an edge of graph $G=(V, E)$ and v, u be its endpoint. Define the strength of edge e:

$$S(e) = \frac{2w(e)}{2w(e) + \sum_{e'\in BE(e)} w(e')}$$

Where $E(v)$ is set of edges linked with v, $E(u)$ is set of edges linked with u, and $BE(e)=(E(u)\cup E(v))-(E(u)\cap E(v))$. As an example to calculate the strength of Figure 3(a), which contains six vertexes and seven edges, according to strength formula:

$$S(v_1, v_2) = 2w_{12}/(2w_{12} + (w_{14} + w_{16} + w_{23} + w_{25} + w_{26}))$$

$$S(v_1, v_4) = 2w_{14}/(2w_{14} + (w_{12} + w_{16}))$$

The strength of edge defined considers both cohesion and coupling between business objects, namely independence of component. Cohesion refers to the degree of similarity between business objects in a business component. On the other hand coupling refers to the extent to which business objects within the business component relate to other business objects, which are not in that business component. The principles of cohesion and coupling are very important and essential factors for well-defined component design. The properties of an ideal business component are low coupling and high cohesion.

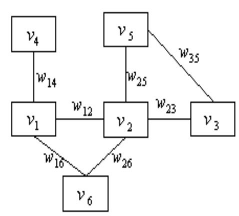


Figure 3(a). A business object relation graph

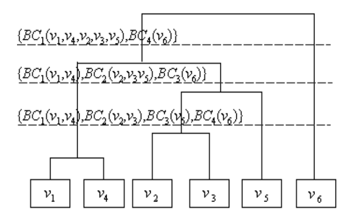


Figure 3(b). A dendrogram

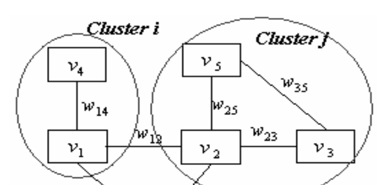


Figure 3(c). A cluster result

Figure 3 Process of Components Identification based on graph

To identify business object components, we propose a hierarchical clustering technique based on business object relation graph using suggested edge strength metric. A hierarchical clustering technique requires inputting the edge strengths for all possible pairs of business objects to be clustered, and produces dendrograms as the output. The process of business component identification based on G starts by placing each vertex in a cluster by its own, giving origin to $|V|$ clusters. Clusters are iteratively defined by combining at each the iteration the pair of cluster i and cluster j having the greatest edge strength in G. At each iteration, G is properly reconstruct by combining cluster i and j to a new cluster l. when combining cluster i and j to a new cluster l, assume a cluster k are linked with both cluster i and cluster j, and the weight of cluster j and cluster k is $w(j,k)$, and the weight of cluster i and cluster k is $w(i,k)$. The weight $w(l,k)$ of cluster l and cluster k is defined as following [12]:

1) Single linkage: $w(l,k) = \min\{w(j,k), w(i,k)\}$

2) Complete linkage: $w(l,k) = \max\{w(j,k), w(i,k)\}$

3) Weighted average linkage: $w(l,k) = (w(j,k), w(i,k))/2$

4) Unweighted average linkage: $w(l,k) = (w(j,k)\,|v_j| + w(i,k)\,|v_i|)/(|v_j| + |v_i|)$

Where $|v_j|$ and $|v_i|$ are respective number of vertexes in cluster j and cluster i.

The clustering process can be constrained by defining granularity of business component and the number of business components in system. Granularity of business component that represents the size or length of business logic realized by the business component is an important factor effecting on quality of business components. Granularity of business component is quantified the sum of business elements realized by the business component. There are different ways of granularity measure for various type of business component. For example, granularity of business object component is the sum of business objects. Business component is neither large nor small. If a business component is larger, it has more complex interface and possibility to be affected by change. However, if business components are small, the time and platform resources of inter-component communications are expensive. Therefore a balance is important to business component Identification. Granularity of business component depends on the component complexity in system. Designer can set minimum number of business components desired and the maximum number of business objects that are allowed in a business component. The number of business component refers to maximum number of business component in an enterprise information system. In the process of business component identification, designer may set the number of business component. If designer does not assign any constraint condition, the clustering process can terminates when the number of clusters left G in is one.

If designer uses different linkage technique or weight of edge, and there will produce different dendrograms as output. The designer can identify several sets candidate for the definition of reusable business components. Figure 3 describes a process of business components identification, Figure(a) is a business object relation graph, and Figure 3(b) is corresponding dendrogram produced form the business object relation graph shown as Figure 3(a). The dendrogram shows that there are three optional candidate sets of business components that include: $CS_1=\{BC_1(v_1,v_4),\ BC_2(v_2,v_3),\ BC_3(v_5),\ BC_4(v_6)\}$, $CS_2 = \{BC_1(v_1,v_4), BC_2(v_2,v_3,v_5), BC_3(v_6)\}$, $CS_3=\{BC_1(v_1, v_4, v_2, v_3, v_5), BC_3(v_6)\}$, and Figure 3(c) is a set selected from the candidate sets of business components.

3.3 Evaluation of business components

Well-defined component designs are driven by a variety of non-functional factors such as capacity, cost, security etc, besides functional factor. In this research, we provide guideline to well-defined set of business components among candidate sets.

Let $F = \{f_1, f_2, \ldots, f_n\}$ represent finite set of factors effecting on quality of business component, $w_j (j=1,2,\ldots,n)$ represents the weight of factor f_j, and $CS = \{CS_1, CS_2, \ldots, CS_m\}$ represents set of candidate set of business components identified, so the rule of evaluation of candidate set of business components can be defined:

$$ROE(CS_i) = \sum_{k=1}^{|CS_i|} \sum_{j=1}^{n} w_j v_j^k$$

Where $|CS_i|$ represents number of business components in set CS_i, and v_j^k represents the value of evaluation of business component BC_i on factor f_j. Rule of evaluation is classified into maximum rule and minimum rule, for minimum rule, it can be transferred into maximum rule, so we only think over maximum rule. According to maximum rule of evaluation, set CS_i which processes maximum value of ROE is selected as the best set of business components identified.

There are two type factors effecting on quality of business components: functional factors and non-functional factors. In the business component identification phase, we mainly consider the functional factors, and in the evaluation of business component phase, we mainly non-functional factors. There are many non-functional factors such as cost effectiveness, ease of assembly, customization, maintainability and capacity etc. The evaluation value of the factor relates to actual application system.

4 A case study

This case study uses the Quality management subsystem developed for describing business object components identification proposed. The quality management subsystem is divided into the following business process: Quality Information Management, Quality Checking Management, Quality Supervision Management and Quality Report Management. Each business process is then decomposed into business activities. Due to page limitation, here we only consider Quality Planning Management and Quality Checking Management. The main business objects in Quality subsystem include: object, Item, quality plan, organization, position, personnel, apply, sample and check report. Figure 4 represents the Object-activity relation matrix, and figure 5 represents static relationships of business objects.

To calculate static resemblance degrees between business objects, we set w_g=1, w_c=0.8, w_a=0.6, w_{as}=0.4. To calculate dynamic resemblance degrees between business objects, we use (3), and set $w_1 = w_2 = w_3 = w_4 = w_5 = 1/5$, $\alpha=\beta=1/2$. Figure 6 is business object relation graph.

If we assign the maximum number of business objects is five and the minimum number of business components is three, and use unweighted average linkage, according to the algorithm of hierarchical clustering proposed.

We can output a dendrogram shown as Figure 7, which can produce several sets of candidate business components. Figure 8 shows the selected set of business components that have been evaluated based on rule of evaluation.

BA / BO	Basic Information management					Quality Check management			
	Def BO	Edit QP	Def Org	Def Posit	Def Per	Reci Appl	Input sam	Input Re	Aud Rep
Object	C	U							
Item	C	U							
QP		C						U	
Org		U	C	U	U				
Position		U		C	U				
Person					C	U	U	U	U
Apply						C	U		
Sample							C	U	
Report								C	U

Figure 4 Object-activity relation matrix in quality subsystem

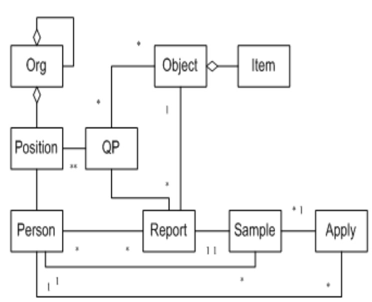
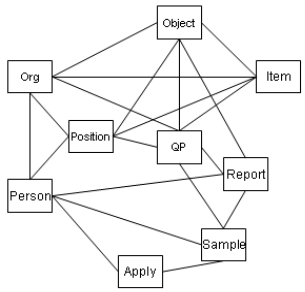


Figure 5 Static relationships of business objects in quality subsystem

0-7695-2430-3/05 $20.00 © 2005 IEEE

IEEE COMPUTER SOCIETY



	Object	Item	QP	Org	Position	Person	Apply	Sample	Report
Object		0.8	0.35	0.17	0.2				0.1
Item			0.25	0.17	0.2				
QP				0.17	0.3	0.14		0.35	0.35
Org					0.73	0.1			
Position						0.23			
Person							0.39	0.39	0.39
Apply								0.45	
Sample									0.45
Report									

Figure 6 business object relation graph

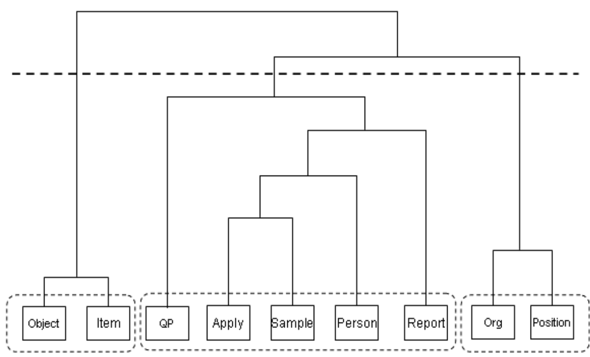


Figure7. Clustering results

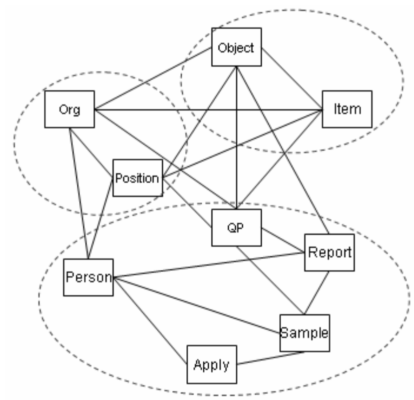


Figure8. Business components identified

5 Conclusion

In this study, we propose method of business component identification based on business model, which use hierarchical clustering technique and exploit edge strength based on clustering of graph. Different to traditional hierarchical clustering [3, 4], we use the edge strength to substitute for edge weight. In this approach, we not only take into account the cohesion of business component, but also the coupling between business components, namely independence of business component. Designer can identify several sets candidate set of business components through using different metrics. To acquire high quality business components, we give the rule of evaluation of business components. Now we are developing a tool of business component design based on business modeling.

References

[1] Somjit Arch-int, Dentcho N. Batanov. Development of industrial information systems on the Web using business components. Computer in Industry 2003,50(2):231~250.

[2] Df. D'Souza, A C Wills. Objects, Components and Frameworks with UML: the Catalysis Approach. Addison-Wesley, 1999.

[3] Jain H, Chalimeda N, Ivaturi N, Reddy B. Business component identification - A formal approach. In: Proceedings of the 5th IEEE International Enterprise Distributed Object Computing Conference. Seattle: IEEE Computer Society Press, 2001. 183~187.

[4] Lee JK, Jung SJ, Kim SD, Jang WH, Ham DH. Component identification method with coupling and cohesion. In: Proceedings of the 8th Asia-Pacific Software Engineering Conference. Macau: IEEE Computer Society Press, 2001. 79~88.

[5] Lee SD, Yang YJ. COMO: A UML-based component development methodology. In: Proceedings of the 6th Asia Pacific software Engineering conference. Takamatsu: IEEE Computer Society Press, 1998. 54~63.

[6] Ganesan R, Sengupta S. OSBC: A technique for the design of component-based applications. In: Proceedings of the 39th International Conference and Exhibition on Technology of Object-Oriented Language and Systems. IEEE computer Society Press, 2001, 46~55.

[7] Yves Chiricota, FabienJourdan, GuyMelancon. Software components capture using graph clustering. In Proceedings of the 11th IEEE International workshop on program comprehension(IWPC'03). IEEE Computer Society Press, 2003. 217~226.

[8] Brian. Mitchell. A heuristic approach to solving the software clustering problem. IEEE Proceedings of the 1999 International conference on software maintenance(ICSM'03).2003.

[9] S.Mancoridis, B.Mitchell, Y.Chen, E.R.Gansner. Bunch a clustering tool for the recovery and maintenance of software system structures. IEEE Proceedings of the 1999 International conference on software maintenance(ICSM'99).1999:50~62.

[10] August-wilhelm scheer. ARIS-Business Process Modeling. Springer-Verlag Berlin Heideberg 2000.

[11] W. Kozaczynski. Architecture framework for business components. 5th International Conference on Software Reuse, Computer Society Press. 1998, 300～307

[12]Chung-Horng Lung, Marzia Zaman, Amit Nandi. Applications of clustering techniques to software partitioning, recovery and restructuring. The Journal of Systems and Software 2004,73:227～244

0-7695-2430-3/05 $20.00 © 2005 IEEE

IEEE COMPUTER SOCIETY