MBPR：A Business Process Repository Supporting Multi-Granularity Process Model Retrieval

Jiangjun Zhu, Hailong Sun, Zicheng Huang, XuDong Liu
School of Computer Science and Engineering
Beihang University
Beijing 100191 China
{zhujj, sunhl, huangzc, liuxd }@act.buaa.edu.cn

***Abstract*—Business process repository aims at business process model discovery and reusing. However, most of current approaches for process matchmaking have the limitation that the process models should be in the same granularity, wherease a process repository with good support for multi-granularity business process retrieval is still rare. In this paper, we propose a similarity measurement mechanism which can effectively calculate the similarity between business process models in different granularity levels. A case study is used to demonstrate how modelers can search multi-granularity business process. At last, we conduct extensive experiments based on real dataset to study the performance of the repository.**

Keywords-process similarity;multi-granularity; repository.

I. INTRODUCTION

The capabilities to easily find useful business process become increasingly critical for business process repository [1,2], as more and more business process accumulated with the evolvement of enterprises' and organizations' business procedure. Companies document their daily routines in the form of business process models. Business process helps companies understand, communicate upon, or reengineer working procedure to enhance competitiveness. Modeling the business processes of an enterprise is an essential part of any IT development or implementation process. However, model design consumes a considerable amount of time and requires determining of activities to be performed, ordering of their execution, handling exception cases that might occur, etc. [3]. Benefiting from the already developed process models in the resource repository of company, reusing seems to be a promising approach to reduce the time consumed to develop new models [3].

Current repositories [1,2] for process retrieval are based on the matchmaking between business processes. Metrics are mostly limited to the matching of their syntactic [4], semantic [5], structural [6] and behavioral [7] information. As surveyed by paper [8], due to the large number and different granularity levels of processes, business process models are most commonly described by a hierarchy [9], [10], [11]. The models to be compared can have different granularity levels for achieving the same functionality. For example, one business process has a single activity to achieve certain functionality, while in another business process the same behavior is achieved by composing several activities. But current repositories rarely consider this situation as their retrieval scope is generally limited in processes with the same granularity, which makes their retrieval ability not flexible enough in this context. Thus, new mechanism is required to expand the retrieval capability of process repository to meet the multi-granularity requirement.

In this paper, we propose Multi-granularity Business Process Repository (MBPR) supporting similarity measurement of business processes with different granularities, and the idea in this paper is to decompose the coarse-granularity business processes into fine-grained before similarity calculation. The main contribution of this paper is a novel approach for measuring similarity between multi-granularity business process models which include: business process similarity calculation algorithm adapted from graph matching algorithm, coarse activity decomposition mechanisms. The decomposition mechanism includes three parts: (i) a control flow segment and markup language (SCMT), which can segment and mark the control flow information of annotated information contained in coarse activity, (ii) a series of decomposition rules, (iii) and corresponding algorithm based on SCMT.

In the next Section, we indentify the problem of multi-granularity retrieval and characterize the solution roadmap of MBPR. Section 3 presents existing approaches for business process retrieval and shows their drawbacks. In Section 4, we propose a business process similarity calculation method based on graph matching algorithm. In Section 5, we propose the decomposition mechanism which includes (i) granularity classification definition, control flow segment, (ii) markup language and the decomposition rules, (iii) and algorithm. In Section 6, we present the implementation of MBPR and a case study. In Section 7, several experiments are conducted to study the performance of MBPR. Finally, section 8 presents ongoing work and conclusion.

II. RELATED WORK

This paper mainly relates to research of similarity between business processes, based on which business process repository support process model retrieval.

The topic of retrieving business process based on model similarities has gained a lot of attention recently. The approach to measure the similarity between two process models has been addressed from different perspectives. The syntactic metric calculate an optimal matching between the activities in the process models by comparing their labels based on string edit distance [4]. To exploit semantic

 ISBN: 978-1-61208-105-2

features paper [5] have utilized wordNet synonyms. Structured data of business process have been taken into account, paper [6] discuss a structural match method based on classic graph matching algorithm, in which, business process are converted into directed graph, then edit distance between two directed graph is calculated to represent the similarity. Mendling, *et al*. [7] presented a preliminary discussion of influence of behavior in determining the business process similarity, it first convert business process models into causality graph footprint vectors, then compute the cosine value between two vectors to assess similarity. However, these approaches' context is that business processes to be compared are with same granularity, as we analyzed in Section 2, they are not flexible enough to handle the scenario in Fig. 1.

There is rare existing work on calculating similarity between processes on a degree of multi-granularity. Paper [18] proposed two new graph edit operations to take into account the difference of granularity levels, but it only consider the one-to-two relation. In this paper, the idea to solve this problem is decomposing the coarse activity process into fine-grained before the implementation of matchmaking. In Business Process Reengineering (BPR), refinement means explain exiting business process in more detail from multiple perspectives [19]. As shown in Fig. 2, the modeler use annotated information of activities to refine the coarse-granularity business process.

In summary, business process repository technology supporting retrieval based on business process matchmaking has been widespread concern, and has achieved some results. But most of the work focus on measuring similarity between business processes at same granularity level, there is lack of a multi-granularity matchmaking mechanism to expend retrieval capabilities.

III. OVERVIEW OF MBPR

In this section, we first present a scenario requiring multi-granularity similarity measurement. The scenario is situated in the context of developing of Enterprise-Specific Business Process Models. Then we present the multi-granularity retrieval procedure of MBPR.

Based on the prevalent granularity division, paper [8] further defined CBPM(Content-Based Process Modeling) which concentrate on developing of Enterprise-Specific Business Process Models with the help of existing business process, the procedure of this approaches can summarized as follows:

a) High level modeler (Enterprise analyzer or Manager) draw a business process at a coarse-granularity level to identify the general characteristics of the enterprise's business procedure.

b) Low level modeler (IT employees) need to refine the coarse-granularity business process into detailed fine-grained business process which comprehensive describe the enterprise's working procedure.

c) To reduce the developing time, low level modeler can retrieve those fine-grained business processes which are similar with the coarse-granularity one with help of repository search tool.

d) Fine tune the selected model to ensure that all relevant processes have been included, and unnecessary processes eliminated, and this is suggested business process caters the need of the implementing enterprise.

We can see the crucial step is c) and it implies a multi-granularity business retrieval scenario. Fig. 1 shows a overview of the multi-granularity retrieval scenario, take a coarse-granularity business process as an input, then measure similarity with all the business processes in repository, the most similar ones are recommended.

As shown in Fig. 1, the upper part is the input of coarse-granularity business process editor, including information of process structure, events, activities, etc. (Dotted P1 part). The lower part of Fig. 1 shows the available well-refined fine-grained business processes in repository.

However, the traditional retrieval approaches are not flexible enough for above scenario as they rarely take the granularity level information into account. A phenomenon is an coarse activity can map to a fragment of other business process according to their annotated information, as shown in Fig. 2, coarse activity "*hotel service*" can be decomposed into a BPMN [12] fragment which execute three tasks

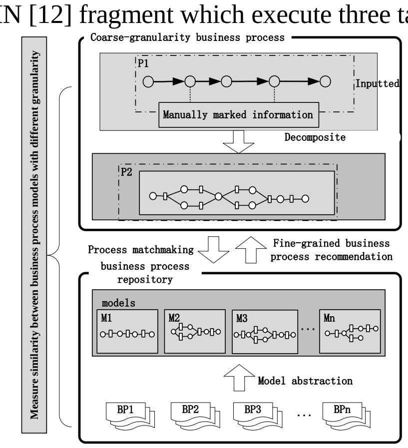

Figure 1. Scenarios of multi-granularity business process retrieval

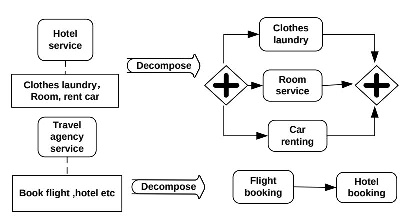

Figure 2. Example of coarse activities decomposition

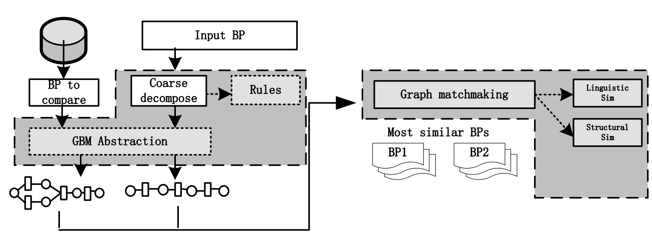

Figure 3. Multi-granularity retrieval procedure of MBPR

simultaneously include: "*clothes laundry*", "*room service*" and "*car renting*", but traditional similarity measuring approaches usually do not handle this situation, for processes which describe a same business logic in different granularity level, the similarity between them calculated by traditional approaches is low, so the retrieval result can not be satisfactory.

In order to effectively measure similarity between business processes with different granularity, we decompose the coarse activities of business process into fine-grained before similarity calculation. So in Fig. 1, before matchmaking, we add a step that executes the decomposition operation according to manually marked information (Dotted P2 part). The multi-granularity business process retrieval procedure of MBPR is shown in Fig. 3, business process models are abstracted to memory models and pushed into database server of MBPR, coarse-granularity process is decomposed before matchmaking, after similarity measurement, the most similar ones are recommended. Next, we will elaborate on the multi-granularity similarity measurement mechanism of MBPR.

IV. SIMILARITY MEASUREMENTS

In this section, we adapt an graph matchmaking algorithm to calculate the similarity between business process models modeled with BPMN.

A. *Process model abstraction*

Currently, BPMN has become one of the mainly used modeling languages during business process development. In this paper, we use BPMN as business process modeling language. To simplify the problem, we only discuss a core subset of BPMN. We refer to the definition of core BPMN proposed by [13], which is a subset of BPMN specification includes the key elements used to describe process control flow, core BPMN can meet most of the business process modeling needs.

As BPMI (Business Process Management Initiative) does not provide strict theoretical standard, BPMN so far still has no definite execution semantics, leading to that business processes developed by different BPMN modeling tools have large semantic difference in the structure and behavior, which lead to the complexity of refinement of BPMN

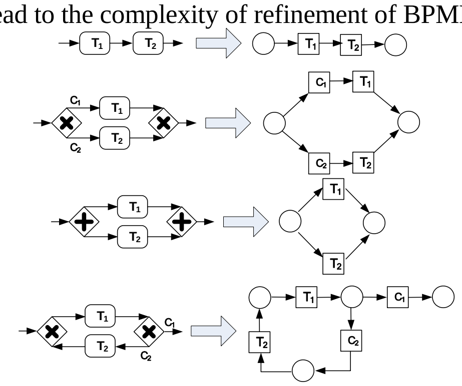

Figure 4. Four process and their GBM

process model. We refer to the notion of WCBP (Well-structured core BPMN process) proposed also by [13]. WCBP add some constraints based on Core BPMN: Firstly, two parallel flows initiated by a parallel fork gateway, should be joined by a parallel join gateway. Secondly, two alternative flows created via a decision gateway should be synchronized by a merge gateway.

First, we abstract a process model into a directed attributed graph, as following definition.

Definition 4.1 GBM(Graph based Business process Model)

Let Z be a set of labels. A GBM is a tuple (N,E,λ,Z), in which:

—N is the set of nodes;

—$E \subseteq N \times N$ is the set of edges;

—$\lambda: N \rightarrow Z$ is a function that maps nodes to labels

When abstracting a BPMN described model, we drop the types of nodes, Fig. 4 shows the GBM of a business process model. The left part shows the original process models. The right column shows the corresponding process graphs after abstraction. Each node is named with the node label, which originate from activity name or branch condition, gateways and events are abstracted into empty nodes

B. *Node similarity metric*

The similarity of nodes is determined by the similarity of node labels.

To exploit linguistic features we have utilized WordNet [14] as a background ontology, for each activity(node) in GBM we extract its name as main information [15] to measure its similarity with other activity, most activities' name are not formulated but most are presented as a phrase such as "*Credit Review*" or "*book ticket*", we can use a vector $\vec{v}$ < *verb* , *noun*> extract from a activities' name to describe the activity.

With the definition of $\vec{v}$, we improve the linguistic metric from [15]. The original linguistic function used by [15] is *L(n1,n2)*, it can calculate unorganized similarity value between two node' labels based on wordNet. Since all the nodes' label have been abstracted into $\vec{v}$ < *verb, noun*>, we propose function *sim*（$\vec{a}$, $\vec{b}$）to compute the similarity between two node, each label is donated by vector $\vec{a}$, $\vec{b}$. W is a semantic similarity matrix, W_{ij} represents the similarity value between the pair a_i and b_j, which can obtain using wordNet APIs.

$$sim(\vec{a},\vec{b}) = \frac{\vec{a} W \overrightarrow{b^T}}{|a||b|} \quad (1)$$

The two vectors may contains different number of words，word pair with the same syntax will be chosen to calculate similarity.

C. *Adapt graph matching algorithm for process similarity*

We use exiting graph matchmaking algorithm to measure similarity between GBMs. Paper [16] overview four main graph matching algorithms based on graph edit distance, but all of them are time consuming, to simplify the matching complexity, we use path index based algorithm *GraphGrep*, which is proposed by [17]. There are three basic steps of *GraphGrep*: (1) build the index to represent the database of graphs as sets of paths, this step is done by repository system previously, (2)filter the database based on the submitted query and the index to reduce the search space, and (3) perform exact match. The algorithm details can refer to the paper [17] .

Synthesize with GraphGrep and node similarity metric, let *G1 = (N1;E1; λ1)* and *G2 =(N2;E2; λ2)* be two GBM, the total Graph similarity are measured as follows:

$$SynSim(G_1,G_2)=\frac{\sum_{i=0}^{\|PH_1\|}\max\left\{Sim(P_i,P_j)\mid P_j\in PH(G_2)\right\}}{\|PH_1\|}, \quad (2)$$

$PH(G_2)$ donate set of paths of G_2

$$Sim(P_1,P_2)=\frac{GrapGrepSim(P_1,P_2)*\sum_{i=0}^{\|N_1\|}\max\left\{Sim(\vec{a}_i,\vec{a}_j)\mid a_j\in N(P_2)\right\}}{\|N_1\|},$$

$N(P_2)$ donate *nodes* of G_2

Equation (2) is a comprehensive similarity metric which takes accounts of the linguistic, structural information. As we discussed in Section 2, the granularity information must be taken into account when measuring the similarity, (2) do not fulfill this request, in the next section, a coarse activity decomposition mechanism will be proposed to handle this situation.

V. COARSE ACTIVITY DECOMPOSITION MECHANISM

In this section, we propose the decomposition mechanism for coarse-granularity business process.

A. Basic definition

For a coarse activity, its activity names may be a brief description of several tasks. To handle this situation, usual practice is attaching annotated information to the activity to explain the details. For example, As shown in Fig. 2, an activity "*Hotel service*" has annotated information which indicate that the single activity need to perform several tasks in a certain order, including: "*clothes laundry*", "*room service*", "*rent car*", etc. According to the annotated information, it can be decomposed into a fragment. As displayed in Section 2, the inputted coarse-granularity process is decomposed by MBPR before implementing matchmaking. In this paper, we take annotated information as basis to decompose coarse granularity activities.

Previous discussion prompts us to assess the granularity level of business process from the perspective of annotated information. Several basic definitions are given as follows:

Definition 5.1 coarse activity

An activity is coarse if and only if it has annotated information. We say an activity is decomposable if it is coarse.

Definition 5.2 fine-grained business process

A business process is fine-grained if and only if all its activities are not coarse.

Definition 5.3 coarse-granularity business process

A business process is coarse-granularity if it is not fine-grained. We say a business process is refinable if it is coarse-granularity.

B. Control flow markup Tags and decomposition rules

Based on the definition of granularity level, the idea of the decomposition is： Decompose all the coarse activities of coarse-granularity process until it become fine-grained. So the decomposition of a coarse activity is basic, which will replace the activity with a process fragment.

The procedure of the decomposition of a coarse activity can be divided into four steps: firstly, activity names are extracted from the annotated information as a set for new activities generation; secondly, the logic relations between new activities is analyzed, corresponding to WCBP as described in last section, our main concern is the Sequence, Switch, Loop and Parallel control flow relations; thirdly, new activities, arcs and gateways are created according to the activity names and relations; finally, the coarse activity is replaced with the generated fragment.

It is hard to extract the logic relations and activity names of annotated information attached to coarse activity by understanding natural language. To make it feasible, a small quantity of manually marks need to be added to annotated information, so a small markup language is introduced in the following segment.

Back to the Section 2, consider this situations, when a low-level modeler get a coarse-granularity business process, he mark and adjust the annotated information with some special tags complying with certain scheme, which MBPR can understand, then accuracy of the retrieval will be improved. To do this, reference to the traditional programming language design principles, we design the Control flow segment and markup Tags (SCMT). Using SCMT, a low-level modeler can mark the crucial logic relation and activity names in the annotated information to affiliate retrieval.

Existing business process modeling language usually supports the four basic control flow patterns: sequential (sequence), select (switch), concurrent (parallel) and loop, corresponding to the four relation presented by BPMN, we design four group of tags as shown in table 1: *&THEN* mark sequence relation, &*SIMU* mark parallel relation, *&IF* and &*ELSE* mark switch relation, &*WHILE* and &*REPEAT* mark Loop relation. In table 1, *A*,*B* and *C* can be a complete sentence or part of it. Through the segmentation tag, each of them can be divided into several activity names：{*A0, A1......An*}, {*B0,B1……Bm*}, {*C0,C1…… Ck*} .

The complete definition are defined by BNF in definition 5.4, we only concern the control flow representation. To improve the user experience, the compiler of MBPR does not have strictly syntax restrictions on the user marked annotated information.

Definition 5.4. **Control flow segment and markup Tags(SCMT)**

<*DecomposableInformation*>::=<*Clause*>{ <*Clause*> }
<*Clause*> ::= < *Sequence* >|< *Parallel* >|< *Switch* > |< *Loop* >
< *Sequence* > ::= <*Element*> ***&THEN*** <*Element*>
<*Parallel*>::=&***SIMU***<*Element*><*Element*>{<*Element*>}
<*Switch*>::=&***IF***<*Element*>&***ELSE***<*Element*>{&***ELSE*** <*Element*>}
<*Loop*>::=&***WHILE***<*Element*>|&***REPEAT*** <*Element*>
<*Element*>::=<*Segments*>|<*Clause*>{<*Segments*>|<*Clause*> }
<*Segments*>::=<*ActivityName*>{*******<*ActivityName*>}
<*ActivityName*>::=human readable phrase

Fig. 5 shows decomposition rules of these basic control flow patterns from SCMT to BPMN. BPMN use a number of tasks to describe the sequence of behavior, as shown in Fig. 5 (a), the switch pattern is decomposed to a BPMN fragment triggered by an exclusive gateway, determine the appropriate conditions and started a number of optional activities set, each of which corresponds to a specific chosen branch, as shown in Fig. 5 (b), the parallel pattern is decomposed to a BPMN fragment triggered by a an parallel gateway, start a number of concurrent collection of activities, shown in Fig. 5 (c), Similar to the mapping of switch pattern, the loop pattern is decomposed to a BPMN fragment which contains two optional branches, as shown in Fig. 5 (d).

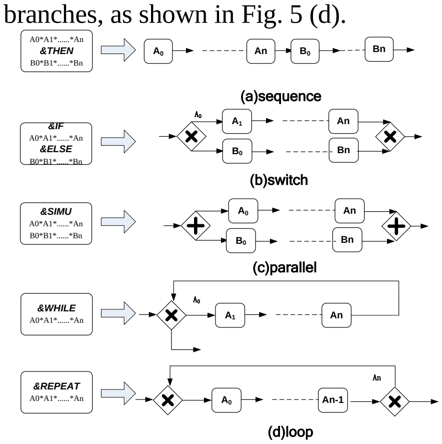

Figure 5. Decomposition rules between SCMT patterns to BPMN fragments

TABLE I. BASIC MARKS OF SCMT

Relation	Mark	Pattern
Sequence	&***THEN***	A &***THEN*** B
Parallel	&***SIMU***	&***SIMU*** A B C
		&***SIMU*** A B
Switch	&***IF*** &***ELSE***	&***IF*** A &***ELSE*** B &***ELSE*** C
		&***IF*** A &***ELSE*** B
Loop	&***WHILE***	&***WHILE*** A
	&***REPEAT***	&***REPEAT*** A
	∗(Segmentation tag)	***A0∗A1∗......∗An***

C. *Decomposition algorithm*

According to the decomposition mechanism, we design an coarse activity decomposition algorithm, which can understand the annotated information marked by SCMT. As shown in algorithm 1, the SCMT marked information is compiled and the coarse activity is decomposed into business process fragment.

Described as algorithm 1, activity names are extracted from SCMT marked information according to segmentation tag, when encounter the key elements marks the logic relations, choose appropriate function to handle it with the assistant stack and generated the correspondent BPMN fragment.

Algorithm 1 decomposition of coarse-granularity activity

```
Input： coarse-granularity BPMN business process
Output: fine-grained BPMN business process
begin
A：= {a| activity set};
A*：={a| a∈A & a has annotated information};
    stack =Φ; //Stack of activities
for each a∈A* {
    In(a);//activities attached to a and prior to a
    Out(a);//activities attached to a and succeed to a
    S=a.getAnnotatedinformation()；
    for each e∈ S {
      if (e is a ActivityName)
           New activity and push to stack；
      else if(e is a tag of Sequence)
         HandleSequence();//handle sequence relation
      else if(e is a tag of Switch)
         HandleSwitch ();//Switch relation
      else if(e is a tag of  Parallel)
         HandleParallel();//Parallel relation
      else if(e is a tag of Loop)
         HandleLoop ();//Loop relation
    }
    attach In(a) to the beginning of fragment；
    attach Out(a) to the end of fragment；
end
HandleSwitch(){//other functions are similar
    a1 = stack.pop;
    a2 = stack.pop;
    new DatabasedExclusiveGateway g；
    attach g，a1；//create switch branch
    attach g，a2；
}
```

VI. CASE STUDY AND EXPERIMENTS

In this section, we implemented MBPR with the architecture presented by Fig. 7 and Fig. 8 shows corresponding perspective of MBPR.

The case study in Fig. 6 displays the inner process of loan applications which are modeled using BPMN. Part(a) describe the application procedure in coarse-granularity with annotated information to explain its details, part(c) is a fine-grained business process exits in database server of MBPR, suppose that part(a) is query input and part(c) is the target of retrieval, it is obvious that in terms of structure, syntax or behavior, they do not have a strong similarity. If we measure the similarity between them in no consideration of granularity information, the return quantitative similarity will be very small, which means the retrieval result may not contain the target, but in fact they describe the same application procedure.

In the context of MBPR, using SCMT, the annotated information of activity "*application assessment*" in (a) can be marked and segmented as follows：

&SIMU { draw up contract ∗ verify application information ∗ ***&THEN*** then check credit information, ***&WHILE &IF*** if not sound ∗ waiting ***&ELSE*** else offer loan***}***

Then SCMT compiler of MBPR which implement Algorithm 1 will decompose coarse activity "*application assessment*" into the fragment described in part(b), after replace "*application assessment*" with this fragment, we can obviously see that, no matter from structure, execution path, or connection relations of these processes, the fine-grained business process are more similar to the target process, so it is more promising that the retrieval result contains the target part(c).

Next，we design and conduct experiments to study the performance of MBPR.

The multi-granularity retrieval procedure of MBPR includes two operations: decompose the coarse-granularity business process，calculate similarity using equation (2). The time complexity of equation (2) is O (| *V1* | x | *V2* | x | *E1* | x | *E2* |), where *V1* and *V2* are the number of nodes in *GBM*, *E1* and *E2* are the number of paths in *GBM*. For a coarse activity, the time complexity of decomposition is O(*S*), where S is the number of activity names contained in annotated information. If the coarse-granularity business process contains *n* coarse activities, the overall algorithm complexity is polynomials O(|*S*| x |*n*|+| *V1* | x | *V2* | x | *E1* | x | *E2* |).

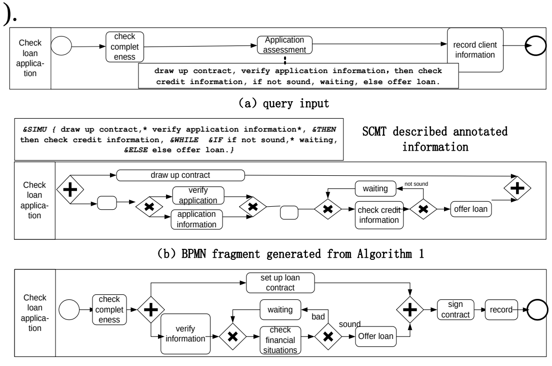

Figure 6. Case study

We assess the effectiveness and efficiency of MBPR. We use 120 bioinformatics process workflows as experimental data set collected from biological research community www.myexperiment.org, which provides a platform for biotechnology researchers worldwide to publish, share and test their processes. In order to maintain consistency with the premise and the context of this paper, we first remodel the collected bio-computing processes using BPMN and post them to the database server of MBPR, each function of a bio-computing process is considered to be an activity in BPMN, and its note is attached to the correspond activity as annotated information, which we marked them with SCMT. The experiments were conducted on a Windows machine with a 2GHz Pentium IV CPU and 2G main memory.

The change of business process versions demonstrate the refinement procedure, so the original versions represent coarse-granularity business process and the latest ones represent fine-grained, experiment 1 simulate the scenario of Fig. 1 by comparing the similarity between the original version and latest version of a same business process. Fig. 9 shows the static results of these records. As we seen, making use of decomposition mechanism before matchmaking, the average similarity is 0.60 while without decomposition is 0.21. This means decomposition mechanism is promising to affiliate the MBPR retrieve more accurate result.

In the second experiment, we consider traditional precision and recall measures that have been extensively used in information retrieval. We randomly extract 10 business processes and set their original versions as "search query models", then set their other versions as "relevant models". We take "search query models" as the input of MBPR, Fig. 10 shows the average precision and recall scores across all the queries. We can see that if querying without decomposition, the precision become very low(only 0.2) when recall rate is equal to 0.55, but if add the decomposition step, precision dramatically decrease until recall rate exceed 0.8. This graph shows that on average, with the decomposition mechanism, MBPR can effectively affiliate multi-granularity retrieval. Using CPU time as metrics, experiment 3 evaluated the relation between retrieval time and the number of coarse activity contained in input business process, as shown in Figure 11；the result was that it is polynomial.

VII. CONCLUSION

In this paper, we proposed MBPR supporting multi-granularity business process retrieval based on a novel method for multi-granularity business process similarity measurement. The contributions of this paper include：A process matchmaking algorithm, Control flow segment and markup Tags (SCMT) is designed; A series of decomposition rules are proposed to refine coarse-granularity business process using annotated information; The effectiveness of MBPR is evaluated based on real data set.

Currently, MBPR has been used to query business process in different granularity. However, the coarse activity decomposition mechanism is highly dependent on the annotated information inputted by modeler. Our future work

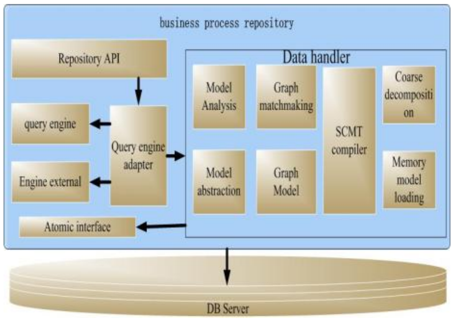

Figure 7. Architecture of MBPR

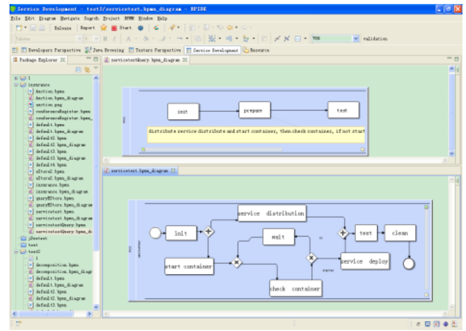

Figure 8. Perspective of MBPR

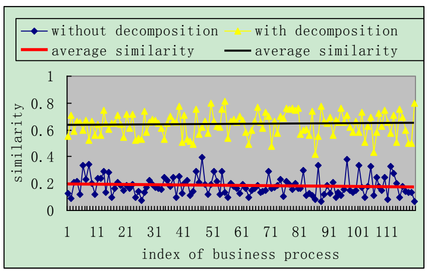

Figure 9. Effectiveness comparison

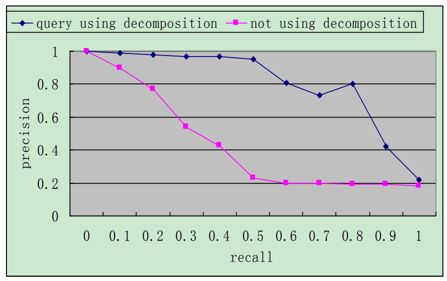

Figure 10. Precision-recall curve

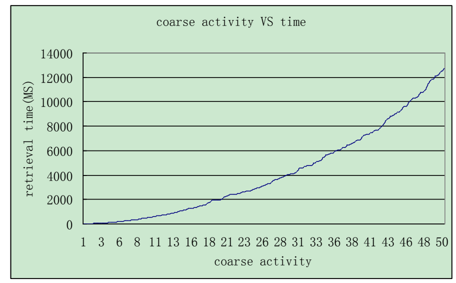

Figure 11. Coarse activity vs time

will be further enhancing the accuracy of decomposition by considering the more information about activity such as Interaction protocol information.

ACKNOWLEDGMENT

This work was partly supported by China 863 program under Grant No. 2007AA010301 and partly supported by China 973 program under Grant No. 2005CB321803.

REFERENCES

[1] K. Shahzad, B. Andersson, M. Bergholtz, A. Edirisuriya, T. Ilayperuma, P. Jayaweera, and P. Johannesson. Elicitation of Requirements for a Business Process Model Repository. BPM 2008 Workshops, pp. 42-53. Springer Heidelberg(2008).

[2] Ma, Z., Wetzstein, B. Anicic, D. Heymans, S. and Leymann, F.: Semantic Business Process Repository. In: Proceedings of the Workshop on Semantic Business Process and Produce Lifecycle Management, Innsbruck (2007)

[3] A. Awad, A. Polyvyanyy, and M. Weske. Semantic querying of business process models. In EDOC, pp. 85-94. IEEE Computer Society, 2008.

[4] B. F. van Dongen, R. M. Dijkman, and J. Mendling, “Measuring similarity between business process models,” in Proceedings of the 20th International Conference on Advanced Information Systems Engineering (CAiSE), ser. LNCS, vol. 5074. Springer, 2008, pp. 450–464.

[5] M. Ehrig, A. Koschmider, and A. Oberweis, “Measuring similarity between semantic business process models,” in APCCM ’07: Proceedings of the fourth Asia-Pacific conference on Comceptual modelling. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2007, pp. 71–80.

[6] R. Dijkman, M. Dumas, and L. Garc´ıa-Ba˜nuelos. Graph matching algorithms for business process model similarity search. In Proc. of BPM 2009, Ulm, Germany, September 2009.

[7] J. Mendling, B. Dongen, and W. Aalst. On the Degree of Behavioral Similarity between Business Process Models. In CEUR-Workshop, pp. 39–58, 2007.

[8] A. Wasser, M. Lincoln, and R. Karni: Accelerated Enterprise Process Modeling Through a Formalized Functional Typology. BPM 2005: LNCS 3649 pp. 446-451.

[9] Oracle Business Models (OBM) http://download.oracle.com/docs/cd/B53825_04/current/acrobat/121cacug.pdf 2009 2010-6-22

[10] SAP Business Map. http://help.sap.com/ 2010 2010-6-22

[11] Phios Process Repository for Supply Chain Management. http://repository.phios.com/SCOR/Default.asp 2010 2010-6-22

[12] White, S.A. (2008-01-17) Business Process Modeling Notation,V1.1. http://www.omg.org/spec/BPMN/1.1/ 2010-6-22

[13] Zicheng H, Jinpeng H, Hailong S, Xudong L, and Xiang L, "BestRec: A Behavior Similarity Based Approach to Services Recommendation," services, pp.46-53, 2009 Congress on Services - I, 2009

[14] G. Miller, “Wordnet: A lexical database for english,” Commun.ACM, vol. 38, no. 11, pp. 39–41, 1995.

[15] T. Pedersen, S. Patwardhan, and J. Michelizzi, WordNet: Similarity–measuring the relatedness of concepts. In Proc. AAAI, pp. 1024-1025, 2004.

[16] Bunke, H. (2000), Graph matching: Theoretical foundations, algorithms, and applications, in 'Proc. Vision Interface 2000, Montreal', pp. 82-88.

[17] D. Shasha , J. Wang , and R. Giugno, Algorithmics and applications of tree and graph searching, Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, June 03-05, 2002, Madison, Wisconsin

[18] P. Bradley, J. Browne, S. Jackson and H. Jagdev, “Business process reengineering (BPR): A study of the software toolscurrently available”, Computers in Industry 25(3) (1995) pp.309-330.

[19] D. Grigori, J.C. Corrales, and M. Bouzeghoub. Behavioral matchmaking for service retrieval: Applicationto conversation protocols. Inf. Syst., 33(7-8): pp.681–698, 2008.

 ISBN: 978-1-61208-105-2