Human Task Scheduling in the Service Composition

Lei Yang
College of Information Science and Technology, Northeastern University, Shenyang, P.R.China
Email: yanglei@ise.neu.edu.cn

Yu Dai
College of Software, Northeastern University, Shenyang, P.R.China

Bin Zhang and Jiawen Pan
College of Information Science and Technology, Northeastern University, Shenyang, P.R.China

Abstract—**This paper presents the architecture of composite service execution engine which can support the execution of the human task. To ensure the efficiency of the human task execution, a method for selecting an appropriate human service resource to perform the task is presented. In this method, the performance evaluation model is formed including the functions of time benefit, load balance and reliability benefit, which can reflect the performance of the human service resources objectively and comprehensively. Based on this model, the scheduling algorithm is proposed to find the human service resource with better performance. The system implementation and the experimentation verify the effectiveness of the method.**

Index Terms—**service; service composition; human task, scheduling of human task; composite service execution engine**

I. INTRODUCTION

BPEL [1] is an XML-based, platform-independent language for business process modeling and execution. It is one of the important technical foundations to implement web service composition [2], which has been widely studied and applied in the field of industry and academia. However, with the extensive use and development of web service and the composite applications, it was found that fully automated business processes have become increasingly unable to meet the needs of practical applications, because some activities of business processes often require human's interaction, which needs the relevant person to take part in.

In 2007, IBM, Adobe, SAP, Oracle and Active Endpoints jointly issued a specification entitled "WS-BPEL4People". This specification [3] is an extension of WS-BPEL standards, which includes human tasks, and allows relevant person to participate in business processes. At the same time, the industry's famous business process suppliers modify their product in order to support human tasks. For example, the human tasks introduced in IBM's WebSphere Process Server [4] is based on the typical human workflow system, and is extended with the functions of notification, the user interface and role-based human tasks allocation. Oracle BPEL Process Manager [5] takes the human tasks as built-in BPEL activities to provide the functions of multi-level approval, standard user interface and policy-based access control. ActiveBPEL [6] provides an interface for customizing task.

Due to human tasks are implemented by relevant persons in the business process, its running time are closely related to the execution capabilities of these persons. For example, in the banking systems, there is a human activity of loan approval. This task needs the participation of relevant persons in the bank. For new recruits and old staffs, the execution time required in performing the tasks is different. Older staffs are more familiar with the related terms of loans and correspondingly need shorter time, on the contrary, the new ones may require a longer time. However, since different business processes may have different execution time constraints, the constraints on human task execution time are different. Thus, in order to ensure the efficiency of the entire business process, human tasks should be allocated dynamically according to performance of the human service resources (the human service resource refers to the person who performs human tasks). However, the current studies for human task allocation in the service composition environment are less, and existing studies are mostly a static and priority pre-specified human tasks allocation, such as the human task allocation strategy in Oracle BPEL. Therefore, how to modeling the performance of the human service resource to reflect the actual effect of performing human tasks, and how to achieve a dynamic human task allocation with the

Manuscript received Oct 30, 2010; revised Dec 23, 2010; accepted Jan. 15, 2010.
Corresponding Author: Lei Yang, email: yanglei@ise.neu.edu.cn
The work is partially supported by the national natural science foundations (No. 60903008), the Fundamental Research Funds for the Central Universities (No. N090404011) and Dr. start-up foundation of Liaoning Province (No.20091022).

doi:10.4304/jsw.6.10.1914-1920

consideration of the performance in order to improve the efficiency of human tasks is the key to be studied and solved.

Thus, this paper proposes a method for scheduling an appropriate human resource service to perform the task. This method uses the established performance evaluation model to reflect the actual performance of the human service resource. Based on this model, the proposed scheduling algorithm can find the human service resource which can implement the human task with better performance. The contributions of this paper include:

(1) The composite service execution engine which can support the human task is designed and implemented. In the engine, the architecture of human service container is proposed in order to flexibly invoking the human service resource.

(2) The model for evaluating the performance of human service resource is proposed including the functions of time benefit, load balance and reliability benefit, which can reflect the effect of the human service resources objectively.

(3) The objective function of the performance evaluation model based human task scheduling problem is presented and the corresponding algorithm for solving the problem is shown. The system implementation and the experimental results demonstrate the effectiveness of the proposed human task scheduling method.

In the following, we will give the architecture of the composite service engine in the section 2; the performance evaluation model of a human resource service and the human task scheduling algorithm is introduced in the section 3; in the section 4, the system implementation and experimentation are presented; in the section 5 the related works are shown; finally, we concludes the paper.

II. Composite Service Execution Engine in Support with the Human Task

Compared to the general service resources (such as storage services, data services, etc.), human service resource is a particular service resource performed by human, which focuses on integrating the human into the service-oriented applications, compared to the general service resources (such as storage services, data services, etc.). At present, the traditional composite service execution engine only supports the invocation of the general service resources. In order to achieved the flexible invocation of human service resources in composite service execution engine, according to BPEL4People specification, this paper takes human task as a kind of basic activities in BPEL, and takes human service resource as a service with the ability to perform the human tasks. Then, the entire execution procedures of human service resources are completely transparent to the requester of business process, which enables loosely coupling between the human service resources and business processes.

Figure 1 shows the framework of a composite service execution engine which can support human task, including *process parser*, *process manager*, *process scheduler*, *activity executor* and *service execution agent*. The entire execution procedures of composite service execution engine are as follows:

(1) *process parser* parses the BPEL4WS [7] documents and WSDL [8] documents, forms the business process templates and hibernates them.

(2) When the request of the execution is received, the *process manager* creates the corresponding business process instance according to the process template, and manages the business process instance along its lifecycle;

(3) *process scheduler* establishes an activity invoking queue according to the control flow of the activities in the business process;

(4) *activity executor* gets the get-ready activity from the activity invoking queue, and passes it to *service execution agent*;

(5) *service execution agent* first determines whether the get-ready activity is a human task. If so, the activity will be handed over to human service container to inform corresponding human service resources; if not, the activity will be handed over to general service resources. After the implementation of the service resources, the *service execution agent* returns the result.

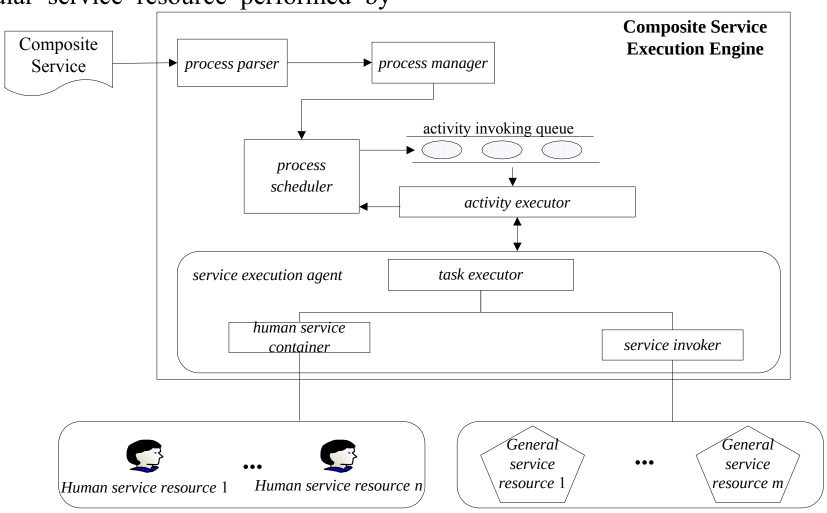

Figure 1. Framework of the proposed composite service execution engine

In the proposed composite service engine, human service container is used to interact with the human resource service. Figure 2 shows the architecture of the human service container. The human service container mainly includes *Human Task Manager*, *Human Task Scheduler*, *Human Services Execution Interface* and *Human Service Resources Registry*.

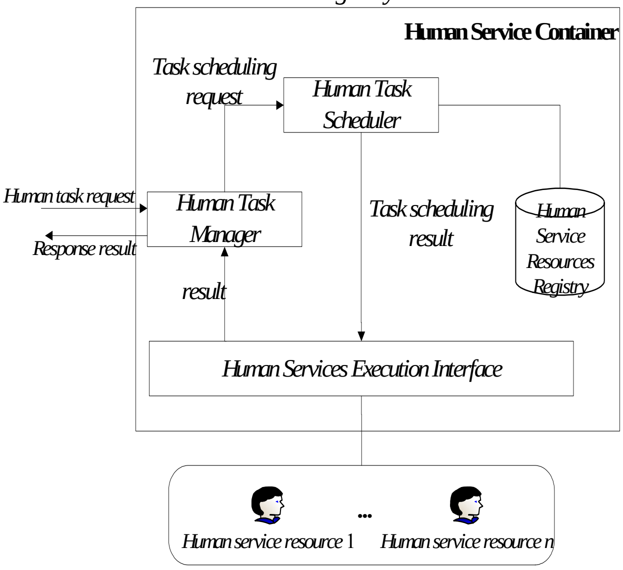

Figure 2. Architecture of human service container

(1) *Human Task Manager*. It is primarily used for managing human task along its lifecycle, including the human task initialization, execution and implementation.

(2) *Human task scheduler*. It is primarily used for allocating human task to the specific human service resources according to their performance capabilities and execution constraints.

(3) *Human Service Execution Interface*. It is mainly used for interacting with the human service resources to implement the human tasks.

(4) *Human Service Resource Registry*. It stores information of human service resources, including the description of the ability and the performance of human service resources.

Since the execution time of a same human task performed by different human service resources can vary a lot. Allocating human task according to the performance capacities of human service resources is needed in order to ensure the efficiency of composite business process. Thus, in the human service container, it needs to study and solve the human task scheduling problem. To this end, we introduced a human task scheduling algorithm based on the performance evaluation model of human service resource. Next, this paper will introduce the performance evaluation model of human service resource first, and on this base, introduce the human task allocation method.

III. Human Task Scheduling Algorithm Based on the Perfromance Evaluation Model

In the execution of human task, due to the different performance of the participants, that is the performance of human service resources varies, the execution time of the human task is different, which results in the different execution time of business processes. Therefore, how to models the execution performance of the human service resources needs to be studied and solved.

Since the execution time of the same human task service implemented by different human service resources can vary a lot, to ensure the efficiency of business processes, human tasks are often assigned to the human service resources whose execution time is relatively short. However, such a scheduling method may cause serious load unbalance problem. That is to say, some of the human service resources are free while the others are over busy. Then, the efficiency of human tasks can be affected seriously. Meanwhile, the performance of human resource services may change frequently (such as in traveling, illness, etc.), so if the human task assigned to these human service resources, it will be unable to be implemented, that will lead to the rate of successful completion of human service resources falls, and affect the efficiency of human task. Therefore, in evaluating the performance of human service resources, it is required to take the execution time of human service resources, load balance, and successful scheduling rate into account. Based on the analysis above, the performance evaluation model of human service resource will be presented in the following.

Definition 1. Performance evaluation model of human service resource is used to evaluate the performance of human service resources u in implementing task t. It can be signified as: $CE(u, t)=<ET(u, t), L(u, t), TR(u, t)>$, where $ET(u, t)$ is a time benefit evaluation function, $L(u, t)$ is a load balance benefit evaluation function, $TR(u, t)$ is a reliability benefit evaluation function.

Definition 2. Time benefit evaluation function $ET(u, t)$ is used to evaluate the relative timeliness of human service resources u in implementing task t, and can be calculated as formula (1).

$$ET(u,t) = \frac{\min_i (FT(u_i,t))}{FT(u,t)} \tag{1}$$

Here, $FT(u, t)$ represents the expected execution time that human task t is assigned to the human service resources u to perform its function. It is a historical average execution time of task t implemented by human service resource u previously. The set of human service resources which can implement human task t is signified as: $U=\{u_1, u_2,\ldots, u_n\}$, $\min_i(FT(u_i,t))$ is the shortest expected time of human task t implemented in set U.

Obviously, $0 \le ET(u, t) \le 1$. In human service resources set U, the closer $ET(u, t)$ to 1 indicates that the relatively shorter time of human task t implemented by human service resources u.

Definition 3. The load balance benefit evaluation function $L(u, t)$ is used to evaluate the relative degree of load balance of human service resources u in the current moment, and can be calculated by formula (2).

$$L(u,t) = \frac{\sum_j Load(u_j) + FT(u,t)}{\max_j(Load\_after(u_j)) \times n} \quad (2)$$

Here, the set of human service resources which can implement human task t is signified as $U=\{u_1, u_2,\ldots, u_n\}$. $Load(u_j)$ represents the current load of human service resources u_j, that is the expected time to implement all the tasks assigned to u_j in the current moment; $\sum_j Load(u_j)$ represents the current load of all the human service resources in set U; $\max_j(Load\_after(u_j))$ represents the maximum load of human service resources in set U after the time when u is picked out to implement the human task t; n represents the number of the human services resources in set U; $FT(u, t)$ represents the expected time of human task t implemented by human service resources u.

Obviously, $0 \le L(u, t) \le 1$. In human service resources set U, the closer $L(u, t)$ to 1 indicates that the more load balance of human service resource u in implementing human task t.

Definition 4. The reliability benefit evaluation function $TR(u, t)$ is used to evaluate the relative reliability of human task t implemented by human service resources u, which can be calculated by formula (3).

$$TR(u,t) = \frac{UT(u,t)}{\max_i(UT(u_i,t))} \quad (3)$$

Here, $UT(u, t)$ represents the reliability of human task t being implemented by human service resource u at the current moment. The reliability can be calculated by formula (4). $\max_i(UT(u_i,t))$ represents the current greatest reliability of all human service resources in set U.

$$UT(u_j,t) = \frac{Stcount(u_j,t)}{Tasksum(u_j,t)} \quad (4)$$

Here, $Stcount(u_j, t)$ represents the number of tasks which have been implemented successfully by human service resource u_j, $Tasksum(u_j, t)$ represents the total number of tasks human service resources u has received.

Obviously, $0 \le TR(u, t) \le 1$. The closer $TR(u, t)$ to 1 indicates that the more reliability of human service resource u, and the higher success rate of human task t. That is to say, the higher possibility of human service resource u performs t successfully.

For human service resource u which can implement human task t, its performance evaluation function $CE(u, t)$ is used to evaluate the comprehensive performance of u in implementing t, which can be calculated by formula (5) as following.

$$CE(u,t) = ET(u,t) \times w_{ET} + L(u,t) \times w_L + TR(u,t) \times w_{TR} \quad (5)$$

Here, $0 \le w_{ET}, w_L, w_{TR} \le 1$ and $w_{ET}+w_L+w_{TR}=1$.

The larger value of $CE(u, t)$ indicates the better comprehensive performance of human service resource u in implementing human task t, and consequently the higher efficiency of the business process.

The set of human service resources which can implement the human task t is signified as: $U=\{u_1, u_2,\ldots, u_n\}$. Base on the performance evaluation model of human service resources, this paper establishes an objective function for human task scheduling (as formula (6) shows): to make human task t get the highest performance evaluation on human service resources.

$$\max_{u_i \in U}(CE(u_i,t)) \quad (6)$$

Here, $U=\{u_1, u_2,\ldots, u_n\}$ is a set of human service resources and for each human service resource in U, it can implement the task t.

According to the formula (6) which describes the performance evaluation model based objective function for human task scheduling, the human task scheduling considers three performance factors including reliability, execution time and load balance to improve the comprehensive performance of human task. In the human task scheduling process, the system will calculate the objective function value according to performance evaluation of human service resources, and select the human service resource which gets the largest objective function value as scheduling results. The human task scheduling algorithm is as follows:

Input: human task t, set U formed with human service resources which can complete human task t

Output: human service resources that can implement human task t

Scheduling process:

For each u_i in U

  Calculate $ET(u_i, t)$;

  Calculate $L(u_i, t)$;

  Calculate $TR(u_i, t)$;

  Calculate $CE(u_i, t)$;

  If the maximum performance evaluation *max* is less than $CE(u_i, t)$ then

    $max=CE(u_i, t)$ and $u=u_i$;

Endfor

Output u.

IV. System Implementation of Human Service Container and Experiments

A human task container is developed based on the existing composite service engine whose architecture is shown in Figure 2. Figure 3 shows the sequence diagram of human task container. Firstly, the human task with the highest priority is taken from human task request queue, and the corresponding human task instance is built and stored by persistence layer. And then, human task scheduler will finish the human task dispatching. Then, the human task is arranged into human task pool, and waiting for the execution interface's invoking. Once the human task instance is finished, it is eliminated from pool and its status is changed in human task manager. Finally, the response message is returned to the invoker.

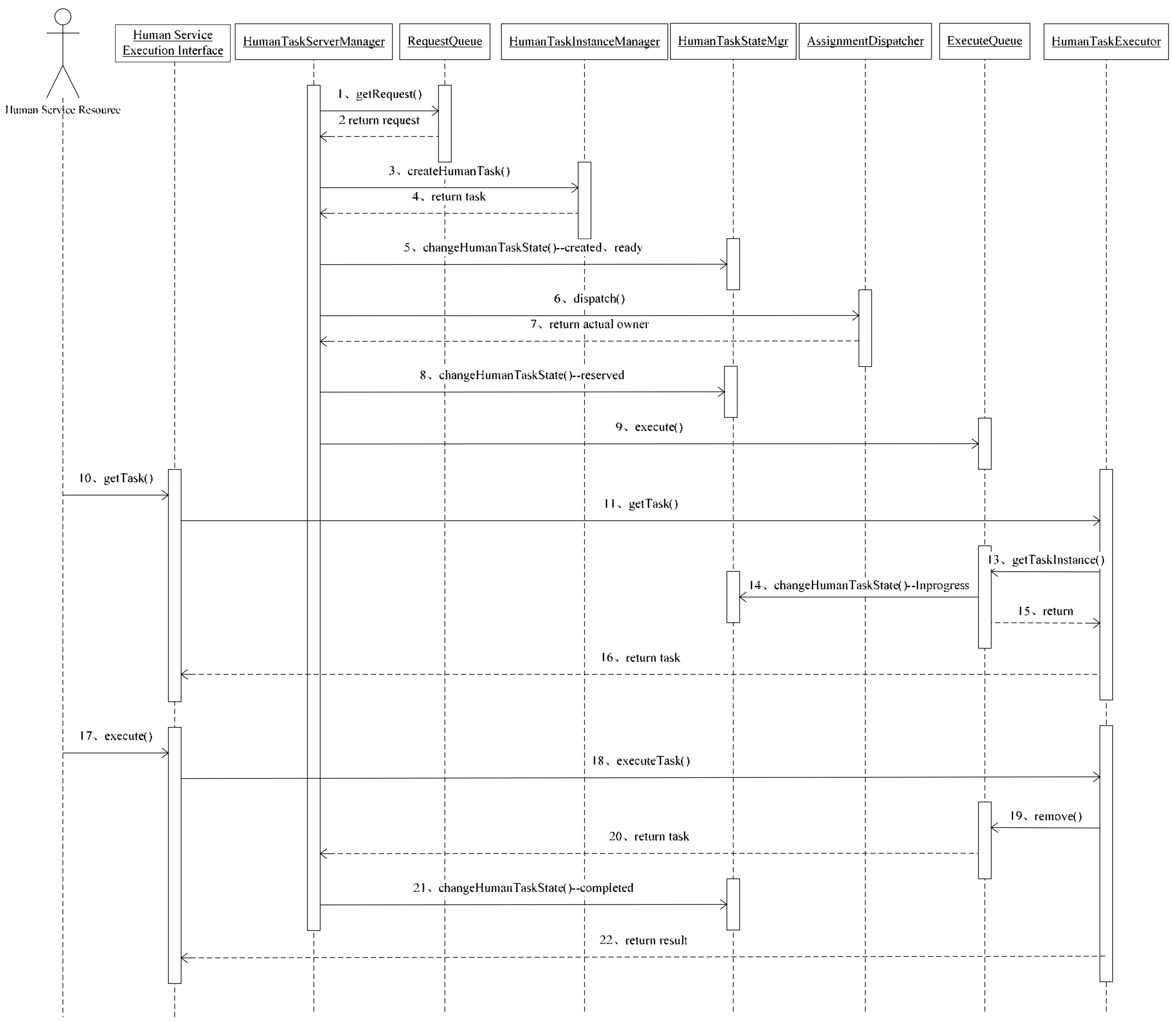

Figure 3. Sequence diagram of human service container

In the human task container, XML is used to represent variables in human tasks which makes it easy to store and illustrate the variables. In another hand, to illustrate the interaction with the human flexibly, the data of the human service execution interface is also formulated in XML document, and exhibited through XSLT technique. Figure 4 is an example of human service execution interface which provides loan approval function.

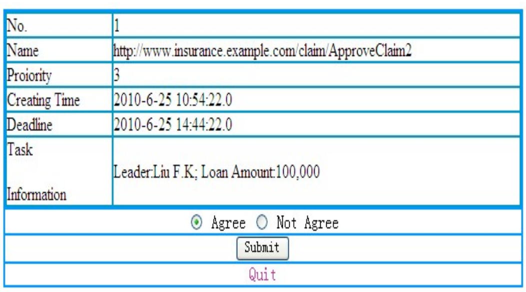

Figure 4. Human service execution interface for loan approval

To verify the effectiveness of the proposed human task scheduling algorithm, the following experiment is done. In the experiment, success rate (the number of human tasks finished before deadline divided by the total number of the human tasks) is to evaluate the effectiveness of the algorithm. 10 human tasks are used for testing and each one can be implemented by 20 human service resources. In different situation, w_{ET}, w_L and w_{TR} will be set to different value. For different task, the experiments will be done 20 times and in different time, the human service resources will have different performance. The experiment results are shown in Figure 5.

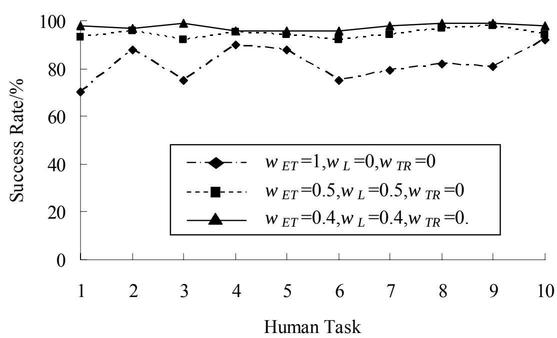

Figure 5. Comparison of success rate

Figure 5 shows that the proposed algorithm can guarantee the successful executions of the human tasks when considering the execution time, load balance and reliability. When only considering the execution time or the load balance, the success rate of the human task is lower. This is mainly because the proposed performance evaluation model can reflect the actual performance of the human resource service more objectively and comprehensively, as a result of which the success rate of the human task can be ensured.

The above system implementation and experimentation show the better performance of the proposed method.

V. RELATED WORKS

Independent task scheduling algorithm is one of the classic scheduling problems, which has been studied wisely and deeply. Currently, a lot of efficient task scheduling models and methods are proposed. In Ref.[9], load balance and execution efficiency is considered to implement task scheduling. In Ref. [10], reliability is introduced to task scheduling, in which the security and reliability of the tasks are considered and an aggregation function is used to combine these two factors together in order to improve the comprehensive performance of the task scheduling. Ref.[11] considers the QoS of users and system factors, and proposes an algorithm which can improve the scheduling performance, especially in reducing the execution time and load balance. However, most of current task scheduling approaches focus on general computing resources but not on human service resources. As the special nature of the human service resource (i.e. the execution performance changes frequently and lasts for a long time, the execution performance also is affected by the load balance largely, et al), the actual performance of the human task may be affected to a large extent. Thus, when solving the problem of human task scheduling, it needs to consider the characteristics of human service resource. For this, the paper proposes a model for evaluating the performance of human service resource and based on this model, the proposed scheduling algorithm is given.

VI. CONCLUSIONS

For the problem of human task execution in the service composition, this paper presents a composite service execution engine. In the engine, the architecture of human service container is proposed in order to flexibly interact with the human service resource. Meantime, in order to ensure the efficiency of the human task execution, the model for evaluating the performance of human service resource is proposed including the functions of time benefit, load balance and reliability benefit, which can reflect the performance effect of the human service resources objectively and comprehensively. Based on this model, the objective function of the performance evaluation model based human task scheduling problem is presented and the corresponding algorithm for solving the problem is shown. The system implementation and the experimental results demonstrate the effectiveness of the proposed human task scheduling method.

However, in this paper we does not consider the situation when the human tasks have relations with each other, for example, some tasks need to be implemented by the same human resource service. In the future, we will research on this problem.

ACKNOWLEDGMENT

The work was partially supported by the national natural science foundations (No. 60903008), the Fundamental Research Funds for the Central Universities (No. N090404011) and Dr. start-up foundation of Liaoning Province (No.20091022). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] N Lohmann, P Massuthe, C Stahl, et al. “Analyzing Interacting WS-BPEL Processes using Flexible Model Generation”. Data and Knowledge Engineering.Volume 64, Issue 1,pp 38-54. January 2008.

[2] L Z Zeng, H. N Anne, B Benatallah, et al. “Dynamic composition and optimization of Web services ”. Distributed Parallel Databases. Volume 24, Issue 1-3,pp 45-72. December 2008.

[3] A Agrawal, M Amend, M Das, et al. “WS-BPEL Extension for People Version 1.0”. http://www.ibm.com /developerworks/webservices/library/specification/ws-bpel4people/, 2008.

[4] D F Ferguson, R kerth. “WebSphere as an e-business server”. IBM Systems Journal. Volume 40, Issue 1,pp 25-45. January 2001.

[5] Ocacle Corp. “Oracle BPEL Process Manager”. http://www.oracle.com/technology/products/ias/bpel/index .html, 2008.

[6] Active Endpoints. “ActiveBPEL Engine”. http://www. active-endpoints.com, 2008.

[7] G Ning, Y Zhu Y, T Lu, et al. “BPELGEN: An Algorithm of Automatically Converting from Web Services Composition Plan to BPEL4WS”. Proceedings of the 2nd International Conference on Pervasive Computing and Applications. Brmingham: IEEE Computer Society, pp 600-605. July 2007.

[8] J Nitzsche, T VanLessen, F Leymann. “WSDL2.0 Message Exchange Patterns: Limitations and Opportunities”. Proceedings of the 3rd International Conference on Internet and Web Applications and Services. Athens: IEEE Computer Society, pp 168-173. June 2008.

[9] C Banino, O Beaumont, L Carter, et al. “Scheduling Strategies for Master-Slave Tasking on Heterogeneous Processor Platforms”. IEEE Transactions on Parallel and Distributed System. Volume 15, Issue 4, pp 319-330, April 2004.

[10] S Song, Y K Kwork, K Hwang. “Trusted job scheduling in open computational grids: Security-driven heuristics and a fast genetic algorithm”. Proceedings of the 19th IEEE International Parallel & Distributed Processing Symposium. Denver: IEEE Press, pp 33-40, April 2005.

[11] Y H Li, D P Zhao, J Li. "Scheduling Algorithm Based on Integrated Utility of Multiple QoS Attributes on Service Grid" Proceedings of the 6th International Conference on Grid and Cooperative Computing. Washington D.C. IEEE Computer Society, pp 288-295. August 2007.

Lei Yang is bon in Liaoning province, China, in 1974. He received the M.A and Ph.D. in computer science from Northeastern University, China in 2004 and 2007 respectively. He is a lecture at college of information science and engineering, Northeastern University, China. His current research interests include web service composition and quality management.

Yu Dai is born in Liaoning province, China, in 1980. She received the Ph.D. in computer science from Northeastern University, China in 2008. She is a lecture at college of software, Northeastern University, China. Her current research interests include web service composition and quality management.

Bin Zhang is born in Liaoning province, China, in 1974. He received the Ph.D. in computer science from Northeastern University, China in 1997. He is a professor at college of information science and engineering, Northeastern University, China. His current research interests include web service composition and information management.

Jiawen Pan is born in Liaoning province, China. He received his B.A. in computer science from Northeastern University, China in 2009. He is now a master candidate and currently he is interested in the web service composition.