A Service-Merging-Enabled SOA Framework for Inter-Enterprise Collaboration*

Sen Zeng**
Department of Automation, Air Force Airborne Academy, Guilin 541003, China
Email: zengsen04@gmail.com

Mingguang Yang and Yimin Shen
Department of Computer Engineering, Chengdu Technological University, Chengdu 610071, China
Email: shen_yimin@sina.com

Yuanhua He
Department of Automation, Air Force Airborne Academy, Guilin 541003, China
Email: zs12181@163.com

***Abstract*—The functional constraints of service matching and performance requirements (i.e., quality of service, QoS) of service selection are separated distinctly under the traditional service-oriented architecture (SOA). Hence the service matching and service selection may be performed in turn and independently. And the complexities of the implementation of SOA are simplified also. In fact, the functional constraints and QoS are stretching and compromising frequently. And service providers with similar functions would come into being a merged-service with more powerful function and higher performance by collaboration with each other. This paper proposes an extended QoS (exQoS) system that contains some service functional constraints and traditional performance indicators. Then a service merging enabled SOA (meSOA) is established based on the exQoS, and the meSOA may be used to support service merging across enterprises. Furthermore, the problem of service matching and selection based on merged-service is formulated as a nonlinear planning (NLP) problem, and its corresponding solving algorithm is proposed also. At last, this NLP method and traditional method are compared with a numerical value experiment, which verified its correction and the superiority of this method over the traditional one.**

***Index Terms*——service merging, service-oriented architecture (SOA); inter-enterprise collaboration; extended-QoS**

I. INTRODUCTION

Service-Oriented Architecture (SOA) unifies various resources as services, and a suit mechanics of service search, match, selection, binding, and running are proposed too[1][2]. With the emerging and rapid development of SOA and cloud computing, the flexibility and availability of forming new business solutions by service orchestration and choreograph with various business services and IT resources are enhanced evidently[3]. Especially with the economic globalization and the emergence of service-oriented manufacturing (SOM) paradigm, the merging of across enterprise manufacturing capability implemented by service merging usually provides more powerful service functions and higher level of QoS to the merged manufacture enterprise cluster [4][5][6]. For example, an enterprise cluster can produce lager quantity of products in a certain period to meet a customer's requirement, but each of them can't provide that quantity alone [7][8]. Correspondingly, an enterprise's ability of across enterprise service merging is required urgently. That's to say, if a service provided by an enterprise can not be merged with other enterprises' services, a shared resource deployed on a cloud computing platform can't be formed, thus restricting its being researched and selected.

The flow of service search, service matching, and service selection is described approximately as below. Function matching is carried out first to decide whether a service provider satisfies the service requester's function requirement. Then an optimal one is selected from all the candidate service providers that satisfy the requester's functional requirement by QoS, in order to optimize the overall performance of a business process or service composition that the service requester is orchestrated[9][10][11][12][13][14][15]. Therefore, the functional constraints and performance requirements play distinct roles in the flow of service search, matching, and selection.

The process of service matching is carried out under the rigid functional constraints of the service requester. That's to say, if a service provider can't reach the threshold of the functional constraint set up by the service requester, it will be washed out from the candidates.

The qualifiers will be selected further by performance indicators, i.e. QoS, in order to optimize certain overall performance of business solutions. And the performance indicators usually are stretching and compromising.

Apparently, the distinct separation of functional matching and performance optimization brings independent implementation of service matching and service selection. However, this implementation mode of

Manuscript received September 5, 2012; revised December x, 2012; accepted January x, 2013.
*The work of the paper was granted by Guangxi Natural Science Foundation, China (No. 2010GXNSFB013057). **Corresponding author

doi:10.4304/jsw.8.7.1586-1591

SOA for service-oriented enterprise suffers unexpected practical problems. First, the boundary between the functional rigid constraints and QoS is not so absolute. In fact, some seemingly rigid functional constraints are compromising. For example, the requirement of "delivery-cycle: two weeks" may be a rigid constraint for a service requester, but when the delivery-cycle postponed 1 day will make it obtain concession of price from the service providers which will make up the punishment of extended deliver-cycle for customers, the rigid constraint of delivery-cycle can be degradated to an ordinary performance indicator of QoS. The instance is frequent in service composition, i.e. almost all constraints of non-critic path can be transformed to negotiable performance indicators of QoS.

However, it is a pity that the existing architecture of SOA provides just composition methods of QoS (mainly for sequence, parallel, branch, and loop structures) for different functional services [10][11][15]. It can not support service-merging for services with the same functions in order to compose a merging-service that has not only higher level of quality of service but also more powerful business functions. This paper constructs a service merging-enabled SOA (meSOA) framework that makes service resources in SOA and cloud computing platform have higher level of service (LoS) in order to support across enterprise collaboration under SOA and cloud computing environment.

I. A Service Merging-Enabled SOA Framework

Suppose a service requestor, its service performance is evaluated by n individual indicators of QoS, $\{q_1,q_2,...,q_n\}$. Among the m service providers arrival and response to the service requestor in succession $\{S_1,S_2,...,S_m\}$, the value of corresponding QoS indicators of S_j is $\{q_{1j},q_{2j},...,q_{nj}\}$.

For the selected service provider S_j, the service requester requires it satisfying the constraints as below:

(1) Its values of QoS indicators not less than pre-determined constant Q_i:

$$q_{ij} \geq Q_i (i=1,2,...,n) \quad (1)$$

(2) The service requester pursues the maximum of the linear composition of $\{q_{1j},q_{2j},...,q_{nj}\}$, i.e., the objective function is:

$$v = \max\left(\sum_{i=1}^{n} w_i q_{ij}\right)_{j=1}^{m} \quad (2)$$

Where, $\{w_i|i=1,2,...,n\}$ are constants selected by the service requester with its own needs. Many references have discussed how to select the weighted constants such as Ref. [10][11][15][16]. Since the n indicators of QoS $\{w_i|i=1,2,...,n\}$ have different dimensions, normalization is not required.

The traditional methods of service matching and selection usually have two steps:

Step 1, the service providers of $\{S_j\}_{j=1}^{m}$ that can't satisfy equation (1) are filtrated;

Step 2, take traverse or other methods to pick out a service provider which will optimize equation (2) from the providers that satisfy equation (1).

In fact, with step 1, large numbers of service providers will be washed out thus constraint the available service providers significantly. However, on the premise of service having the capability of merging, this paper improves above traditional method in order to make the service requester have much better qualities of service. The essentials of this new method are:

(1) When a service requester releases requests, specify whether the responding service providers should have the capability of merging;

(2) Service providers will send responds to the service requester in succession, and they become the sources of service merging;

(3) The server of service merging obtains the optimal scheme of service merging by the method of this paper, and notifies service requester of the QoS of the merged-services.

(4) Service requester binds the sources of service merging through the server of service merging, and starts to consume them.

Note that since SOA has the feature of P2P, after the server of service merging have provided the merged-services to the service requester, it will not participate in the subsequent processes of service matching, selection, and binding. Consequently, if a service requester expects to know the details of sources of service merging (i.e., the service providers composing the merged-service) before its binding with merged-service, it may send the request to the server of service merging, and the information including the information of the service providers and the original QoS) will be provided by the latter.

The service merging enabled SOA framework is shown in Fig.1.

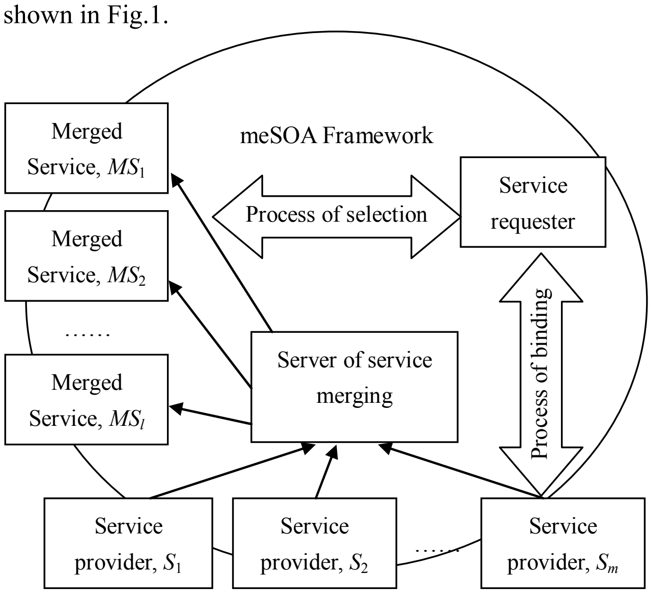


Figure 1. A service merging enabled SOA Framework

II. Extended QoS and Its Nonlinear Planning Problem

Although the practical QoS may be various in different application context, they are usually classified into five classes in literatures namely time, cost, reputation, success, and availability[10][11][15][16][17][18].

It is obvious that the indicators of quantity and quality are far more important than the five classes of QoS enumerated above. The reason why they are not listed above is just about that these indicators are functional indicators with the feature of rigidity. The meSOA framework shown in Fig.1 makes it possible for the first time that these rigid indicators may be merged into SOA.

In fact, with the across enterprise business collaboration represented by service merging, the performance indicators including quantity and quality will be enhanced again possibly. For example,

- In a period of time, two manufacturing enterprises can produce 1000 parts and 1200 parts individually. Obviously, by the merging of their capability of manufacturing, it is possible for them to finish a task of manufacturing 1800 parts together, though each one of them can not do it by itself.
- The ratio of one time correction commonly is 99%, i.e., the ratio of error is 1%, in the service domain of data input in business process outsourcing (BPO) industry. With the addition of stage of QA, the ratio may be improved up to 99.9%. But it is possible that an enterprise still can not reach the requirement of correct ratio for some critic services. However, with the across enterprise business collaboration by merging more enterprises' capabilities into the stage of QA, they will increase the correct ratio up to 99.99% for a key customer.

The QoS system comprising quantity, quality, and other rigid performance indicators is denoted as extended QoS (exQoS) in this paper.

Let x_j denote the quantity or workload assigned to the j_{th} service provider, and Ψ_i the merging operator of the i_{th} indicator of exQoS when these service providers are comprising a virtual enterprise. Thus, the performance indicators of exQoS of the virtual enterprise merged with these service providers are:

$$q_i=\Psi_i(q_{ij}, j=1,2,...,m)$$

Where Ψ_i may be various linear and (or) nonlinear function format.

Combined with equation (2), a service selection problem of meSOA can be formulated as a non-linear planning (NLP) problem as below:

$$\min \quad -\sum_{i=1}^{n} w_i \Psi_i\left(\{x_j\}_{j=1}^{m}\right)$$

$$s.t.\quad \begin{cases} \Psi_1\left(\{x_j\}_{j=1}^{m}\right)=Q_1 & & (3) \\ \Psi_i\left(\{x_j\}_{j=1}^{m}\right)\geq Q_i & i=2,3,...,n & (4) \\ x_j \geq 0 & j=1,2,...,m & (5) \end{cases}$$

Where, constraint equation (3) insures to complete the whole tasks, constraint equation (4) insures the satisfaction of the requirements of the other six performance indicators of exQoS, constraint equation (5) indicates that the quantity of task of each service provider is no less than zero.

It is obvious that the possible solution of the above NLP problem depends on the representation format of *Ψi*. Referring to the classification of operators of service composition in [10], the mathematical format of service-merging-operators for service requester to estimate its value of performance objective is recommended in Table 1.

TABLE I. MATHEMATICAL FORMAT OF SERVICE-MERGYING - OPERATORS

Extended QoS indicators	Merging operators
Quantity (q_1)	$\Psi_1=\sum_{j=1}^{m} x_j$
Qulity (q_2)	$\Psi_2=\sum_{j=1}^{m} \frac{q_{ij}x_j}{Q_1}$
Cost (q_3)	$\Psi_3=-\sum_{j=1}^{m} \frac{q_{ij}x_j}{Q_1}$
Time (q_4)	$\Psi_4=-\max\left(\frac{q_{ij}x_j}{Q_1}\right)_{j=1}^{m}$
Reputation(q_5)	$\Psi_5=\sum_{j=1}^{m} \frac{q_{ij}x_j}{Q_1}$
Success (q_6)	$\Psi_6=\sum_{j=1}^{m} \frac{q_{ij}x_j}{Q_1}$
Availability (q_7)	$\Psi_7=\sum_{j=1}^{m} \frac{q_{ij}x_j}{Q_1}$

Among them, for the indicators of cost and time, the lower values mean better performance, so that the operators of q_3、q_4 always comprise minus.

It can be verified, with the definition of service-merging-operators in Table 1, the NLP problem above is a convex planning problem and its available solution domain is convex. And its local optimal value is the whole optimal value. Therefore, the method of this paper has well available calculation. Many common solution algorithms can be used to solve the NLP problem [19].

Generally speaking, the NLP problem can be run off-line, i.e., by waiting for a span of time or a certain numbers of service providers coming and responding to the service requester, users construct the above NLP problem and solve it to obtain the optimal scheme of service merging solutions at one time. When the intervals among the coming service providers are relative long, users can run the above NLP model on-line, i.e., when a new service provider comes, solve the above NLP problem immediately and judge whether the result of the service merging is satisfying, then decide whether to wait a new comer responding to the service requester.

III. CASE STUDY

Suppose a customer demands Q_1=1,000 parts in -Q_4=25 days, the cost no more than -Q_3=6000 dollars, and the relative quality up to Q_2=0.99. For the supplier, its reputation must better than Q_5=0.9, the success and availability must larger than Q_6=0.95, Q_7=0.99 respectively. The customer wants the weights of quality, cost, time, reputation, success, and availability as w_2=1、w_3=0.05、w_4=0.0002、w_5=1、w_6=1、w_7=1.

Under the context of this customer's demand, m=1,000 service providers are constructed randomly. Values of their performance indicators are all subject to normal distribution with parameters of mean and variance as Table 2.

TABLE II. PARAMETERS OF PERFORMANCE INDICATORS OF SERVICE PROVIDERS CONSTRUCTED RANDOMLY

QoS	Mean	Variance
Quantity (q_1)	-	-
Qulity (q_2)	0.99	0.1
Cost (q_3)	6000	1000
Time (q_4)	25	5
Reputation(q_5)	0.9	0.1
Success (q_6)	0.95	0.1
Availability (q_7)	0.99	0.1

For comparison, adopting the traditional method (as the one used in Ref. [10] [11]) first, all service providers are filtrated with the requirements of q_1~ q_7, then traverse the left and select the optimal service provider according to the objective function. Then, the optimal merging-service can be selected with the method proposed in this paper by solving the NLP convex planning problem. Comparative results of a typical numerical experimentation are shown in Table 3.

It can be seen from this experimentation that the number of available schemes of service composition increased 3 orders of magnitude with the method of this paper. Among them, the merging-service composed by S_{251} and S_{446} is selected. With the quality, reputation, success, and availability all satisfying the requirements, its cost reduced 13.58%, time shortened 56.71%, and the overall objective is increased 23.55%.

TABLE III. COMPARATIVE RESULTS OF A TYPICAL NUMERICAL EXPERIMENTATION

Metrics \ Methods	Objective	Traditional method	Merging method
Number of available schemes of service composition	-	12	52202
Optimal schemes of service composition	-	S_{251} produces all the 1000 parts	S_{251} produces 433 parts, and S_{446} produces 567 parts
Qulity (q_2)	≥0.99	1.151039685	1.072544504
Cost (q_3)	≤6000	3773.751179	3261.275
Time (q_4)	≤25	23.03823089	9.972507453
Reputation(q_5)	≥0.9	1.046552477	0.948414277
Success (q_6)	≥0.95	1.062279699	1.000112414
Availability (q_7)	≥0.9	0.947347158	0.972167835
Optimal value of overall performance objective	-	2.300557238	2.842358657

The results of the above numerical experimentation repeated 300 times are shown in Table 4.

TABLE IV. RESULTS OF THE ABOVE NUMERICAL EXPERIMENTATION REPEATED 300 TIMES

Items \ Method	Traditional method	Method of this paper	Improve
Number of average available schemes of service composition	8.026666667	51036.99	6,357 times
Average value of overall performance objective	2.305285518	2.851777002	23.71%

In fact, probability distributions of the two methods are shown as Fig.2.

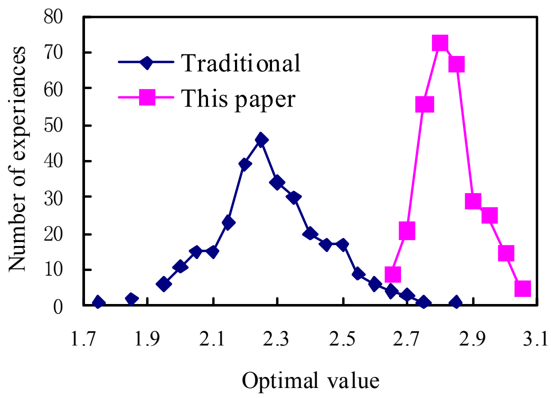


Figure 2. Optimal value' distribution probabilities of this paper's method and the traditional ones

Fig.2 shows the mean and variance of the method of this paper are better than those of the traditional method significantly.

Obviously, the results of numerical experimentations will be influenced by the parameters set in Table 2. With the values of variance in Table 2 increased or decreased 20% each time, the results of the numerical experimentations about the average value of overall performance objective obtained by the traditional method and the method of this paper are compared in Fig.3.

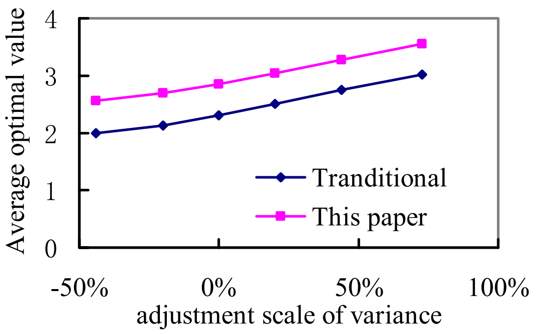


Figure 3. Optimal value of overall performance objective by different Parameters

Fig.3 shows, although the adjustment of variance is important to the results of the numerical experimentations about the average value of overall performance objective, i.e., larger variances bring better results, the method of this paper is always better than the traditional ones: the average values of overall performance objective with the method of this paper are always larger than those with the traditional method by about 0.6.

IV. 4 CONCLUSIONS

SOA and cloud computing promote enterprise to expose and access well-defined services, and abstract them into business process layer and application of service composition further, which facilitate across enterprise collaboration and business process sharing. The overall performance of business process and service composition depends on their structure, the QoS and function of the services selected and bound. The functional constraints of service matching and QoS of service selection are separated distinctly under the traditional SOA: if a service provider can't reach the threshold of the functional constraint set up by the service requester, it will be washed out from the candidates; Then the qualifiers will be selected by performance indicators of QoS to optimize certain overall performance of business solutions; At last, each service requester binds a service provider. The above traditional method has simplified the complexities of the implementation of SOA. In fact, the functional constraints and QoS are stretching and compromising frequently. And some service providers with similar functions would come into being a merged-service that is more functionally powerful and with higher performance by collaboration with each other.

The extended QoS (exQoS) system can be used to contain quantity, quality, and other functional requirements and traditional QoS according to business demands. The service merging enabled SOA (meSOA) established based on the exQoS has expressed the implementation process of service merging in detail and has promoted across enterprise collaboration. Merging-service based across enterprise business collaboration and business process sharing enhance the reuse of services and expand the candidate service providers for service requester of service matching and service selection across enterprise significantly. And the performance indicators including quantity and quality can be enhanced again possibly.

With a nonlinear planning (NLP) problem, this paper has formulated the merging-service based service matching and selection problem. By comparing with traditional method we verified the correctness of exQoS and meSOA and the superiority of our method.

Future work is mainly to improve the practicality of the exQoS, meSOA and NLP proposed in this paper. First is the problems of selection and rationality of performance indicators of rigid functional constraints and traditional QoS included in exQoS. The real world problem is much more complex than the numerical experimentations demonstrated in this paper. It is therefore difficult to select appropriate performance indicators and quantify their values. Then is the implementation of meSOA, for example, how to describe the feature of service merging for service requesters and service providers, how to construct server of service merging, and so on.

V. REFERENCES

[1] J. M. Tien, "Services: A system's perspective," *IEEE Systems Journal*, vol. 2(1), pp. 146-157 March 2008.

[2] M. P. Papazoglou, *Web Services & SOA: Principles and Technology*. Pearson – Prentice Hall, 2012.

[3] D. S. Linthicum, *Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide*, Chinese Simplified Language Edition, Beijing: Posts & Telecom Press, 2011.

[4] L. Y. Sun and G. Li, "Service-Embedded Manufacturing: Advanced Manufacturing Paradigm in 21st Century," *Chinese Mechanical Engineering*, vol.18(19), pp. 2307-2312, 2007.

[5] H. Gebauer, E. Fleisch, and T. Friedli, "Overcoming the Service Paradox in Manufacturing Companies," *European Management Journal*, vol. 23(1), pp. 14–26, 2005.

[6] J. Marceau and C. Martinez, "Selling solutions: Product-service packages as links between new and old economies," Available from: http://www.druid.dk/conferences/summer2002/Papers/Martinez_Marceau.pdf [Accessed 20 March 2010]

[7] Y. Hu, X. H. Zhou, and C. X. Li, "Internet-based intelligent service-oriented system architecture for collaborative product development," *International Journal of Computer Integrated Manufacturing*, vol. 23(2), pp. 113-125, 2010.

[8] P. Phaithoonbuathong, R. Harrison, A. West, R. Monfared, and T. Kirkham, "Web services-based automation for the control and monitoring of production systems," *International Journal of Computer Integrated Manufacturing*, vol. 23(2), pp. 126-145, 2010.

[9] J. M. Ko, C. O. Kim, and I. H. Kwon, "Quality-of-service oriented web service composition algorithm and planning architecture," *Journal of Systems and Software*, vol. 81(11), pp. 2079-2090, 2008.

[10] Y. M. Shen and Y. S. Fan, "Online Selection Approach for Service Composition in Enterprises Cooridination," *Computer Integrated Manufacturing Systems-CIMS*, Vol. 14, pp. 799-805, 2008. (in Chinese)

[11] S. Zeng, S. X. Huang, and Y. S. Fan, "Service-Oriented enterprise network performance analysis," *Tsinghua Science and Technology*, Vol. 14(4), pp. 492-503, 2009.

[12] N. B. Mabrouk, S. Beauche, E. Kuznetsova, N. Georgantas, and V. Issarny, "QoS-aware service composition in dynamic service oriented environments," *Proceedings of the 10th ACM/IFIP/USENIX International Conference on Middleware*, LNCS 5896, pp. 123–142, 2009.

[13] D. A. Menascé, E Casalicchio, and V. Dubey, On Optimal Service Selection in Service Oriented Architectures, " *Journal of Performance Evaluation*, vol. 67(8), pp. 659-675, 2010.

[14] W. Zhang, C. K. Chang, T. Feng, and H. Jiang, "QoS-based Dynamic Web Service Composition with Ant Colony Optimization," *IEEE 34th Annual Computer Software and Applications Conference*, pp. 493-502, 2010.

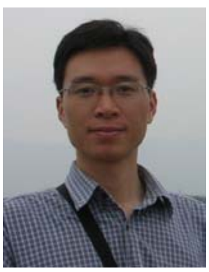
[15] S. X. Huang, S. Zeng, Y. S. Fan, and G. Q. Huang, "Optimal service selection and composition for service-oriented manufacturing network," *International Journal of Computer Integrated Manufacturing*, vol. 24(5), pp. 416-430, 2011.

[16] B. Liu and Y. S. Fan, "A New Performance Evaluation Model and AHP-Based Analysis Method in Service-Oriented Workflow". *Grid and Cooperative Computing GCC2007*, Aug.16-18, Urumchi, Xinjiang, China. pp. 685-692, 2007.

[17] Y. Dai, L. Yang, B. Zhang, and Y. Gao, "QoS for Composite Web Service and Optimizing", *Chinese Journal of Computers*, vol. 29(7), pp. 1167-1178, 2006. (in Chinese).

[18] L. Z. Zeng, B. Benatalliah, A. H. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "QoS-aware middleware for web services composition," *IEEE Transaction on Software Engineering*, vol. 30(5), pp. 311-327, 2004.

[19] D. G. Luenberger and Y.Y. Ye, "Linear and nonlinear programming," third edition, Springer Science + Business Media, LLC, New York, USA, 2008.



Sen Zeng received the Ph.D. degree in control theory and engineering from Tsinghua University, Beijing, China in 2009. His research interests include enterprise architecture, business process modeling methods and optimization analysis, complex system modeling and simulation, performance analysis and optimization, service-oriented computing, Web services, etc.