# The Service Recommendation using QoS in Ubiquitous Computing

Yun-Young Hwang*, Jungsun Yoon*, Kyu-Chul Lee**
*KISTI, **Chungnam National University, Korea
E-mail : {yyhwang, jsyoon}@kisti.re.kr, kclee@cnu.ac.kr

## 1. Introduction

In ubiquitous computing, we considered the following points. The fist, the user shall not be limited by the communication protocol. Each of the ubiquitous devices is conformed to different communication protocols. The second is frequently movement of devices and user. This paper proposed the operation recommendation approach providing 'seamless' service to user. In order to support 'seamless' service, a list of alternative services is composited services, which have same functionality and are satisfied the user requirements. In this paper we propose an enhanced service substitution method of service brokering approach (universal service broker: US-Broker) in [2]. The approach addresses interoperability and composition between heterogeneous ubiquitous services. The basic concept of US-Broker composition is to derive a mapping between the target operation that should be substituted and substitute operation that offers similar functionality through a different interface. In addition, [2] represents the operation grouping information grouped by operation functionality. If target operation and another operation are belongs to same operation group, the operation is substitute operation. It is very simple, but it does not guarantee substitute service is executable because it does not consider I/O message structure and I/O parameter type. Furthermore, it does not take account of replacement costs or operation position. To solve this problem, we suggest the method for service recommendation of semantic based service substitution in ubiquitous computing. Our approach can provide a list of ranking services considering replacement cost, degree of satisfaction of user requirements, availability, execution time of operation, and execution cost, and location of user and operation. To make ranking service list, we defined the weights about user requirement. The rest of the paper is structured as follows. Next section introduces related work. Third section discusses our approach for substitution of stateful services. Finally, we present the conclusions of this study and further research related to this study.

## 2. Quality Criteria for Elementary Services

We define the Quality-of-Services (QoS) as a set of perceivable characteristics expressed in user-friendly language with quantifiable parameters that may be subjective or objective. The characteristics of quality and their parameters are based on the user or client requirements. In addition, we consider the where the service is. The user decides whether the location is important or not. Our system makes the candidate service list according the user decision.

We consider four generic quality criteria for elementary services:

- execution time: the execution time of a operation is defined as the time spent by the system execution that operation, including the time spent execution run-time and the time spent lifting and lowering schema mapping. The execution time is computed using the expression
$$q_{time}(op) = T_{process}(op) + T_{mapping}(onto, op), \qquad \textbf{(1)}$$
meaning that the execution time is the sum of the processing time $T_{process}(op)$ and the schema mapping time $T_{mapping}(onto, op)$. The schema mapping time is estimated based on executions of lifting and lowering schema mapping, i.e.,
$$T_{mapping}(onto, op) = T_{lifting}(onto, o) + T_{lowering}(o, i), \qquad \textbf{(2)}$$
where $T_{lifting}(onto, o)$ is lifting schema mapping time between operation group ontology *onto* and operation output parameter *o*, and $T_{lifting}(onto, o)$ is lowering schema mapping time between operation *onto* and operation input parameter *i*.
- execution price: Given an operation *op* of service, the execution price $q_{pr}(op)$ is the fee that an operation requester has to pay for invoking the operation $F_{invoke}(op)$. In addition, it includes the fee to use software $F_{invoke}(soft)$ which supports schema mapping.
$$Q_{pr}(op) = F_{invoke}(op) + F_{invoke}(soft) \qquad \textbf{(3)}$$
- successful execution rate: The successful execution rate $q_{rat}(op)$ of an operation *op* is that how many the operation is completed without failed or error occurred. The value of the success rate is computed from data of past invocations using the expression
$$q_{rat}(op) = N_{completed}(op) / T, \qquad \textbf{(4)}$$
where $N_{completed}(op)$ is the number of times that the operation *op* has been successfully completed, and *T* is the total number of invocations.1

- use frequency: Given an operation *op* of service, the use frequency $q_{fr}(op)$ is that how many the operation is invoked $N_{selection}(op)$ by the operation requester. The expression is

$$q_{fr}(op) = \sum_1^n N_i(onto) \,/\, N_{selection}(op, onto), \qquad \textbf{(5)}$$

where $\sum_1^n N_i(onto)$ is the total number of invoked operation in same operation group ontology *onto*.

Our system makes the recommend operations list used by the weight of user requirement. The users are able to decide that how much each qos criteria are important. Our system supports ranked operation list that is the result to calculate with weighted qos criteria. The ranked operation list is made by the expression

$$Q = \left(q_{time}(op) \times W_{time}(op)\right) + \left(q_{pr}(op) \times W_{pr}(op)\right) + \left(q_{rat}(op) \times W_{rat}(op)\right) + \left(q_{fr}(op) \times W_{fr}(op)\right) \qquad \textbf{(6)}$$

where $W_{time}(op)$ is the weight of execution time that is the value entered by user. For example, there are two candidate operations (Table 1). The candidate operation B is free, many users used this operation. The candidate operation C is faster than candidate operation B, but user has to pay using this operation. The user, who needs seamless services now, prefers a good software performance. In addition, this figure shows the calculated result according to expression (6). Although many users had used the operation B, operation C is fit to current user, who wants software which supports good performance.

[Table 1] Example of candidate operations

| QoS criteria | Candidate operation B | Candidate operation C | Weight |
|---|---|---|---|
| Execution time | 0.15 | 0.5 | 4 |
| Execution price | 0 | 0.4 | 1 |
| Successful execution rate | 0.9 | 0.9 | 0 |
| Use frequency | 0.4 | 0.2 | 1 |
| Q | 1.0 | 2.6 | |

## 3. Conclusions

This paper proposed the approach for the operation recommendation considering the ubiquitous computing environment. This approach is based a Semantic Web technique. In addition, we investigated ubiquitous in the environment QoS(quality of services) that it has been used to research. We identified QoS of the existing ubiquitous computing environment that affects the user based on the substitution of alternative operations execution time, execution cost, availability and location. In addition, we defined new QoS that is suitable in substitution and based on QoS list identified.

## 4. ACKNOWLEDGMENTS

This research was supported by the Global resource utilization project in science and technology from operation of KOSEN of the National Research Fundation(NRF) funded by the Ministry of Science, ICT & Future Planning(N-14-NM-IR09)

## 5. References

[1] Chriristina Brodersen, Susanne Bødker and Clemens Nylandsted Klokmose, “Ubiquitous Substitution,” Human Computer Interaction 07, p179-192, 2007
[2] Hyung-Jun Yim, Yun-Young Hwang, Kyu-Chul Lee, “Task-based Services Composition for Ubiquitous Computing”, SocialCom2011, 2011
[3] N.Ibrahim, F.Le Mouel and Frenot, “Semantic Service Substitution in Pervasive Environments,” International Journal of Services/Economics and Management, 2010
[4] Yehia Taher, Djamal Benslimane, Marie-Christine Fauvet, Zakaria Maamar, “Towards an Approach for Web Services Substitution,” IEEE IDEAS06, p.166-173, 2006
[5] Byungsun Jun, “SOA, What & How, “ WowBooks, 2008.03
[6] Liangzhao Zeng, Boualem Benatallah, “QoS-Aware Middleware for Web Services Composition,” IEEE Transaction on Software engineering, vol.30, p311-327, 2004
[7] X.Wang, T.Vitvar, M.Kerrigan, and I.Toma. “A QoS aware selection model for semantic web services,” In 4th International Conference on Service Oriented Computing (ICSOC 2006), p.390-401, 2006.
[8] Software Engineering-Product Quality-part 2: Internal Metrics. ISO/IEC TR 9126-2, 2003
[9] OASIS, “Web Services Quality Factors v1.0,” OASIS Working Draft, 07 June 2010