



HAL
open science

Prioritizing Consumer-Centric NFPs in Service Selection

Hanane Becha, Sana Sellami

► **To cite this version:**

Hanane Becha, Sana Sellami. Prioritizing Consumer-Centric NFPs in Service Selection. Advances in Conceptual Modeling - ER 2014 Workshops, ENMO, MoBiD, MReBA, QMMQ, SeCoGIS, WISM, and ER Demos, Oct 2014, Atlanta, United States. hal-03984740

HAL Id: hal-03984740

<https://hal.science/hal-03984740v1>

Submitted on 13 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Prioritizing Consumer-Centric NFPs in Service Selection

Hanane Becha¹ and Sana Sellami²

¹ Aix Marseille University, CNRS, LIF UMR 7279, 13288 Marseille, France

² Aix Marseille University, CNRS, LSIS UMR 7296, 13397 Marseille, France

hanane.becha@univ-amu.fr, sana.sellami@lsis.org

Abstract. Service Selection continues to be a challenge in Service Oriented Architecture (SOA). In this paper, we propose a consumer-centric Non-Functional Properties (NFP) based services selection approach that relies on an externally-validated set of NFP descriptions integrated with the Web Service Description Language (WSDL). Our approach is based on three steps: (1) a Filtering step based on Hard NFPs defined in the consumer's request, (2) a Matchmaking step to discover the functionally-equivalent services, and (3) a Ranking step that sorts the resulting set of services based on the Soft NFPs defined by the consumer. The evaluation of our proposed service selection approach shows that the prioritization of NFP usage enhances the performance time of the service selection process while satisfying the functional and the non-functional requirements of the consumer.

Keywords: Service Oriented Architecture (SOA), Web Services, Non-Functional Properties (NFPs), Service Selection.

1. Introduction

The proliferation of Web Services raises some challenges with respect to the ability of the service consumers to select the service most appropriate to their needs. This problem is referred to as the service selection problem and can be divided as the *Matchmaking* problem of discovering the Web services that deliver a given functionality and the *Non-Functional Properties (NFP)-based Ranking* problem of evaluating these services.

The Matchmaking is a tedious, very time consuming step [19]. The NFP-based Ranking is very often treated as an extra burden once the Matchmaking is performed or even simply scarified [18]. In fact, the current service description languages such as Web Services Description Languages (WSDL) do not contain all the elements required to handle NFPs service descriptions [6] impeding the automation of the latter process. However, Non-Functional Properties (NFPs) play an important role in each stage of the SOA process lifecycle since they can impact the functional aspect of composed services as conflicting NFPs might cause redesign [1]. Indeed, NFPs are of critical importance and, at times, functional requirements might be sacrificed to meet them. There is a considerable amount of research effort that has placed emphasis on

the importance of non-functional properties in service descriptions [1][2][3][8]. Hence, it seems worthwhile to get into account the consumer-centric NFPs even before performing the Matchmaking process. By consumer-centric (as opposed to provider centric), we mean the NFPs, once included in the service description, that can help the service consumer decide whether a given service suits best his needs. Exploring the NFP precedence in the service selection is the aim of this research effort that relies on a consumer-centric domain-independent, and externally validated NFP catalogue that was integrated into WSDL [8]. Our research hypothesize is that the prioritization of NFP usage will not only allow the selection of services that meet the user's non-functional requirements but can also improve the execution time of service discovery process. This paper is organized as follows. In Section 2, the related work is presented. Section 3 formalizes the proposed service selection approach. Section 4 details the validation method. Section 5 concludes and presents possible future work.

2. Related Work

NFP-aware service selection approaches in SOA can be classified into two categories. In the first category of service selection approaches, the available set of services is functionally heterogeneous. A Matchmaking procedure [16][17][18] is performed first to find the services that meet the required functionality. Then, an NFP-based Ranking step is performed to select the best services. The second category regroups most of the existing NFP-aware service selection solutions [1][5][12][13][14][15]. These solutions are based on the hypothesis of the availability of a collection of functionally-equivalent services, and do not address the Matchmaking. The essence of these service selection solutions is concerned with *ranking* the already discovered set of services, including eliminating those that do not meet a given NFP threshold defined by the consumer. These two service selection categories are illustrated in Fig. 1.

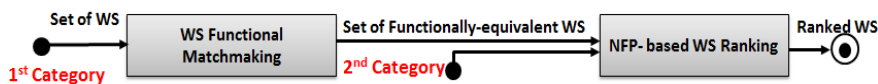


Fig. 1. Traditional approaches

In both categories, the service Ranking can be performed according to different methods [17] using either a unique NFP or a combination of a few NFP values having different weights. The combination of NFPs defines one representative NFP value usually using a Multiple Attribute Decision Making method [4]. However, these approaches have common shortcomings. The Ranking step is based either on NFPs that are defined only at the conceptual level (a.k.a. constraints) or defined based on a very limited set of non-validated NFPs with very simplistic data structures in an ad-hoc ways [7]. These solutions do not either have a clear focus on the service consumer's perspective or identify the publishing role of the service providers that should advertise the consumer-centric NFPs in their service descriptions. The service provider's publishing perspective is required, since the service consumers cannot just define on their own the NFPs that they are looking for at the service selection stage.

Many literature review studies such as [2] and [3] reveal that NFP-based consumer-centric service selection approaches are still needed. To the best of our knowledge, in all the reviewed papers, either the solutions rely on a pre-existing set of functionally-equivalent services or a Matchmaking is always performed before getting the NFPs into account. The NFP-based service Ranking is represented as an extra, separate, luxurious process after having gone (separately) through the tedious process of discovering functionally equivalent services [2] [19]. In opposition to the approach that prioritizes the NFPs, all these service selection approaches that prioritize the functional aspects are referred to in this paper as the traditional approaches, regardless the methods used in their Matchmaking and Web service Ranking processes.

3. NFP-based WS Selection Approach

```

- <NFPdescription>
- <price>
  <unit>EUR</unit>
  - <priceElements>
    <type>Invocation</type>
    <value>2</value>
  </priceElements>
  + <priceElements>
  </price>
- <responseTime>
  <average>4.0</average>
  <best>2.0</best>
  <confidence>95</confidence>
  <unit>microseconds</unit>
  <worst>7.0</worst>
</responseTime>
- <serverLocation>
- <countries>
  <code>CN</code>
</countries>
- <countries>
  <code>FR</code>
</countries>
</serverLocation>
</NFPdescription>

```

Fig. 2. Example of WSDL Instance Document with NFP Description [6]

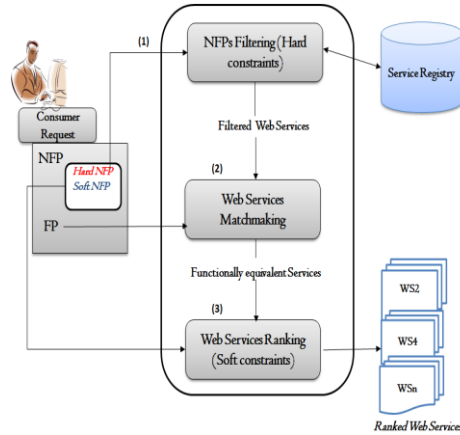


Fig. 3. The proposed Service Selection Approach

Our approach utilizes an externally-validated set of formal domain-independent catalogue of NFPs [6]. This catalogue defines a list of consumer-centric NFPs and their data structures to be published by service providers to better characterize services and enable consumers to perform NFP-aware service selection. The external validation details of this catalogue of NFPs (involving two surveys) can be found in [6]. This catalogue includes the following seventeen NFPs: (1) price, (2) response time, (3) reputation, (4) certification, (5) availability, (6) reliability, (7) usability, (8) accuracy, (9) standards compliance, (10) failure modes, (11) transactional service, (12) security, (13) jurisdiction, (14) service versioning, (15) resource requirements, (16) scalability, and (17) server location. The NFP catalogue description is integrated with the Web service description documents (e.g., WSDL documents) as shown by Fig. 2 and will be used as an additional input in our service selection process. The NFP description of the consumer's request respects a NFP schema provided in [6] and classifies the NFPs as Hard and Soft NFPs. Hard NFPs

are the properties that must be absolutely satisfied. Soft NFPs are the nice to have properties [4]. This framework is extensible to support other NFPs (i.e., domain-specific NFPs) without changing the algorithms and tools for NFP-based service selection.

3.1. Use Case

Fig. 4 illustrates a use case of our approach. The box at the top is an example of consumer request that details the required functionality, Hard and Soft NFPs. The values of the NFPs in the request are thresholds of the required values. The request contains the maximum acceptable value for Price and the minimum values for Reputation and Usability. The box below contains the description offered services. WS1 is representative of services that meet the consumer request in terms of functionality as well as Hard NFPs. However, WS2 fails to meet the request since it charges the consumer *per subscription* and in *USD*. WS3 is representative of services that do not match the functionality looked for by the consumer. In this use case, only the subset of services represented by WS1 and WS3 satisfy the user's request in terms of Hard NFPs. The consumer will perform the Matchmaking process on these two subsets of services and will only retain the subset represented by WS1.

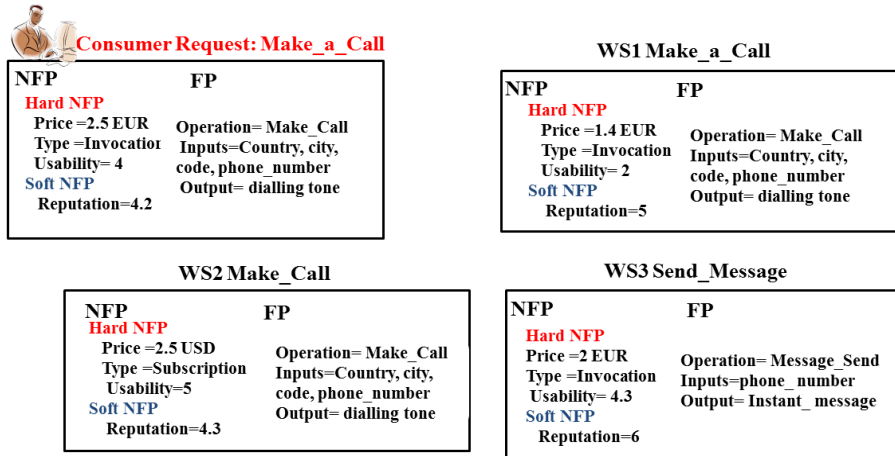


Fig. 4. Use Case of our Consumer-Centric Approach

Our proposed framework, illustrated by Fig. 3, is based on three different steps detailed below: *NFP Filtering*, *Matchmaking*, and *Service Ranking*. The initial set of the available services (*ServiceOffer*) in the registry are heterogeneous. Their descriptions address the functional as well as the NFPs. The request (*Request*) of the user defines the Hard NFPs, the Soft NFPs and the desired functionality. The NFPs description of the services and the NFPs in the consumer's request respect the NFP schema of the used consumer-centric NFP catalogue. The service selection algorithm is detailed in Fig. 5.

Algorithm Selection (ServiceOffer, Request)
<p>Input: ServiceOffer: Set of available services Request: the consumer's Request σ: Matchmaking threshold tab[w]: Weights associated to the Soft NFPs</p> <p>Output: RS: Ranked set of services</p> <p>var:</p> <ul style="list-style-type: none"> - SF: Set of services that correspond to the hard NFP per the consumer's request - SM: Set of services with functionality corresponding to the consumer request <p>Begin{</p> <p>SF=\emptyset;</p> <p>For each Service in ServiceOffer</p> <p style="padding-left: 2em;">SF= Filtering (Service, Request);</p> <p>SM =\emptyset;</p> <p>For each ServiceSF in SF</p> <p style="padding-left: 2em;">SM= Matchmaking (ServiceSF, Request, σ);</p> <p>RS =\emptyset;</p> <p>For each ServiceSM in SM</p> <p style="padding-left: 2em;">RS= Ranking (ServiceSM, Request, tab[w]);</p> <p>return RS;}</p> <p>End</p>

Fig. 5. Algorithm for Consumer-centric NFP-based Service Selection

3.2. Step 1: Services Filtering

The first step of our approach is to compare the NFPs of the available heterogeneous Web services (*ServiceOffer*) in the registry against the absolutely required NFPs defined in the consumer's service request. It filters out those that do not meet the NFP threshold values and return the filtered services *SF* (Fig. 6). When the service consumer denotes the Hard NFPs, this means that these NFPs shall be explicitly exposed as part of the service description, in a formal format and meet certain values expressed by the request as well. For example, in terms of pricing, the service consumer may only be interested in pay-per-invocation services and may not be willing to pay for monthly subscription services. To avoid repetition, the Filtering algorithm, defined by Fig. 6, considers five representative NFPs including Price, ResponseTime, Reputation, Availability and Scalability (eliminating the NFPs that have similar data structures and consequently same Filtering rules). Price and Response Time NFPs have complex data structures and call external comparison functions ComparePrice and CompareResponseTime that are not further explained here due to the space limitations.

Algorithm Filtering (ServiceOffer, Request)
<p>Input: <i>ServiceOffer</i>: Set of available services <i>Request</i>: The consumer's Request</p> <p>Output: <i>SF</i>: Set of services that correspond to the hard <i>NFP</i> per the consumer's request</p> <p>Var: <i>-HardNFP</i>: hard NFPs defined in the request out of the used NFP catalogue that contains 17 NFPs</p> <ul style="list-style-type: none"> - <i>ComparePrice (Service.Price, Request.Price)</i>: function to compare the Price NFP of the offered service versus the Request. Price has a complex data structure - <i>CompareResponseTime (Service.ResponseTime, Request.ResponseTime)</i>: function to compare the ReponseTime NFP of the offered service versus the Request. ReponseTime has a complex data structure <p>Begin{</p> <pre> Boolean Valid = TRUE; For each (Service in ServiceOffer){ For each (HardNFP in Request){ // For the following NFPs, exact values or less are //better for consumer's satisfaction //Comparison of Price by invoking the ComparePrice //function. It returns TRUE if the Service is "too //expensive" compared to the Request. If (HardNFP==Price) If (ComparePrice (Service.Price, Request.Price)) Valid = FALSE; // CompareResponseTime returns TRUE if the Service is too //slow compared to the Request If (HardNFP == ResponseTime) If(CompareResponseTime (Service.ResponseTime, Request.ResponseTime)) Valid = FALSE; //For the following NFPs, exact values or higher are //better for consumer's satisfaction //Reputation is very similar to Usability NFP If (HardNFP == Reputation) If (Service. Reputation < Request.Reputation) Valid = FALSE; // Reliability is very similar to Availability If (HardNFP == Availability) If(Service.Availability< Request.Availability) Valid = FALSE; If (HardNFP == Scalability) If(Service.Scalability < Request.Scalability) Valid = FALSE;} If (Valid==TRUE) ADD(Service, SF); } return SF;}End </pre>

Fig. 6. Algorithm for Web Services Filtering based on the Consumer's Hard NFPs

3.3. Step 2: Web Services Matchmaking

The input of this process is the set of filtered services (SF) that meet the Hard NFPs per step 1. In this step, service Matchmaking process is performed to discover the services that offer similar functionality (SM) defined in the consumer's request. The Matchmaking can be based on different techniques [9][10]. The matchmaker [9] is used in our proposed approach. The Matchmaking techniques support partial matches and associate correspondence degrees that have to exceed a given threshold σ defined in the consumer's request. Partial matches enable the consumer to trade the functional aspect in favor of the prioritized NFPs.

3.4. Step 3: Web Services Ranking

The input of this step is a set of services that deliver the required functionality and meet the Hard NFPs (SM) defined in the consumer's request. This set is inherently sorted based on the correspondence degree of the Matchmaking. The essence of this step is to select the service that suits best the needs of the service consumer. To do so, the services are ranked (RS) based on the Soft NFPs that have different weights defined by the consumers to reflect its preferences over these properties. The Ranking can be performed as described in Preference based Universal Ranking Integration framework (PURI) [11].

4. Experiment Design and Evaluation

The aim of this experiment is to assess the impact of the prioritization of NFP usage on the execution time of service selection process.



Fig. 7. Our Web Service Selection Approach

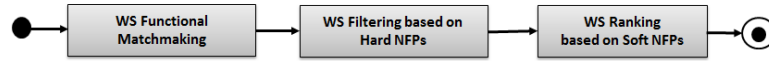


Fig. 8. Traditional Web Service Selection Approach

The experiments measure the total execution time of the Filtering and Matchmaking steps of our approach (Fig. 7) versus the approach that prioritizes the functional aspects (Fig. 8). The same Matchmaking and Filtering processes are invoked but in different order. Both approaches will select the services having the same quality. The Ranking process is not taken into account as it is the same in both approaches.

Experiments are performed on a test collection of 100 Web services¹ from the communication domain described using the language SAWSDL² (Semantic Annotations for WSDL). The Web services descriptions are extended using the NFPs catalogue [6] which defines a library of complex types as an XML schema that can be fully or partially populated and that apply to the whole service as one unit.

The Filtering process (Fig. 6) is implemented using the Xquery language and executed over the BaseX³ database. The consumer's request contains 5 NFPs out of the 17 proposed in the catalogue. Reputation and Reliability are the Soft NFPs. Price, Response Time and Availability are the Hard NFPs. We performed the Matchmaking procedure using the SAWSDL-MX1 matchmaker [9]. We considered five different scenarios that differ in terms of the number of eliminated services after having performing the NFP-based Filtering step.

Table 1 and Table 2 summarize the scenarios of our experiments. Table 1 represents our approach where the first performed process is the Hard NFP-based Filtering. Table 2 represents the approach where the first performed process is the semantic Matchmaking. The initial set of heterogeneous services in the WS registry contains 100 Web services. The first rows of both tables detail the number of the eliminated services after having performed the first process of each approach. Then, the execution time of the second process is measured for each scenario for both approaches. In the first scenario S1, the Filtering step does not eliminate any service (e.g., the entire set of services meet the consumer's Hard NFPs). The execution time of the Filtering process performed against the initial set of 100 WS is 256 ms. The execution time of Matchmaking performed against the initial set of WS is 2790 ms. Respectively, in the four following scenarios, the Filtering step eliminates 25%, 50%, 75%, and 100% of the available services. In last scenario, no service corresponds to the Hard NFPs of the consumer's request. The Hard NFPs have to be met even in the detriment of the functional requirements. As shown by the tables, the execution time of both processes varies as a function of the number of the services against which are performed.

Table 1. Our WS Selection Approach

	<i>S1</i>	<i>S2</i>	<i>S3</i>	<i>S4</i>	<i>S5</i>
Number of eliminated (filtered) WS	0	25	50	75	100
Execution Time of Matchmaking (ms)	2790	2340	1880	1760	0
Total Execution Time (ms)	3046	2596	2136	2016	256

Table 2. Traditional WS Selection Approach

	<i>S1</i>	<i>S2</i>	<i>S3</i>	<i>S4</i>	<i>S5</i>
Number of eliminated (unmatched) WS	0	25	50	75	100
Execution Time of Filtering (ms)	256	200	162	112	0
Total Execution Time (ms)	3046	2990	2952	2902	2790

¹ <http://www.semwebcentral.org/projects/sawSDL-tc/>

² <http://www.w3.org/TR/sawSDL/>

³ <http://basex.org/>



Fig. 9. Performance Comparison of WS Selection Approaches

Fig. 9 illustrates the results of our empirical experiment. In the scenarios described above, our selection approach is shown to be more efficient than the solution that prioritizes the functional aspects over the NFPs. Prioritizing the NFPs in the service selection enhances the performance time while satisfying the functional and the non-functional requirements of the consumer. In fact the Matchmaking process is very costly when compared to the Filtering process. The cost of Matchmaking performed against a single service is as costly as the Filtering step performed against the initial set of available services. The more irrelevant services that can be eliminated during the Filtering process, the faster the Matchmaking. Even when the Filtering step eliminates just a very few services before getting to the Matchmaking step, it is still worth to use our proposed approach.

However, we recognize the limits of our empirical experiment. In order to generalize our findings, further experimentations and/or statistic study are required to access the impact of the complexity of the consumer's request and the NFP descriptions over the time performance of each process. Part of the threats to our validation is the definition of 'extreme' scenarios where the number of Hard NFPs is higher, the data structures of NFPs are more complex (Security can have multiple sub parts), the NFP descriptions of the offered WS is larger, and the functionality looked for is simpler than the scenarios considered in our experimentation.

5. Conclusion and Future Work

In this paper, we presented a consumer-centric NFP-based approach for Web service selection that relies on an externally-validated set of NFPs. The proposed approach filters out of the services that do not meet the Hard NFPs defined by the consumer before even performing the Matchmaking. The evaluation of this service selection approach shows that the prioritization of NFPs usage can enhance the performance time of the service selection process while satisfying the functional and the non-functional requirements of the consumer.

In future work, we will plan to perform statistical study and test in different scenarios to further measure the performance of our approach versus the traditional one. In addition, the required extensions to enable BPEL to support the NFPs should

be identified to encourage NFPs usages in service selection and composition. As SOA is technology agnostic, it is interesting to test our approach in the context of REST services.

Acknowledgments. Special thanks to Prof. Daniel Amyot for his collaboration.

References

1. Klose, K., Knackstedt, R., Beverungen, D.: Identification of services - a stakeholderbased approach to SOA development and its application in the area of production planning. In: ECIS, University of St. Gallen (2007) 1802–1814
2. Hai, D., Farookh, K.H., Elizabeth, C.: Semantic Web Service matchmakers: State of the art and challenges. *Concurrency Computation Practice and Experience* (2012)
3. Teka Abeneh, Y., Nelly, C-F., Brahmananda, S.: A Systematic Literature Review on Service Description Methods. *REFSQ* (2012) 239-255
4. Bartalos, P., Bielikova, M.: Automatic dynamic web service composition: A survey and problem formalization. *Computing and Informatics*, Vol. 30, No.4, (2011) 793-827
5. Becha, H., Mussbacher, G., Amyot, D.: Modeling and Analyzing Non-Functional Requirements in Service Oriented Architecture with the User Requirements Notation. *Non-functional Properties in Service Oriented Architecture: Requirements, Models and Methods*. IGI Global, USA, (2011) 48-72
6. Becha, H.: Exposing and Aggregating Non-functional Properties in SOA from the Perspective of the Service Consumer. Ph.D. thesis, University of Ottawa, Canada (2012)
7. Becha, H., Amyot, D.: Non-Functional Properties in Service Oriented Architecture: A Consumer's Perspective. *Journal of Software*, Vol. 7, No. 3. Academy Publisher, (2012) 575-587
8. Becha, H., Amyot, D.: Consumer-Centric Non-functional Properties of SOA-Based Services. 6th International Workshop on Principles of Engineering Service-Oriented and Cloud Systems (PESOS) (2014), (To appear).
9. Matthias, K., Patrick, K., Ingo, Z.: Hybrid Adaptive Web Service Selection with SAWSDL-MX and WSDL-Analyzer. *Proceeding of the 6th European Semantic Web Conference, ESWC (2009)* 550-564
10. Sellami, S., Boucelma, O.: Towards a Flexible Schema Matching Approach for Semantic Web Service Discovery. *Proceeding of the IEEE 20th International Conference on Web Services (2013)* 611-612
11. María, G. J., Martin, J., David, R., Sudhir, A., Antonio, R. C.: Integrating semantic Web services ranking mechanisms using a common preference model. *Knowl.-Based Syst.* 49 (2013) 22-36
12. Askaroglu, E., Senkul, P.: Automatic QoS evaluation method for web services. *IEEE Symposium on Computers and Communications, ISCC (2012)* 367-369
13. Kritikos, K., Plexouakis, D.: OWL-Q for Semantic QoS-based Web Service Description and Discovery. *Proceedings of the SMRR 2007 Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web (SMRR 2007) co-located with ISWC 2007 + ASWC 2007 (2007)*
14. Shi, C., Lin, D., Ishida, T.: User-Centered QoS Computation for Web Service Selection. In *IEEE 19th International Conference on Web Services, ICWS (2012)* 456-463
15. Frutos, H. M., Kotsiopoulos, I., Gonzalez, L.M., Rodero-Merino, L.: Enhancing Service Selection by Semantic QoS. *Proceeding The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC (2009)* 565-577
16. Ajao, T. A., Deris, S.: Optimal Web Service Selection with Consideration for User's Preferences. *IJCSI International Journal of Computer Science Issues*, Vol. 10, Issue 2, No 3 (2013)
17. Lin, S-Y., Lai, C-H., Wu, C-H., Lo, C.: A trustworthy QoS-based collaborative filtering approach for webservice discovery. *The Journal of Systems and Software* (2014)
18. Huang, A- F. M., Lan, C-W., J. H. Yang S.: An optimal QoS-based Web service selection scheme. *Inf. Sci.* 179(19) (2009) 3309-3322
19. Mukpadhyay, D. Chougule, A.: A Survey on Web Service Discovery Approaches. *CoRR abs/1206.5582* (2012)