

Web Service Component Reusability Evaluation: A Fuzzy Multi-Criteria Approach

Aditya Pratap Singh and Pradeep Tomar

School of ICT, Gautam Buddha University, Greater Noida-201312, UP, India
E-mail: adityapsingh@gmail.com, parry.tomar@gmail.com

Abstract—The service oriented architecture supports reusable components. Component reusability is one of the important features while designing web services for reuse. The reusability is an ideal and key factor to improve the quality and production rate of software. It becomes very helpful for quality assurance, if such quality parameters can be quantified. Non functional quality parameters like reusability are not easy to measure and quantify. This paper attempts to quantify reusability using fuzzy multi criteria approach. This approach is considered due to the unpredictable nature of reusability attributes. For the estimation of reusability, the paper identifies 5 key attributes of reusability i.e. Coupling, Interface Complexity, Security, Response Time and Statelessness in context of web service components.

Index Terms—Software component, component reusability, web service, fuzzy multi criteria approach, coupling, interface complexity, security, response time, statelessness.

I. INTRODUCTION

A web service exploits the World Wide Web (WWW) infrastructure to provide the support for software application to communicate to other software application on the web. Web services use standardized eXtensible Markup Language (XML) messaging system. Web/software applications access web services using ubiquitous protocols like Hyper Text Transfer Protocols (HTTP), Simple Object Access Protocol (SOAP) without worrying about its implementation [1].

Web services are combination of Component-Based Software Development (CBSD) and Service-Oriented Architecture (SOA). The web services are application components as they are self contained and self describing and can be used by other applications. There are some services which are required very often in many applications, so why make these services again and again. By using spirit of CBSD, web services offer such application components e.g. language translation, weather report, Resume parser as service.

Web services are one of the best examples of software reusability. Web service reusability is one of the key criteria for measuring the Quality of Service (QoS). Web

services are used as a black-box reusable building block component in web application development [2]. The web service component exposes its functionality and services through its interface on-line. Web service components are published by developer over internet. The other developer can locate and invoke these web services programmatically over web after purchasing. Some of the web services are freely available. Web services uses SOA architecture and publish its services using XML based Web Services Description Language (WSDL) and can be used across platforms. This ability makes web service component reusability higher than other component types like class, component etc. Notwithstanding the research in the field of web services, a concrete metrics to compare reusability between web service components is still needed.

There may be multiple web service components having similar services, available in internet repository. The customers make decisions for selection of services without having prior knowledge about candidate services. Various available recommender systems help in service selection from repository [3]. The service component must have higher reusability to be selected for reuse.

This paper quantifies reusability using fuzzy multi criteria approach due to the unpredictable nature of reusability attributes. The paper identifies 5 key attributes of reusability i.e. Coupling, Interface Complexity, Security, Response Time and Statelessness in context of web service components for reusability estimation.

The rest of the paper is organized as follows. Section 2 presents the related work in the field of software component reuse and reusability metrics in the context of web service components. Section 3 identifies the factors for reusability estimation. Section 4 presents the fuzzy multi criteria approach. Section 5 presents the reusability estimation approach based on section 4. Section 6 deals with a case study in support to section 5. Finally the paper concludes in section 7.

II. RELATED WORK

A pervasive research has been done in the field of software reuse mainly in the context of Object-Oriented (OO) and Component-Based Software (CBS).

Researchers have been using OO metrics for

conventional software development to measure various quality attributes [4] [5] [6] [7] [8]. Most of the OO metrics works at code level and cannot be directly applied on CBS. Some of the reasons are discussed in Sharma et

al [9]. For CBS some of the reusability metrics in literature [10] [11] [12] [13] [14] [15] [16] considering specific features of components were proposed.

Table 1. Attributes for reusability measurement

	Sedigh-Ali, S. et al. (2001)	Washizaki, H. et al. (2003)	Boxall et al. (2004)	Rotaru O P. et al. (2005)	Gui and Scott (2006, May)	Bertoa, M. F. (2006)	Breivold, H. P. et al. (2007)	Sharma et al. (2009)	V. P. Venkatesan (2009)	Singh, Y. et al. (2011)	Hristov, D. et al. (2012)
Adaptability/ Customizability	√	√		√				√		√	√
Attractiveness						√					
Availability											√
complexity				√							√
composability				√							
Coupling					√		√		√		
Existence of Meta-Information/ Quality of Documentation		√							√	√	√
interface complexity	√		√	√				√	√	√	
Learnability						√					
Maintainability											√
Observability		√									
Operability						√					
Portability		√						√			
Statelessness							√				
Understandability		√	√			√		√		√	
Usability compliance						√					

A detailed mapping of various reusability attributes considered in literature is given in table 1.

In case of web services, there have been considerable research works done in the context of Quality of Service (QoS) [3] [17]. In the context of metrics for web services Ladan [18] classified metrics in two categories: Structural Metrics and Quality Metrics. The structural metrics concerns mainly for different types of coupling metrics. Sufficient literature is available for structural metrics. It is evident through literature [2] [4] [5] [7] [9] [10] [12] [13] that coupling impacts reusability. The paper considers coupling metric as a factor for reusability measurement.

The quality metrics refer to functional and non-functional properties of web services. It may consider reliability, integrity, performance, accessibility, interoperability and security [18]. Such quality factors also impacts decision of selecting a new component from repository.

III. IDENTIFICATION OF FACTORS FOR REUSABILITY ESTIMATION

Various factors were identified for measurement of reusability of black-box components by several researchers gradually in research literature [13] [16]. The table 1 lists the factors being used for reusability measurement.

Based on available literature [2] [13] [16] [18] the paper identifies following potential factors for reusability of component in the context of web service components:

- 1) Coupling
- 2) Interface Complexity
- 3) Security
- 4) Response time
- 5) Statelessness

The first two attributes are considered for web service reusability as these are most common attributes for reusability estimation of software components. The next attribute security is recently proposed by Koteska et al. [16] for component reusability estimation and also one of the quality concerns [18] so the paper selects security as a potential attribute for reusability estimation of web service components. Other two attributes found important in context of web service components [19] and were used as key attributes to measure performance of web services [20].

3.1. Coupling

The coupling is one of the traditional principals of software measurement. It is a most commonly used metric in various software development approaches like object-oriented and component-based software. Many coupling metrics were proposed so far but most of them are not specific for web service components. In case of web service components, coupling has a significant role in respect to reusability. The coupling of web service components is influenced by many aspects like if a web service component is completely independent, stateless and self-contained then the composition from such components will have low coupling.

In context of web service component, coupling is the degree of interactions between services [18]. In particular, web service components should have fewer considerations between web service implementation, its contract and its consumer. To keep low coupling web services have independent design and service logic at the same time keeps baseline interoperability with the consumers. The number of message exchanges/ service calls between services may be counted for measuring coupling.

3.2. Interface Complexity

In black box web service components there are input ports and output ports to receive input data and to provide processed/resultant data. Some components also provide a separate port for errors notification. The available input ports of a web service component act as formal parameters and accept data values at run time from actual parameters. In the same way output data is also returned by web service component through output port. This way web service component can return data concurrently with the complex service execution.

Web service components use Interface Definition Language (IDL) to define its interfaces as input ports. The IDL uses Distributed Computing Environment (DCE) and Remote Procedure Call (RPC), which allows the web service creation without worrying about different network and platforms. The quality of interfaces decides the attractiveness, reusability and marketability of web services.

The interface complexity can be measured by using interface complexity metric proposed in Singh and Tomar [26]. The metric quantifies the interface complexity by analyzing the parameters and return values of interface

methods and the weight values can be assigned on the basis of the data type of return value and formal arguments of interface method.

3.3. Security

Security is a composite factor that has integrity, confidentiality and availability of service. The integrity of web service component means that the service does not perform any untoward alteration to the application or any information. The confidentiality represents the ability of not disclosing any information to unauthorized user. The availability refers to the readiness of desired service component. The certain service momentary outage may not create big problems for non-critical applications but may cause disastrous consequences for safety critical applications.

The security measurement seeks attention for web services as you cannot secure what you cannot measure. For measurement of security depends on various factors like authentication, authorization, data confidentiality, encryption and non-repudiation. They can be estimated through existence metrics for security mechanisms with such factors.

3.4. Response time

The response time of any service is the time duration between the client sending request and receiving a response. In context of web services the response time is a key factor for selection of web service component by a user. The metric for response time of web services is still a topic of concern for researchers. Various researchers use Hidden Markov Model (HMM), Knowledge Base and Bayesian Network for prediction of response time of web service components [25]. Response time can be estimated directly through observer agent services/applications which can log response time directly. These observer agents can apply load and gather statistics for selected web services. By analyzing these results, the web service component with suitable response time can be simply identified.

3.5. Statelessness

The web service statelessness refers to the property of not maintaining state information by service itself. The stateless service components are more efficient for most uses and have very reduced complexity. Web services API that adhere to the Representational State Transfer (REST) architectural constraints are called RESTful APIs. According to Pautasso [26] Statelessness is one of the pillars of RESTful web services. The stateless web services are lesser in size than SOAP web services and provide considerable level of scalability and fault tolerance. Due to all these features this attribute is selected as a key attribute for web service component reusability estimation.

For estimation of statelessness the paper suggests to use the existence metric which can take value 1 or 0 according to whether the web service component has statelessness property or not.

IV. FUZZY MULTI CRITERIA APPROACH

Multi-criteria decision making (MCDM) comprises a finite set of alternatives, amongst which the decision-makers select, evaluate or rank according to the weights of a finite set of attributes.

4.1. Fuzzification

Triangular fuzzy membership function is adopted in this paper. The fuzzy sets are represented as fuzzy membership function $\mu(z)$ as shown in figure 1. It is a graphical representation of degree of participation of inputs. According to Zadeh [21] the membership function of fuzzy triangular number is expressed as equation 1.

$$\mu(z) = \begin{cases} \frac{z-L}{M-L}, & L \leq z \leq M \\ \frac{U-z}{U-M}, & M \leq z \leq U \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The process of converting real time problem into fuzzy sets is called fuzzification. The crisp values are transformed in to grades of membership corresponding to fuzzy sets in linguistic terms.

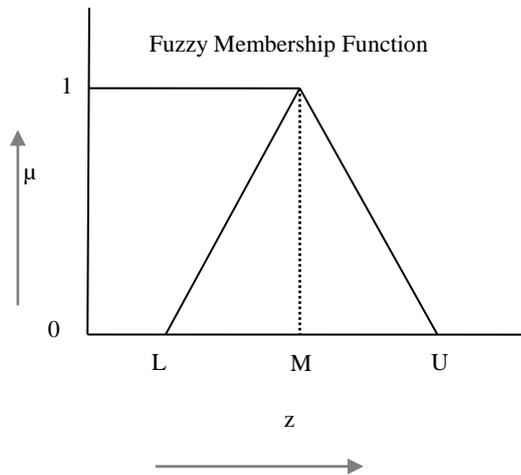


Fig.1. The Fuzzy Membership Function of triangular fuzzy number

For evaluating component reusability based on selected attributes, the triangular fuzzy sets used to represent reusability metric. For every metric, there is a particular rating and weight for different criteria.

The rating and weight of all selected attributes are fuzzified in to triangular fuzzy sets. Five fuzzy sets in the form of linguistic weighting variables, which include “Very High (VH)”, “High (H)”, “Medium (M)”, “Low (L)” and “Very low(VL)”, were utilized to evaluate the importance of input variables. These variables are equivalent to fuzzy numbers on the numeric scale 0–1.

The triangular fuzzy sets for rating and weight are listed in table 2 and table 3. The values of linguistic variables for weights and rating are adopted form [22] and [23]. These papers used same representation.

Table 2. Linguistic variables for importance weights of Metrics

Importance of Criteria	Corresponding TFNs
Very High (VH)	(0.7, 1.0, 1.0)
High (H)	(0.5, 0.7, 1.0)
Medium (M)	(0.2,0.5,0.8)
Low (L)	(0.0,0.3,0.5)
Very Low (VL)	(0.0,0.0,0.3)

Table 3. Linguistic variables for possible rating

Importance of Criteria	Corresponding TFNs
Very High (VH)	(0.7, 0.9, 1.0)
High (H)	(0.5, 0.7, 0.9)
Medium (M)	(0.3,0.5,0.7)
Low (L)	(0.1,0.3,0.5)
Very Low (VL)	(0.0,0.1,0.3)

4.2. Fuzzy Operations

The component reusability is evolved using the weighted average technique of fuzzy sets. The extension principle by Zadeh[21] is adopted for fuzzy operations in this paper. The addition and multiplication of triangular fuzzy numbers are performed as follows:

Let $R= (L_1, M_1, U_1)$ and $S= (L_2, M_2, U_2)$ are two triangular fuzzy numbers.

$$R+S= [\max(L_1, L_2), \max(M_1, M_2), \max(U_1, U_2)] \quad (2)$$

$$R \times S= (L_1 \times L_2, M_1 \times M_2, U_1 \times U_2) \quad (3)$$

4.3. Defuzzification

The process of converting fuzzy output to scalar quantity or crisp data is called Defuzzification [21]. The input for defuzzification is the aggregate output fuzzy set and the output is a singular number. The Centroid method [21], which returns the centre of gravity under the curve, is adopted in this paper for defuzzification process of output fuzzy set. It can be expressed as:

$$z^* = \frac{\int \mu(z).z.dz}{\int \mu(z).dz} \quad (4)$$

Here z^* is the defuzzified output, $\mu(z)$ is the membership function and z is the value on x-axis.

V. REUSABILITY EVALUATION

In order to evaluate web service component reusability using fuzzy multi-criteria approach, it is important to make certain assumptions for the simplicity and convenience in quantifying the component reusability using fuzzy sets.

- i. The values of all the characteristics have been quantified in the range of 0 to 1.
- ii. The component reusability after quantification will be in range of 0 to 1.
- iii. The weight considered in this paper may vary from case to case.
- iv. The fuzzy weighted average of all the quantified criteria is taken in order to arrive at the final reusability value. This has been done to maintain consistency so that the range of final values lies between 0 and 1.

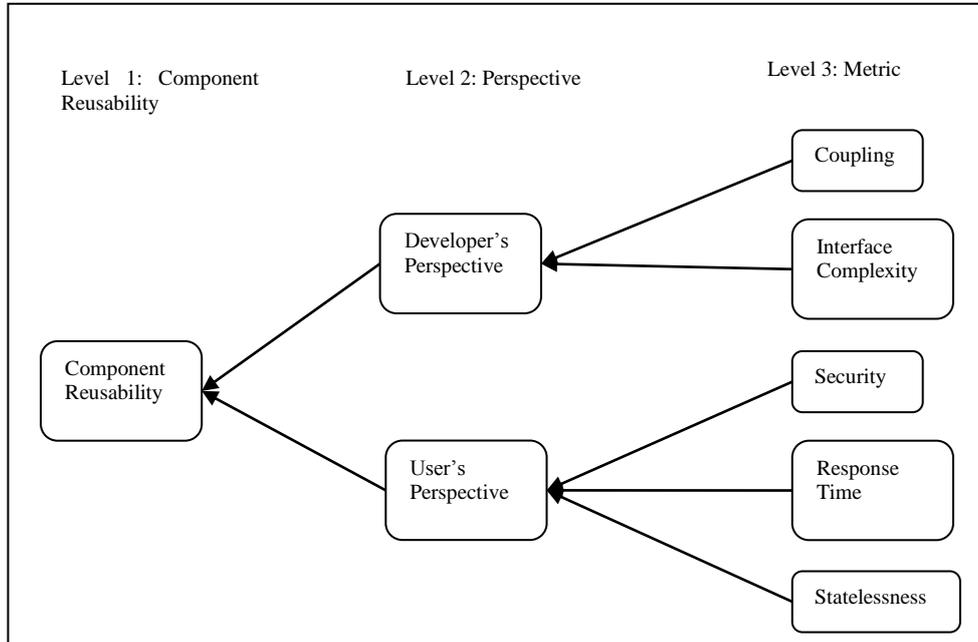


Fig.2. Evaluation hierarchy process of the proposed reusability evaluation

To quantify component reusability the selected attributes are used. The selected attributes can be sub divided in to perspective as shown in figure 2.

The procedure to quantify component reusability is as follows:

Step 1: Assign fuzzy ratings to every metric that is selected for web service component reusability evaluation and perspectives.

Step 2: Assign fuzzy weights to selected attributes and perspectives.

Step 3: Take weighted average of each perspective at level 2 as shown in figure 2.

Step 4: Take final weighted average of component reusability at level 1.

Step 5: The resultant fuzzy number in step 4 is to be defuzzified using equation (4) for centroid formula to get crisp value for component reusability.

The perspective fuzzy rating is evaluated by weighted average of attributes affecting it:

$$Perspective\ Rating = \sum_{i=1}^m r_i \times w_i \quad (5)$$

where i belongs to the set of attributes applicable for that perspective and m is the number of attributes.

Now the fuzzy rating of component reusability can be evaluated as:

$$r_{reusability} = r_d \times w_d + r_u \times w_u \quad (6)$$

where d represents Developer’s and u represents User’s rating and weight.

This paper considers the required inputs of ratings and weights at different levels through a group of practitioners view (Developers and Users).

In this approach, the component reusability is evaluated in respect to developer’s perspective and user’s perspective. The selected attribute are divided to relative perspective as shown in figure 2.

5.1 Determining possible rating of selected factors

The evaluators ($E_i, i = 1 \dots m$) are invited to express their judgement regarding the possible rating of each selected factor ($F_j, j = 1 \dots n$) by employing linguistic numbers as shown in table 3.

The decision metrics \tilde{X} is defined as:

$$\tilde{X} = \begin{matrix} & E^1 & \dots & E^m \\ F_1 & \left[\begin{matrix} \tilde{X}_1^1 & \dots & \tilde{X}_1^m \end{matrix} \right] \\ \vdots & \left[\begin{matrix} \vdots & \vdots & \vdots \end{matrix} \right] \\ F_n & \left[\begin{matrix} \tilde{X}_n^1 & \dots & \tilde{X}_n^m \end{matrix} \right] \end{matrix}, i=1,2,\dots,m; j=1,2,\dots,n \quad (7)$$

where m and n are number of evaluators and selected factors respectively. \tilde{X}_j^i indicates the possible fuzzy rating by i^{th} evaluator for j^{th} factor.

The decisions of different evaluators are aggregated to obtain possible fuzzy rating of factors. This paper applies the average value method to get consolidated fuzzy judgment of m evaluators, as:

$$\tilde{r}_j = \frac{1}{m} [\sum_{i=1}^m \tilde{X}_j^i] \tag{8}$$

where $\tilde{r}_j = (Lr_j, Mr_j, Ur_j)$ represents synthesized fuzzy possible rating of j^{th} selected factor.

5.2 Determining the importance weights of selected factors

The selected factors for component reusability evaluation have different meanings, not all of which are equally important. The idea that the importance weight of factors is more realistic and straightforward for evaluators to suggest weights according how much that factor influences the reusability evaluation. The evaluators expressed their preferences regarding the importance weights of various selected factors in the form of simple linguistic variables parameterized using triangular fuzzy numbers.

For this a decision matrix \tilde{Y} is constructed for the importance of weights of selected factors ($F_j, j = 1 \dots n$). The evaluators ($E_i, i = 1 \dots m$) are asked to give subjective opinion about weight of particular factor in component reusability evaluation in the form of linguistic variable.

$$\tilde{Y} = \begin{matrix} F_1 \\ \vdots \\ F_n \end{matrix} \begin{bmatrix} \tilde{Y}_1^1 & \dots & \tilde{Y}_1^m \\ \vdots & & \vdots \\ \tilde{Y}_n^1 & \dots & \tilde{Y}_n^m \end{bmatrix}, i=1,2,\dots,m; j=1,2,\dots,n \tag{9}$$

where m and n are number of evaluators and selected factors respectively. \tilde{Y}_j^i indicates the possible fuzzy weight importance by i^{th} evaluator for j^{th} factor.

Since the opinion of individual evaluators vary in according to their experience, intuition or knowledge, this paper applies the average score method to integrate the fuzzy weight values of m evaluators, namely

$$\tilde{w}_j = \frac{1}{m} [\sum_{i=1}^m \tilde{Y}_j^i] \tag{10}$$

where $\tilde{w}_j = (Lw_j, Mw_j, Uw_j)$ represents synthesized fuzzy weight values of j^{th} selected factor.

VI. CASE STUDY (TEMPCONVERT)

This paper evaluates the component reusability based on developed approach for a real world web service component TempConvert, available openly on W3schools [24]. This web service can be used for temperature conversion. It provides two types of operations i.e. CelsiusToFahrenheit and FahrenheitToCelsius.

Such freely distributed web services can be remotely invoked using HTTP requests. The web services expose their functionality over network and this exposed functionality of web service can be used by other applications. The user of a web service can utilize this existing web service by opening a network connection and sending request. The one of two methods provided by TempConvert web service can be invoked by simply

passing web service address with desired function in form action tag. The required parameter can be passed using input tag and submit button in simple HTML.

This section clearly explains the procedure of evaluation of component reusability with respect to developer and user perspectives and then finally combines them to get the final component reusability.

6.1 Results and Analysis

The selected case study is used to validate the proposed approach of reusability evaluation. The proposed approach is applied to evaluate reusability of TempConvert web service.

As a first step the identified 5 factors are assigned fuzzy rating in respect to TempConvert web service by 6 evaluators from a group of practitioners who are developing applications using web services. They were asked to provide the ranking for each attribute and the values are fuzzified. The assigned fuzzy ratings by different evaluators are shown in table 4.

The fuzzified value of the ratings form evaluators are consolidated as per equation (8). These consolidated ratings are also listed in table 4.

Table 4. Fuzzy numbers for selected factor's rating

	E ₁	E ₂	E ₃	E ₄	E ₅	E ₆	\tilde{r}_j
F ₁	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.0, 0.1, 0.3)	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)	(0.15, 0.33, 0.53)
F ₂	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.17, 0.37, 0.57)
F ₃	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.6, 0.8, 0.95)
F ₄	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.0, 0.1, 0.3)	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)	(0.0, 0.1, 0.3)	(0.1, 0.27, 0.47)
F ₅	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.37, 0.57, 0.75)

Table 5. Fuzzy numbers for selected factor's weight

	E ₁	E ₂	E ₃	E ₄	E ₅	E ₆	\tilde{w}_j
F ₁	(0.7, 1.0, 1.0)	(0.5, 0.7, 1.0)	(0.7, 1.0, 1.0)	(0.7, 1.0, 1.0)	(0.5, 0.7, 1.0)	(0.5, 0.7, 1.0)	(0.6, 0.85, 1.0)
F ₂	(0.5, 0.7, 1.0)	(0.5, 0.7, 1.0)	(0.2, 0.5, 0.8)	(0.5, 0.7, 1.0)	(0.2, 0.5, 0.8)	(0.5, 0.7, 1.0)	(0.4, 0.63, 0.93)
F ₃	(0.2, 0.5, 0.8)	(0.0, 0.3, 0.5)	(0.0, 0.3, 0.5)	(0.2, 0.5, 0.8)	(0.0, 0.3, 0.5)	(0.0, 0.3, 0.5)	(0.06, 0.32, 0.57)
F ₄	(0.5, 0.7, 1.0)	(0.5, 0.7, 1.0)	(0.5, 0.7, 1.0)	(0.7, 1.0, 1.0)	(0.7, 1.0, 1.0)	(0.5, 0.7, 1.0)	(0.57, 0.8, 1.0)
F ₅	(0.0, 0.0, 0.3)	(0.2, 0.5, 0.8)	(0.0, 0.0, 0.3)	(0.0, 0.0, 0.3)	(0.2, 0.5, 0.8)	(0.2, 0.5, 0.8)	(0.1, 0.25, 0.55)

In the next step the fuzzy weight values in respect to the TempConvert web service for identified 5 factors are assigned as per their importance in reusability evaluation judged by 6 evaluators. The assigned fuzzy weight values are shown in table 5.

The assigned fuzzy weight values are integrated using average score method as per equation (10). These consolidated weights for each attribute are also listed in table 5.

In the next step Developer's perspective reusability has been evaluated by weighted average of the ratings of coupling and interface complexity factors using equation (5).

$$r_d = \tilde{r}_{j1} \times \tilde{w}_{j1} + \tilde{r}_{j2} \times \tilde{w}_{j2}$$

$$r_d = (0.21, 0.28, 0.53)$$

The User's perspective reusability is also evaluated by weighted average of ratings of Security, Response Time and Statelessness using equation (5).

$$r_u = \tilde{r}_{j3} \times \tilde{w}_{j3} + \tilde{r}_{j4} \times \tilde{w}_{j4} + \tilde{r}_{j5} \times \tilde{w}_{j5}$$

$$r_u = (0.06, 0.26, 0.54)$$

Now with the help of fuzzy ratings of reusability for developer and user perspective the net reusability is evaluated using equation (6) and the weights assigned to different perspective shown in table 6.

$$r_{reusability} = (0.04, 0.18, 0.54)$$

Table 6. Fuzzy weights for different perspective

Name of Perspective	Weights	Fuzzy Values
Developer	M	(0.2,0.5,0.8)
User	H	(0.5, 0.7, 1.0)

As per step 5, after evaluating the final reusability fuzzy value, this value is defuzzified using centroid method as equation (4).

$$r_{reusability} = 0.253$$

On applying the centroid method (4) on fuzzy value of $r_{reusability}$ gives the crisp value of 0.253 as final reusability of TempConvert web service. In this way the reusability of a black-box component can be estimated and quantified.

VII. CONCLUSION

Improving software quality, reducing software development cycle, decreasing development cost and improving competitiveness are generally cited as key objectives for software quality improvement initiatives. Since component reusability adds such benefits to

software development, the reusability metrics and its quantified value may ease the process of quality evaluation. This paper proposed an approach to quantify the component reusability. This paper first identifies the five key attributes for reusability estimation. The measuring methods of identified attributes are also discussed. For estimation of web service component reusability as a crisp value the fuzzy multi criteria approach is used. The paper first determines the possible ranking and importance weights of selected attributes for a real world freely available web service component. The paper successfully quantifies the selected component reusability by applying fuzzy multi criteria approach. With the help of proposed approach the non functional quality attribute like reusability can be easily quantified. In future the proposed approach can be applied to reusability estimation for different context other than web service context.

REFERENCES

- [1] Richardson, L., Ruby, s.: RESTful web services. O'Reilly Media, Inc., (2008).
- [2] Qian, K., Liu, J., Tsui, F.: Decoupling metrics for services composition. 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse. (pp. 44-47). (2006, July).
- [3] Hasan, M. H., Jaafar, J., Hassan, M. F.: Fuzzy-based Clustering of Web Services' Quality of Service: A Review. Journal of Communications, Vol. 9 No. 1. (2014).
- [4] Chidamber, S. R., Kemerer, C. F.: A metrics suite for object oriented design. IEEE Transactions on Software Engineering, Vol. 20 No. 6, 476-493. (1994).
- [5] Basili, V. R., Briand, L. C., Melo, W. L.: A validation of object-oriented design metrics as quality indicators. IEEE Transactions on Software Engineering, Vol. 22, No. 10, 751-761. (1996).
- [6] Abreu, F.B., Carapuça, R.: Candidate Metrics for Object-Oriented Software within a Taxonomy Framework, Journal of Systems and Software, Vol. 26, No. 1, 87-96. (1994).
- [7] Briand, L. C., Daly, J. W., Wust, J. K.: A Unified Framework for Coupling Measurement in Object-Oriented Systems, IEEE Transaction on Software Engineering, Vol. 25, No. 1, 91-121. (1999).
- [8] Briand, L. C., Morasca, S., Basili, V. R.: Property-based Software Engineering Measurement, IEEE Transactions on Software Engineering, Vol. 22, No. 1, 68-86. (1996).
- [9] Sharma, A., Kumar, R., Grover, P. S.: A critical survey of reusability aspects for component-based systems. World academy of science, Engineering and Technology, Vol. 33, 35-39. (2007).
- [10] Washizaki, H., Yamamoto, H., Fukazawa, Y.: A metrics suite for measuring reusability of software components, In Proceedings of IEEE Ninth International Software Metrics Symposium, 211-223. (2003).
- [11] Boxall, M. A., Araban, S.: Interface metrics for reusability analysis of components, In Proceedings of IEEE Australian Software Engineering Conference, 40-51. (2004).
- [12] Rotaru, O. P., Dobre, M.: Reusability metrics for software components, In 3rd ACS/IEEE International Conference on Computer Systems and Applications, 24-29. (2005).
- [13] Gui, G. Scott, P. D.: Coupling and cohesion measures for evaluation of component reusability, In Proceedings of

- ACM International workshop on Mining software repositories, 18-21. (2006).
- [14] Gill, N. S.: Reusability issues in component-based development, ACM SIGSOFT Software Engineering Notes, Vol. 28 No. 6, 30-33. (2003).
- [15] Yingmei, L., Jingbo, S., Weining, X.: On Reusability Metric Model for Software Component, In Wu, Y. (eds.) Software Engineering and Knowledge Engineering. LNCS, Vol. 114, 865-870. Springer, Heidelberg. (2012).
- [16] Koteska, B. Velinov, G.: Component-Based Development: A Unified Model of Reusability Metrics, In ICT Innovations, 335-344. Springer Berlin Heidelberg. (2013).
- [17] Li, L., Rong, M., Zhang, G.: A web service qos prediction approach based on multi-dimension qos, In 6th International IEEE Conference on Computer Science & Education (ICCSE), 1319–1322. (2011).
- [18] Ladan, M. I.: Web services metrics: A survey and a classification. Journal of Communication and Computer, Vol. 9, No. 7, 824-829. (2012).
- [19] Rodriguez, A.: Restful web services: The basics. IBM developerWorks. (2008).
- [20] Potti, P. K., Ahuja, S., Umopathy, K., Prodanoff, Z.: Comparing Performance of Web Service Interaction Styles: SOAP vs. REST. In Proceedings of the Conference on Information Systems Applied Research Vol. 2167, 1508. (2012).
- [21] Zadeh, L. A.: Fuzzy sets, Information and Control Vol. 8, No. 3, 338–353. (1965).
- [22] Tahriri, F., Mousavi, M., Haghghi, S. H., Dawal, S. Z. M.: The application of fuzzy Delphi and fuzzy inference system in supplier ranking and selection. Journal of Industrial Engineering International, Springer, Vol. 10, No. 3, 1-16. (2014).
- [23] Challa, J. S., Paul, A., Dada, Y., Nerella, V., Srivastava, P. R., Singh, A. P.: Integrated Software Quality Evaluation: A Fuzzy Multi-Criteria Approach. Journal of Information Processing Systems, Vol. 7, No. 3, 473-518. (2011).
- [24] <http://www.w3schools.com/webservices/tempconvert.aspx>. (Accessed on 27.04.15).
- [25] Atluri, V. V., Mohanty, H.: Web Service Response Time Prediction Using HMM and Bayesian Network. In Intelligent Computing, Communication and Devices, 327-335. Springer India. (2015).
- [26] Singh, A., Tomar, P.: Estimation of Component Reusability through Reusability Metrics. World Academy of Science, Engineering and Technology, International Science Index 95, International Journal of Computer, Information, Systems and Control Engineering, Vol. 8, No. 11, 1729 - 1736. (2014).
- [27] Pautasso, C.: RESTful web services: principles, patterns, emerging technologies. In Web Services Foundations, 31-51. Springer New York. (2014).

national and international conferences. His work has appeared in IEEE Explore. He has served in program committee relating to national conferences on cyber security issues.



Dr. Pradeep Tomar is working as Faculty Member in the School of Information and Communication Technology, Gautam Buddha University, Greater Noida, INDIA since 2009. Dr. Tomar earned Ph.D. from Department of Computer Science & Applications, M. D. University, Rohtak, Haryana, INDIA.

He is also a member of IEEE, IEEE Computer Society, Computer Society of India (CSI), Indian Society for Technical Education (ISTE), Indian Science Congress Association (ISCA), International Association of Computer Science and Information Technology (IACSIT) and International Association of Engineering (IAENG). He served as a reviewer of journals and conferences and worked as advisory board members in national and international conferences.

Two books “Teaching of Mathematics” and “Communication and Information Technology” at national levels have been authored by Dr. Tomar.

Dr. Tomar has been awarded with Bharat Jyoti Award by India International Friendship Society in the field of Technology in 2012. He has been awarded for the Best Computer Faculty award by Govt. of Pondicherry and ASDF society.

His biography is published in Who’s Who Reference Asia, Volume II. Several technical sessions in national and international conferences had been chaired by Dr. Tomar and he delivered expert talks in FDP, workshops, national and international conferences. Three conferences have been organized by Dr. Tomar.

How to cite this paper: Aditya Pratap Singh, Pradeep Tomar, "Web Service Component Reusability Evaluation: A Fuzzy Multi-Criteria Approach", International Journal of Information Technology and Computer Science(IJITCS), Vol.8, No.1, pp.40-47, 2016. DOI: 10.5815/ijitcs.2016.01.05

Authors' Profiles



Aditya Pratap Singh received his Master of computer application degree from Uttar Pradesh Technical University, Lucknow in 2003. He is perusing PhD from Gautam Buddha University, Greater Noida. He is assistant professor in the Department of MCA at Ajay Kumar Garg Engineering College, Ghaziabad. His current research interests are in component-based software engineering and software measurement. He has presented his work in several