Strength Estimation of Relation between Metrics in Evolution based Model

G. Shanmugasundaram
Research Scholar
Department of Banking Technology
Pondicherry University

V. Prasanna Venkatesan
Associate Professor
Department of Banking Technology
Pondicherry University

C. Punitha Devi
Research Scholar
Department of Computer Science & Engineering
Pondicherry University

ABSTRACT
Object oriented system was the programming paradigm which aimed at the concept of software reuse. This reuse concept which has obtained its significance there upon needed to be strengthened in software systems and design concepts. This paved the basic idea behind evolving of software design paradigms into component and service oriented systems respectively. An evolution based model had been formed based on a template designed to study and record how the metrics are categorised between the three systems. This paper projects the improvement done over the model in order to relate the metrics quantitatively. The maturity level of reuse metrics stated through the evolution based model is established by bringing out the strength of the relationship that is estimated through the study.

General Terms
Object Oriented System, Component based System and Service Oriented System.

Keywords
Software Reusability metrics, OO, Component and Service Reusability metrics, Evolution model, Strength of relations

1. INTRODUCTION
Software Reuse allows the use of existing software artifacts or the use of knowledge for creating new software instead developing from the scratch (i.e. which is already developed, implemented and tested by other developers). Reusing the software is one of the key methods which significantly increase the software productivity. For the programming paradigms like object-oriented, component based, aspect oriented and service oriented, reusability acts as a main constituent, which reduce the software development cost and increases the quality of the software. Measuring reusability enables one to analyze the reduction in cost or development time when building any software. Metrics for reusability in object oriented system have been stated by [2][3][4][5], for component based system have been addressed by [11][14][15] and similarly for service oriented system [42][43].
There exists various metrics for measuring reusability of object oriented system and component based system. For service oriented system the metrics lie in the immature state. The work done [55] previously states that the metrics of one system have great influence on metrics of the others system either directly or indirectly. This concept has been brought through an evolution based model which clearly says how metrics have evolved from object oriented system to component based system and from there to service oriented system.

The objective of this paper is to strengthen the evolution based model [53] by stating measures and relations from observation made through the study. The paper is organized into 4 sections, where section 2 briefs the works done previously towards reusability metrics for the three systems considered. Section 3 details the proposed work of strength evaluation process and section 4 gives the findings of the proposed work and finally the conclusion.

2. RELATED WORKS
Literature shows that several works have been carried out towards building metrics and models for software reusability. As reusability has been addressed widely by these three systems object oriented, component based and service oriented systems this section briefs the work done on metrics for reusability.

Object Oriented System (OOS)

In object oriented system, the reusability is estimated by measuring complexity, understand ability, testability and Portability as given below.

Rajaraman and Lyu [1] proposed four coupling metrics, intended primarily for C++ software, which could be extended to other object-oriented languages. These metrics were to be used in calculating reusability and maintainability.

Chidamber and Kemerer [2] proposed the design metrics such as DIT, WMC, NOC, RFC, LCOM, and CBO which assess different characteristics of the Object Oriented Development. It also supports in calculating reusability, maintainability and complexity of OO systems.

Bieman and kang [3] developed a set of functional cohesion measures based on program slices and has used it to evaluate the reuse from the server perspective

Young Lee and Kai H. Chang [5] proposed a quality model for object-oriented software and an automated metric tool, Reconfigurable Automated Metrics for Object-Oriented Software (RAMOOS) targeted at the maintainability and reusability aspects of software which can be effectively predicted from the source code.

El Emam [6] uses the lines of code (LOC) to measure the complexity of the OOS.

C. Neelamegam et.al, [8] propose the **QMOOD** (Quality Model for Object-Oriented Design) a comprehensive quality model that establishes a clearly defined and empirically validated model to assess OOD quality attributes such as understandability and reusability, and relates it through

mathematical formulas, with structural OOD properties such as encapsulation and coupling.

Component Based System (CBS)

The following are the significant works on component-based development in measuring reusability of the system.

Cho et al [9] proposes a set of metrics for measuring various aspects of software components like complexity, customizability and reusability.

Washizaki et al. [11] proposes quality factors for measuring the reusability of black-box components; understandability, adaptability and portability. The work defines several metrics for measuring the quality attributes.

Rotaru's work [14] proposes reusability metrics for software components; composability and adaptability. Composability is mostly affected by complexity of component interfaces. Adaptability measures the conventional adaptability of components for the given context. The proposed metrics for reusability are applied only to software components.

Gui's work [15] proposes reusability metrics which focus on evaluating cohesiveness of components. Cohesiveness in this work reflects the degree of direct and indirect similarities between classes.

Luer's work [18] measures reusability with functionality and applicability. Functionality of a component is the number of situations in which a client project might want to use the component, based on its specification. Applicability measures the number of situation in which a module can be reused. This work presents a conceptual description on the quality attributes without metrics.

V. Lakshmi Narasimhan and B. Hendradjaya [19] define suites of metrics, which address static and dynamic aspects of component assembly. The static metrics measure complexity and criticality of component assembly and Dynamic metrics are useful to identify super-component and to evaluate the degree of utilization of various components.

V. Prasanna Venkataesan and M. Krishnamurthy [23] defined metrics to measure the component characters which includes three functional characters namely suitability, accuracy and complexity and four non-functional characters namely usability, maintainability, reusability and portability.

Service Oriented System (SOS)

Reusability in services is a measure of service components. The existing metrics in service oriented system that is listed below does not directly measure reusability. They measure the functional attributes such as service coupling, service cohesion, service granularity, modularity, commonality and composability and non-functional attributes like discoverability, availability, adaptability, portability, testability and modifiability as listed below.

Si Won Choi, Jin Sun Her, and Soo Dong Kim [38] proposes a set of QoS metrics for service providers especially considering the consumer's concern. They defined metrics for availability, dynamic discoverability, adaptability and compos ability.

C Si Won Choi and Soo Dong Kim [42] propose a comprehensive quality model for evaluating reusability of services which defines the metrics for modularity, commonality, adaptability, discoverability and standard conformance.

Zain Balfagih and Mohd Fadzil Hassan [45] presented a quality model that classifies nonfunctional characteristics based on the different stakeholders' requirements (consumer, developer and provider).

Qingqing Zhang and Xinke Li [47] propose a set of complexity metrics for service-oriented system. They defined basic measures like number of services, service granularity, coupling of service and importance of service.

Wang Xiao-jun [46] defines the metrics centered on service design principles concerning loosely-coupled and well-chosen granularity.

On observation it is precise that object oriented and Component based system metrics to measure reusability is at mature level, but in the case of Service oriented system the metrics are not clearly or directly defined.

3. DESIGN OF EVOLUTION BASED MODEL

In the study focused on exhibiting various reusability metrics of the three observed systems, namely object oriented, component base and service oriented systems and stating the quality factors affecting reusability [55]. This study directed to find the relationship between the metrics of the systems considered. The study proposed an evolution based model, to form the relationship of metrics that could be brought out between the systems and thereby stated the maturity level of reuse metrics and identified the gaps to measure complete reusability for service oriented system. A template has been designed and was used to categorize the influence of the measures into direct and indirect in order to bring out the evolution of the metrics in the three systems.

Hence the objective of this study was to improve the evolution based model, by estimating the strength and formulizing the relations. This section give the details, the improvements done over the model and template designed that directs to strength estimation based on each relations stated.

3.1 Template based Strength Prediction

The main ingredients of this template are classified into two, constituents and facets as shown in figure 1.

The constituents focus on measure which consists of direct and indirect metrics. Direct here refers to metrics being influenced completely by the quality factor along with its relevant metrics. Indirect specifies the metrics that has been partially influenced by the quality factor or having minor effects on it.

Facets are the explicit item which is based on the previous work done towards the three systems respectively. This reference has a classification like direct and indirect. References supporting our evolution with respect to measures and quality factors are categorized as direct and which partially support the evolution is termed indirect. Based on these factors various relations are formed which help in estimating the strength of the evolution based model.

The explanation or the template itself could be understood through an illustration specified by the figure 2 given below.

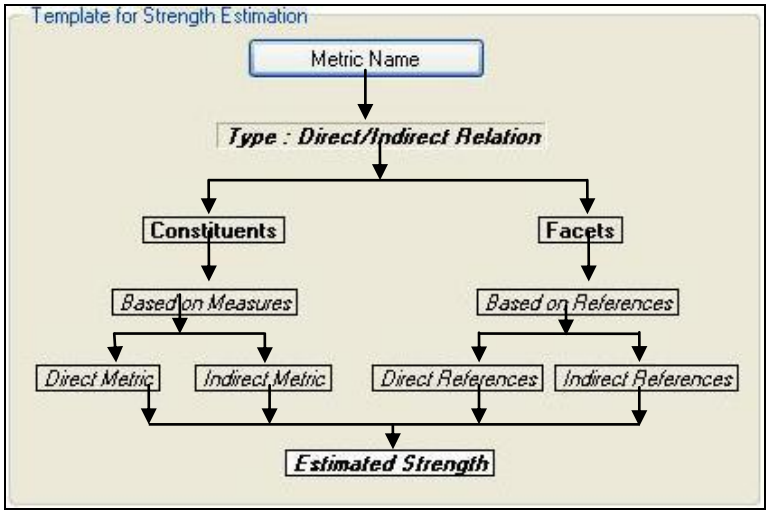

Fig 1: Template based approach for Estimating the strength of each relation

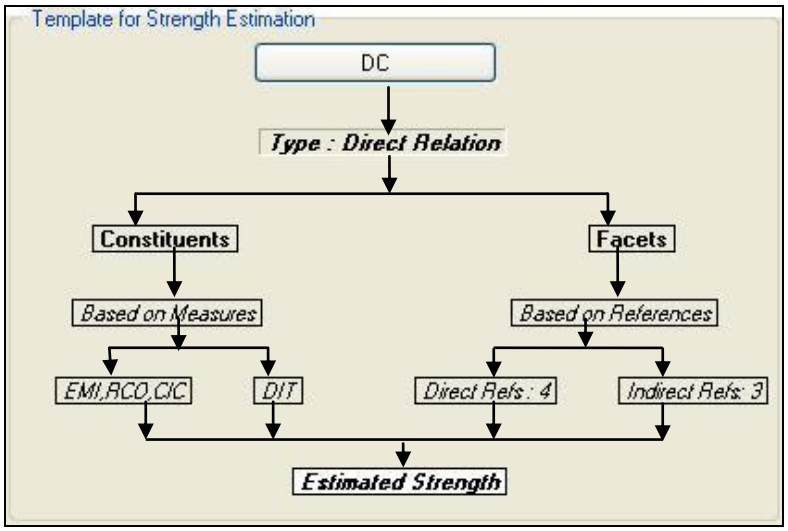

Fig 2: Strength Estimation for Discovery Metric in Service Oriented System

3.1.1 Strength Estimation

Each relation formed is obtained based on the two ingredients: metrics and reference which help in calculating the strength.

Formula for predicting strength of each relation

Direct relation is the average of the sum of ratio of number of direct measures to total number and number of direct reference to total number of references. Indirect relation is the average of the sum of ratio of number ratio of number of indirect measures to total number of measures and number of indirect reference to total number of references.

We used the fuzzy relation on the final values which shows that degree of the relation is strong or weak.

- **Formulae for direct relation in percentage**

$$DRpercentage = \left(\frac{(RDM + RDR)}{2} \right) * 100$$

Where,

RDM= Ratio of Direct Measure to Total Measure

RDRef= Ratio of Direct References to Total References

TM= Total Measures used by the relation

TRef= Total References supporting our relation

- **Formulae for indirect relation in percentage**

$$IDRpercentage = \left(\frac{(RIDM + RIDR)}{2} \right) * 100$$

Where,

RIDM= Ratio of indirect Direct measure to Total Measure

RIDRef= Ratio of Indirect Direct References to Total References

TM= Total Measures used by the relation

TRef= Total References supporting our relation

The value falls between in these ranges $\{.5 > DR \leq 1\}$ are strongest possible degree of our relation.

The value falls between in these ranges $\{0 > IDR \leq .5\}$ are weakest degree of our relation.

The table 1 & 2 given below represents the values of the two proposed metrics. Here [] & () notation denotes high and low influence on relation evolved.

Table 1. Strength Estimation for relating OOS with CBS Metrics

Relation	Facets		DR/IDR metrics value in %	Conclusion
	Classification	References		
[WMC, NOC, DIT] (RFC) => CIC	Direct	Rotaru and Dobre [14], V.lakshmi Narasimhan et. al., [22]Sharma A, et al., [17]	54	**S**
	Indirect	Washizak, et al., [11] Arun Sharma et al., [17]Puneet Goswami et al, [24] V. Prasanna Venkatesan andM. Krishanmoorthy [23] Jianguo Chen, et al., [58,25]	46	
[LCOM] (CBO) => TCC, RLCOM	Direct	G. Gui and P. D Scott, [15], Chuan Ho Loh and Sai Peck Lee [26] Kevin Hoffman and Patrick Eugster [60] Jianguo Chen, et al., [58,25]	56	**S**
	Indirect	Eunjoo Lee, et al., [59] Washizak, et al., [11] G. Gui and P. D Scott [21]	44	
[CBO] => DAC	Direct	G. Gui and P. D Scott[15,21] Jianguo Chen, et al., [58,25]	79	**S**
	Indirect	Eunjoo Lee, et al., [59] Majdi Abdellatief [57] Chuan Ho Loh and Sai Peck Lee [26]	21	
[CBO, LCOM] => CPM	Direct	Jianguo Chen, et al., [58, 25] V.lakshmi Narasimhan et. al., [22]	69	**S**
	Indirect	Washizak, et al., [11] Nasib S. Gill[61] [Arun Sharma , et al., [17,62,20]	31	
[WMC, NOC, RFC, DIT] => CRIT	Direct	V.lakshmi Narasimhan et. al., [19,22] [Rotaru and Dobre [14]	88	**S**
	Indirect	Puneet Goswami et al, [24]	12	
[WMC, DIT] => EMI	Direct	Washizak, et al., [11]	67	**S**
	Indirect	Rotaru and Dobre [24] Nasib S. Gill [61]	33	

[NOC, DIT] => RCC	Direct	Washizak, et al., [11] Arun Sharma , et al., [20]	75	**S**
	Indirect	Rotaru and Dobre, [14] Arun Sharma , et al., [17]	25	
[DIT, LCOM, WMC] => RCO	Direct	Washizak, et al., [11]	75	**S**
	Indirect	Kuljit Kaur Chahal and Hardeep Singh [63]	25	
[LCOM, CBO] => SCC(r,p)	Direct	Washizak, et al., [11] Harish Ramakrishnan [34] Kuljit Kaur Chahal and Hardeep Singh [63] Arun Sharma, et al., [20] V.lakshmi Narasimhan et. al.,[22]	81	**S**
	Indirect	Nasib S. Gill, [10, 61] Giliane Redolfi et al., [64] G. Gui and P. D Scott [21]	19	

Table 2. Strength Estimation for relating OOS and CBS with SOS Metrics

Relation	**Facets**		**DR/IDR metrics value in %**	**Conclusion**
	Classification	**References**		
[CIC,CPM, SCC] (RCC) => BCM	Direct	Si Won Choi and Soo Dong Kim [42] S. Vinoski [32] Bingu Shim et al., [40]	68	
	Indirect	Qingqing zhang and Xinke LI [47] Washizak, et al., [11]	32	**S**
[SCC, CRIT] => MD	Direct	Washizak, et al., [11] Si Won Choi and Soo Dong Kim [42] S. Vinoski [32] Bingu Shim et al., [40]	83	**S**
	Indirect	Wang Xiao-jun, [46] Anthony Hock-koon and Mourad Oussalah [52]	17	
[RCC] => AD	Direct	Washizak, et al., [11] Si Won Choi and Soo Dong Kim [42]	75	**S**
	Indirect	Bingu Shim et al., [40] El-Wakil et al., [7]	25	
[EMI, RCO, CIC] (DIT) => DC	Direct	Washizak, et al., [11] Wang Xiao-jun [46] Helge Hofmeister [41] Si Won Choi and Soo Dong Kim [42]	66	**S**
	Indirect	El-Wakil et al., [7] Qingqing zhang and Xinke LI [47] Anthony Hock-koon and Mourad Oussalah [52]	34	
(CPM) => SDCON	Direct	Washizak, et al., [11] Si Won Choi and Soo Dong Kim [42]	50	
	Indirect	No Reference found	50	**Neither S Nor W**
[CIC, SCC] (RCC) => SG	Direct	A. Khoshkbarforoushha et al., [51,53]	53	**S**
	Indirect	Si Won Choi and Soo Dong Kim [42] Washizak, et al., [11] Dmytro Rud et al., [35]	47	
(CBO, DAC) => SCO	Direct	M. Perepletchikov et al., [33] Helge Hofmeister [41]	25	**W**
	Indirect	El-Wakil et al., [7] Qingqing zhang and Xinke LI [47]	75	

(LCOM,RLCOM) => SC	Direct	M. Perepletchikov et al., [50] Dmytro Rud et al., [35]	25	**W**
	Indirect	El-Wakil et al., [7] Qingqing zhang and Xinke LI [47]	75	
[CIC, RCC, SCC] => MM	Direct	A. Khoshkbarforoushha et al., [51,53]	70	**S**
	Indirect	Si Won Choi and Soo Dong Kim [42] Bingu Shim et al., [40] Washizak, et al., [11]	30	
(CIC) => IS	Direct	A. Khoshkbarforoushha et al., [51,53]	20	**W**
	Indirect	Dmytro Rud et al., [53] Anthony Hock-koon and Mourad Oussalah [52] Mohammad Hadi Valipour et al., [44]	80	
(RFC) => SA	Direct	Washizak, et al., [11] M. Perepletchikov et al., [39]	25	**W**
	Indirect	Dmytro Rud et al., [53] Anthony Hock-koon and Mourad Oussalah [52]	75	

4. DISCUSSION

The evolution based model [55] has brought the relevance of the metrics between the three systems considered while also stating the impact of reusability metrics of one system over the other, for e.g., metrics of object oriented system to component based system. This paper has addressed various relations and template to estimate the strength on the findings of the literature survey. Here the outcome of the proposed systems mentioned in the previous section has been discussed.

The outcome of the modified template is as shown in figure 3.The strength evaluated though the relations can be depicted directly in the evolution based model which is given as an improved model as shown in figure 4. The figure shows the indications of the strong and weak relations of the evolution and also the contributors supporting our relationship. The strength estimation report clearly depicts that estimation of reusability has not reached its completeness. The strength reports the strong and weak factors that give way for more improvements towards the weak factor. At the outset the strength estimation presents a comprehensive view of factors and issues to be concentrated for building the complete reusability evaluation model for SOA.

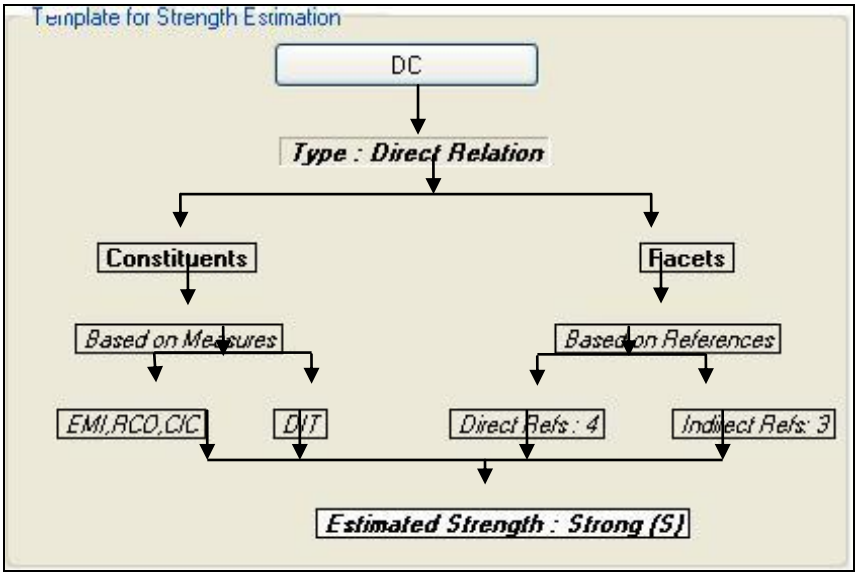

Fig 3: Outcome of the Strength Estimation Template

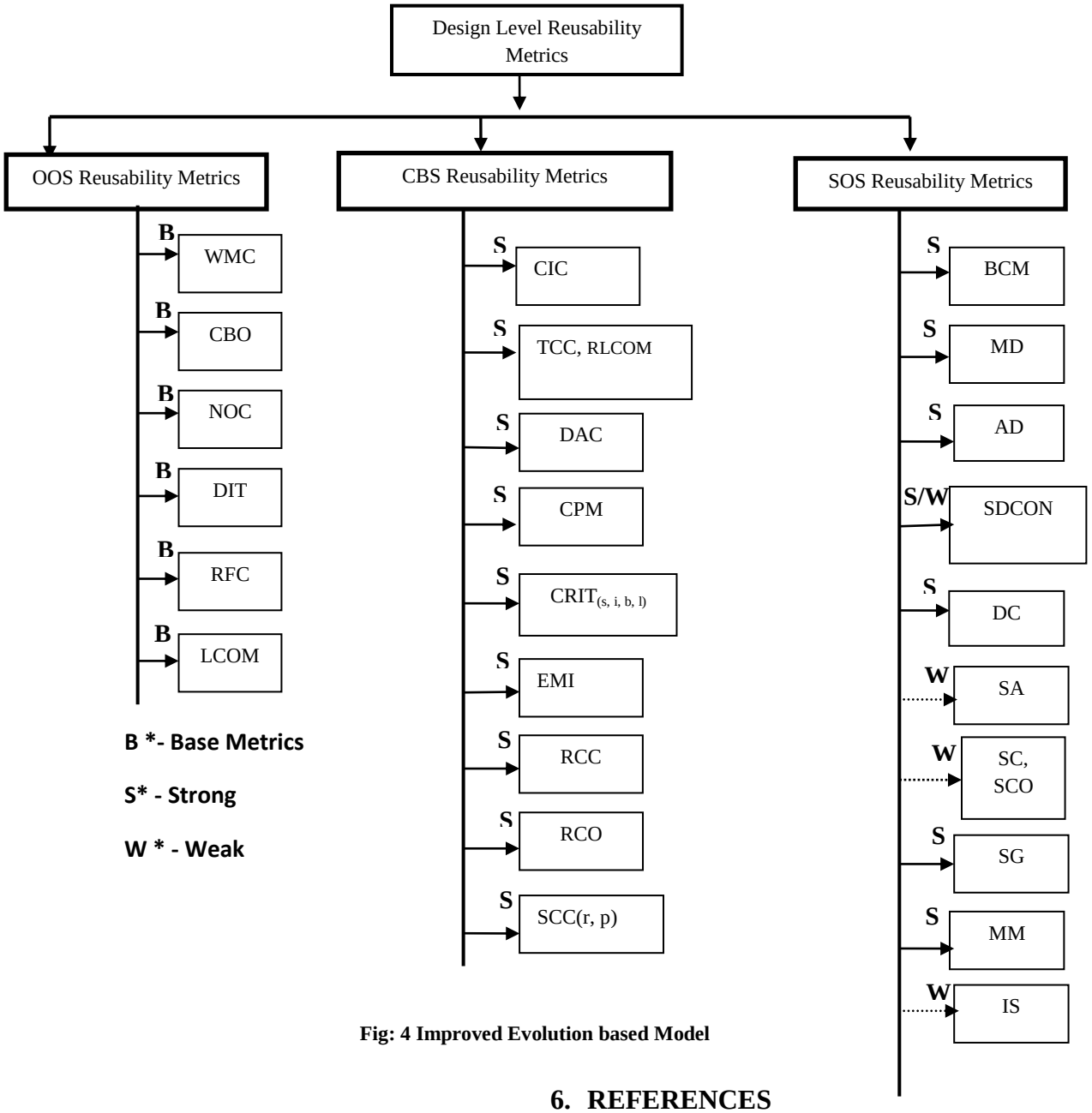

Fig: 4 Improved Evolution based Model

5. CONCLUSION

Evolution Based Model in which was proposed earlier has been taken for study. Relations defined in that model were further strengthened as in the modified template. Estimated strength of each relation between metrics shows that how strong or weak the evolution happens from one system to other. The reusability metrics of service oriented system have evolved from component based and object oriented as reported. This clearly indicates that metrics for measuring reusability in service oriented system is at the early stage. Further research would be focused on proposing a complete set of reusability metrics for SOS.

6. REFERENCES

[1] Rajaraman, C., and Lyu, M., "Reliability and Maintainability Software Coupling Metrics in C++ Programs," Proceedings of the Third International

[2] Symposium on Software Reliability Engineering, pp. 303-311, Oct. 1992.

[3] S. R. Chidamber and C. F. Kemerer , “A metrics suite for object oriented design”, IEEE Transactions on Software Engineering, 20(6):476–493, June 1994.

[4] J.M. Bieman, B.K. Kang: “Cohesion and Reuse in an Object-Oriented System”, Proceedings of the ACM Symposium on Software Reusability, April 1995

[5] Bansiya, J., and Davis, C.G., "An Object-Oriented Design Quality Assessment Model," University of Alabama, EUA, 1997.

[6] Young Lee and Kai H. Chang, "Reusability and. Maintainability Metrics for Object-Oriented Software*",ACM 2000

[7] El Emam, K.; Benlarbi, S.; Goel, N. and Rai, S. N, "The con-founding effect of class size on the validity of object-oriented metrics", IEEE Transactions on Software Engineering, 27(7): 630-650, 2001.

[8] El-Wakil, A. El-Bastawisi, M. Boshra, and A. Fahmy, "Object-Oriented Design Quality Models A Survey and Comparison" , International Conference on Informatics and Systems, 2004

[9] C. Neelamegam and Dr. M. Punithavalli,"A Survey - Object Oriented Quality Metrics",Global Journal of Computer Science and Technology, Vol 9, No. 4,page no 183-186, 2009

[10] E. Cho, M. Kim and S. Kim, "Component Metrics to Measure Component Quality", In Proc. of 8th Asia-Pacific Software Engineering Conference, IEEE CS, 2001.

[11] Nasib S. Gill and P. S. Grover, "Necessary Guidelines for deriving Component Based Metrics",ACM SIGPLAN SEN Vol 28, #6 Page 30, 2003.

[12] Hironori Washizak, Hirokazu Yamamoto and Yoshiaki Fukazawa,"A Metrics Suite for Measuring Reusability of Software Components",Proceedings of the Ninth International Software Metrics Symposium, IEEE 2003.

[13] Boxall, M. A. S. and Araban, S, "Interface Metrics for Reusability Analysis of Components", Australian Software Engineering Conference, 2004.

[14] Gill, N. S., Grover, P. S., "Few Important Considerations for Deriving Interface Complexity Metric for Component-Based Systems", ACM SIGSOFT Software Engineering Notes, Vol. 29 Issue 2, pp: 1-6, 2004.

[15] O.P. Rotaru, M. Dobre, "Reusability metrics for software components," ACS/IEEE International Conference on Computer Systems and Applications, 2005.

[16] G. Gui and P. D Scott,"Coupling and Cohesion Measures for Evaluation of Component Reusability", ACM 2006.

[17] Nael, S., "Complexity Metrics AS Predictors of Maintainability and Integrability of Software components", Journal of Arts and Sciences, Issue 5, pp: 39-50, 2006.

[18] Sharma, A., Kumar, R., Grover, P. S., "Complexity Measures for Software Components", WSEAS Transactions on Computers, Vol. 6, Issue 7, pp: 1005-1012, 2007.

[19] Luer, C., "Assessing Module Reusability", International Workshop on Assessment of Contemporary Modularization Techniques'07, IEEE 2007.

[20] V. Lakshmi Narasimhan, B. Hendradjaya, "Some theoretical considerations for a suite of metrics for the integration of software components", Journal of Information Sciences 177 ,844–864, Elsevier 2007

[21] Arun Sharma et al., "Reusability assessment for software components", SIGSOFT Software Engineering Notes, Volume 34 Number 2 March 2009.

[22] Gui Gui and Paul D. Scott, "New Coupling and Cohesion Metrics for Evaluation of Software Component Reusability", The 9th International Conference for Young Computer Scientists, IEEE 2008.

[23] V. Lakshmi Narasimhan, P. T. Parthasarathy, and M. Das, "Evaluation of a Suite of Metrics for Component Based Software Engineering (CBSE)" Issues in Informing Science and Information Technology, Volume 6, 2009.

[24] V. Prasanna Venkatesan, M. Krishnamoorthy ,"A Metrics Suite for Measuring Software Components", Journal of Convergence Information Technology Volume 4, Number 2, June 2009.

[25] Puneet Goswami, Pradeep Kumar, O P Sangwan, "A Metrics Methodology For Predicting Reusable Suite of Component Based Software System", International Journal of Computer Science and Security, Volume (4): Issue (1) 2010.

[26] Jianguo Chen, Hui Wang, Yongxia Zhou, Stefan D. Bruda ,"Complexity Metrics for Component-based Software Systems", International Journal of Digital Content Technology and its Applications. Volume 5, Number 3, March 2011

[27] Chuan Ho Loh and Sai Peck Lee, "Towards Cohesion-based Metrics as Early Quality Indicators of Faulty Classes and Components", International Symposium on Computing, Communication, and Control (ISCCC 2009) Proc .of CSIT vol.1 (2011)

[28] IEEE Computer Society, IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries: 610, Institute of Electrical and Electronics Engineers, New York, NY, 1990.

[29] E. J. Weyuker,"Evaluating software complexity measures", IEEE Trans. Softw. Eng., 14(9):1357–1365, 1988.

[30] P. Clements, R. Kazman, and M. Klein, "Evaluating Software Architectures", Addison-Wesley, Boston, 2002

[31] L. Brownsword, et al., "Current Perspectives on Interoperability", CMU/SEI-2004-TR-009, Software Engineering Institute, Pittsburgh, PA, March 2004.

[32] Kalepu, S., Krishnaswamy S., Loke S. W. Verity, "A QoS Metric for Selecting Web Services and Providers", Proceedings of the Fourth International Conference on Web Information Systems Engineering Workshops, 2004.

[33] S. Vinoski," Old measures for new services", IEEE Internet Computing, 9(6):72–74, 2005.

[34] M. Perepletchikov, C. Ryan, and K. Frampton "Comparing the Impact of Service-Oriented and Object-Oriented Paradigms on the Structural Properties of Software", OTM Workshops, pages 431–441, 2005.

[35] Harish Ramakrishnan, "Analysis of complexity and coupling metrics of subsystems in large scale software systems", M.S Thesis 2006.

[36] Dmytro Rud, Andreas Schmietendorf, Reiner R. DumkeProduct, "Metrics for Service-Oriented Infrastructures", IWSM/MetriKon 2006.

[37] Liam O'Brien, Paulo Merson, and Len Bass, "Quality Attributes for Service-Oriented Architectures",

International Workshop on Systems Development in SOA Environments,IEEE 2007.

[38] Perepletchikov, M., Ryan, C., Frampton, K., and Tari, Z., " Coupling Metrics for Predicting Maintainability in Service-Oriented Designs", Australian Software Engineering Conference (ASWEC), Melbourne, Australia, 329-340, 2007.

[39] Si Won Choi, Jin Sun Her, and Soo Dong Kim, "Modeling QoS Attributes and Metrics for Evaluating Services in SOA Considering Consumers' Perspective as the First Class Requirement" , Asia-Pacific Services Computing Conference, IEEE 2007.

[40] M. Perepletchikov, C. Ryan, K. Frampton, and H. Schmidt, "Formalising Service-Oriented Design," Journal of Software, vol. 3, NO. 2, pp. 1-14, 2008.

[41] Bingu Shim, Siho Choue, Suntae Kim, Sooyong Park, "A Design Quality Model for Service-Oriented Architecture",15th Asia-Pacific Software Engineering Conference, IEEE 2008.

[42] Helge Hofmeister and Guido Wirtz, "Supporting Service-Oriented Design with Metrics", 12th International IEEE Enterprise Distributed Object Computing Conference, 2008.

[43] Si Won Choi and Soo Dong Kim, "A Quality Model for Evaluating Reusability of Services in SOA" ,10th IEEE Conference on E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services, 2008.

[44] Renuka sindhgatta, et al., "Measuring the Quality of Service Oriented Design", ICSOC-sciencewave, LNCS 5900, pp. 485-499,2009.

[45] Mohammad Hadi Valipour et al., "A Brief Survey of Software Architecture Concepts and Service Oriented Architecture", IEEE 2009

[46] Zain Balfagih and Mohd Fadzil Hassan, "Quality Model for Web Services from Multi-Stakeholders' Perspective", International Conference on Information Management and Engineering, IEEE 2009.

[47] Wang Xiao-jun, "Metrics for Evaluating Coupling and Service Granularity in Service Oriented Architecture", IEEE 2009 .

[48] Qingqing Zhang and Xinke Li, "Complexity Metrics for Service-Oriented Systems", Second International Symposium on Knowledge Acquisition and Modeling, IEEE 2009.

[49] Kanjan Thammarak, "Survey Complexity Metrics for Reusable Business Process", ACTIS2010.

[50] Dominik Q. Birkmeier, "On the State of the Art of Coupling and Cohesion Measures for Service-Oriented System Design", Proceedings of the Sixteenth Americas Conference on Information Systems, 2010.

[51] Mikhail Perepletchikov, Member, IEEE, Caspar Ryan, and Zahir Tari, "The Impact of Service Cohesion on the Analyzability of Service-Oriented Software", IEEE Transactions on Services Computing, vol. 3, no. 2, april-june 2010.

[52] Khoshkbarforoushha et al., "Towards a Metrics Suite for Measuring Composite Service Granularity Level Appropriateness", IEEE 6th World Congress on Services, 2010.

[53] Anthony Hock-koon, Mourad Oussalah, "Defining Metrics for Loose Coupling Evaluation in Service Composition", IEEE International Conference on Services Computing, 2010.

[54] Khoshkbarforoushha, P. Jamshidi, F. Shams, "A Metric for Composite Service Reusability Analysis", ACM 2010.

[55] Saad Alahmari, Ed Zaluska, David C De Roure, "A Metrics Framework for Evaluating SOA Service Granularity", International Conference on Services Computing, IEEE 2011.

[56] G. Shanmugasundaram , et al.," Reusability Metrics – An Evolution based Study on Object Oriented System, Component based System and Service Oriented System", Journal of Computing, Volume 3, Issue 9, ISSN 2151-9617,September 2011.

[57] George Feuerlicht, "Simple Metric for Assessing Quality of Service Design", ICSOC workshops, LNCS 6568,pp. 133-143,2011

[58] Majdi Abdellatief, "Component-based Software System Dependency Metrics based on Component Information Flow Measurements",The Sixth International Conference on Software Engineering Advances, ISBN: 978-1-61208-165-6 ICSEA 2011.

[59] J. Chen, W. K. Yeap, S. D. Bruda, "A Review of coupling Metrics for Component-Based Development", In Proceedings of 2009 World Congress on Software Engineering (WCSE 2009), vol. 4, pp.65-69, 2009

[60] Eunjoo Lee, Byungjeong Lee, Woochang Shin and Chisu Wu, "A Reengineering Process for Migrating from an Object-oriented Legacy System to a Component-based System", Proceedings of the 27th Annual International Computer Software and Applications Conference, IEEE 2003.

[61] Kevin Hoffman and Patrick Eugster, "Towards Reusable Components with Aspects: An Empirical Study on Modularity and Obliviousness", ACM 2008.

[62] Nasib S. Gill, "Importance of Software Component Characterization For Better Software Reusability", ACM SIGSOFT Software Engineering Notes, Volume 31 Number 1 January 2006.

[63] Arun Sharma et al., "Estimation of Quality for Software Components – an Empirical Approach", SIGSOFT Software Engineering Notes Volume 33 Numbers 6 November 2008.

[64] Kuljit Kaur Chahal, Hardeep Singh, "A Metrics Based Approach to Evaluate Design of Software Components", IEEE International Conference on Global Software Engineering 2008.

[65] Giliane Redolfi et al., "A Reference Model for Reusable Components Description", Proceedings of the 38th Hawaii International Conference on System Sciences, 2005.