

# DEFINING RELATIVE QUALITIES OF OBJECT ORIENTED DESIGN IMPLEMENTATIONS USING AHP AND PROMETHEE

Petrus Mursanto\*  
Enterprise Computing Lab (ECL)  
Pusat Ilmu Komputer Universitas Indonesia  
E-mail: [santo@cs.ui.ac.id](mailto:santo@cs.ui.ac.id)

Wida Sari  
Magister Teknologi Informasi  
Fakultas Ilmu Komputer Universitas Indonesia  
E-mail: [wida12@cs.ui.ac.id](mailto:wida12@cs.ui.ac.id)

## ABSTRACT

A combination of Analytical Hierarchy Process and Promethee for defining ranks of object oriented (OO) software quality using MOOD2 metric set has been proposed. Software quality is presumably represented by the implementation of OO codes. MOOD2 metrics are measured on a number of Java-based applications. AHP is applied to determine weights of the MOOD2's criteria and Promethee is employed to determine the final rank of quality posed by each application codes. The rank of implementation quality is beneficial for evaluating and selecting best OO software design. Based on the result of measuring OO quality, software structure can be improved further to achieve better characteristics that also reflect better design of the software.

Keywords: AHP, Promethee, OO Metrics, MOOD2

## 1. Introduction

The advanced of software engineering has shifted software development from procedural to object oriented (OO) paradigm. It results in changes on how software design is evaluated. In procedural, algorithm complexity is the main indicator of software design [1]. However, the metrics of software complexity are considerably not appropriate for object oriented software [2]. There have been many researches to develop set of metrics for systems developed using OO approach. Several studies show the adoption of these metrics to support OO programming such as Ada and Classic-Ada [3], C++ and Eiffel [4].

The adoption of the OO paradigm is to produce better and cheaper software. The main structural mechanisms of this paradigm, namely inheritance, encapsulation, information hiding or polymorphism, are the keys to foster reuse and achieve easier maintainability. However, the use of language constructs that support those mechanisms can be more or less intensive, depending mostly on the designer ability. We can then expect rather different quality products to emerge, as well as different productivity gains.

A measurement tool is required to produce quantitative values representing OO program's qualities. Christariny developed Metrics Calculator for Java program based on Abreu's metrics [5]. Nurmaya improved the usability of the calculator [6] and utilized it for explorative experiments on the applicability of the Confidence Interval Good OO Design (CIGOOD) proposed by Abreu in [7]. The drawback of CIGOOD was revealed by Nurmaya, i.e. the measurement result is much dependent on the quality of

---

\* Corresponding author

sample programs taken as the statistical population of the method. By treating OO programs as alternatives and OO metrics as criteria, Hermawan [8], Dulianto [9] and Rahman [10] had demonstrated the solidity and robustness of the Analytic Hierarchy Process (AHP) introduced by Thomas Saaty [11] in defining ranks of OO program qualities.

This paper presents an alternative approach for defining rank of qualities over a number of Java programs using a combination of AHP and Promethee introduced by Brans [12]. Presumably, the Java executable jars measured in the method represents the implementation quality of their respective OO designs. The approach delivers similar results with the ones produced by applying AHP in [10] and surprisingly, their relative qualities among others remain constant. It shows the solidity of AHP's mechanism in employing multi-criteria based decision making. Moreover, this paper also shows that AHP's pair comparisons can eliminate the lack of objective weighting method in Promethee.

## 2. Literature Review

This paper elaborates three main concepts, i.e. MOOD, AHP and Promethee, each of which is discussed briefly in the following sections.

### 2.1. MOOD2

MOOD (Metrics for Object Oriented Design) and MOOD2 (second version of MOOD), consist of 11 sub-criteria that measure four main structural mechanism of object oriented design, explained in the following points [10]:

- 1) *Encapsulation*: this property describes how much the design hides method and attributes internally within implementation details. Encapsulation is measured in four sub-criteria, i.e. Attribute Hiding Factor (AHF), Operation Hiding Factor (OHF), Attribute Hiding Efficiency Factor (AHEF) and Operation Hiding Efficiency Factor (OHEF)
- 2) *Inheritance*: this property describes class hierarchy and reusability. Inheritance is measured in three sub-criteria, i.e. Attribute Inheritance Factor (AIF), Operation Inheritance Factor (OIF) and Internal Inheritance Factor (IIF).
- 3) *Polymorphism*: this property describes how much interfaces are implemented to ease message passing between objects. Polymorphism is measured in two sub-criteria, i.e. Behavioral Polymorphism Factor (BPF) and Parametric Polymorphism Factor (PPF).
- 4) *Coupling*: this property describes how tight the relationship between classes. Coupling is measured in two sub-criteria, i.e. Class Coupling Factor (CCF) and Internal Coupling Factor (ICF).

### 2.2. AHP

Analytic Hierarchy Process (AHP) is one of Multi Criteria Decision Analysis (MCDA) method to weigh criteria and rank alternatives [11]. AHP has three basic principles:

- 1) *Hierarchy construction*: decompose complex decision problem to simpler sub problems (criteria, sub-criteria, and alternatives).
- 2) *Criteria priority setting*: comparing priorities between paired criteria (pair-wise comparison) which is stated in nine levels scale (equally, moderate, strong, very strong, extremely important and their intermediate values). Priority setting must satisfy transitivity property, i.e. when comparing criteria  $i$  and  $j$ ,
- 3) *Consistency Checking*: determine criteria matrix's consistency ratio by computing  $CR = \frac{CI}{RI}$ . where  $CI$  = consistency index and  $RI$  = random index. The matrix is considered consistent when its consistency ratio is less or equal to 0.1.

AHP is excellent in structuring problems but somehow has unclear methodological aspect in compensation on determining final value of an alternative, lack of visualization, inability to identify incomparable alternatives, and limited classification by using nine level scale that needs too much interaction [13, 14].

### 2.3. Promethee

Promethee (Preference Ranking Organization Method for Enrichment Evaluation) is a multi criteria decision method that has produced six extensions: partial ranking, complete ranking, interval based ranking, continuous case, MCDA with segmentation constraints, and a representation of human brain [15]. This paper explores only Promethee I and II, partial and complete ranking, respectively.

Once a set of alternatives and criteria have been determined, decision maker chooses one of six preference function for each of the criteria, defines required parameters (thresholds and accepted min-max values) and criteria weights to determine preference index for every pair of criteria alternatives. There are six preference functions to choose from, i.e: usual criterion, Quasi criterion, Linear criterion, Level criterion, Linear with indifference criterion and Gaussian criterion [16].

Promethee I or partial ranking is achieved by ranking alternatives' strength and weakness. Figure 1 depicts an example of Promethee I diagram. Promethee II or complete ranking is achieved by calculating net flow for each of the alternatives by reducing weakness from its strength. Figure 2 describes Promethee II diagram for corresponding partial ranks in Figure 1.

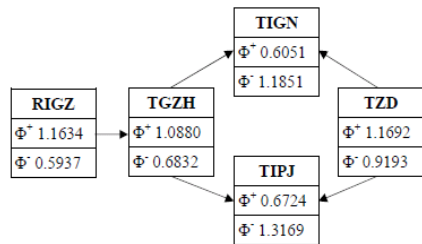


Figure 1. Partial ranking (Promethee I)

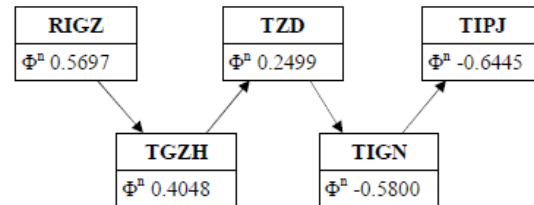


Figure 2. Complete ranking (Promethee II)

Unlike AHP, Promethee has no guidance to determine the weight and its original version. It does not have any support for criteria hierarchy either [12, 13]. Too many non-intuitive inputs are also required in Promethee such as suggested by [17]. However, Promethee is good in involving decision maker to simulate ranking process using different weights and preference functions. It also has simpler calculation and is easier to understand compared to AHP, as this may increase decision makers' confidence [18].

Considering that Promethee does not support weight determination and criteria hierarchy, there is research and literature that combine AHP and Promethee [14, 15, 19]. AHP's pair wise criteria comparison matrix offers consistent criteria weight [14]. Weight produced with this criteria matrix, can be treated as default weight set, which is alterable during simulation. This combination also offers less subjectivity and more stable ranking [19]. It should also be noted that in doing pair wise comparison, one should not always use nine level scale [14, 15].

### 3. Methodology

A set of experiments were conducted upon the same software set with previous research in [10], i.e. six java libraries (util, lang, io, math, ext, and net) and seven open source ERP packages (Adempiere, Freedom ERP, Plazma, jAllInOne, Compiere, TNT, and Millenium BSA). Weight set produced in previous research is used as default weights which is alterable during experiments. Default preference

function for all criteria is type 3 (linear) and each criteria standard deviation is used as threshold. Since AHP does not support minimum-maximum setting, this parameter is set to maximum value for every criteria. An experimental tool has been developed with the possibility to alter weight criteria groups (encapsulation, inheritance, polymorphism, and coupling). The tool is available on <http://telaga.cs.ui.ac.id/~wida12/file/prom/>.

#### 4. Experimental Results

Ranks obtained for java libraries, are displayed in Figure 3, Table I and Table II. By applying Promethee partial ranking we can see that pairs of java.math – java.io and java.lang – java.text are actually incomparable because java.math’s strength and weakness are bigger than the ones of java.io and java.lang’s strength and weakness are bigger than the ones of java.text.

Ranks obtained for ERP packages, are displayed in Figure 4, Table III and Table IV. These complete ranks equal to the ones resulting from [10]. Using Promethee partial ranking, we can see that pair jAllInOne – Millenium BSA is incomparable because jAllInOne’s strength and weakness are bigger than that of Millenium BSA.

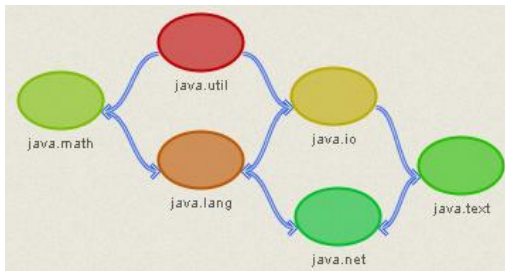


Figure 3. Partial ranking of java libraries

Table 1. Partial Rangking of Java Libraries

	Strength ( )	Weakness ( )
java.util	0.1367	0.0456
java.math	0.1203	0.1014
java.io	0.0818	0.0506
java.lang	0.0744	0.1447
java.text	0.0505	0.0767
java.net	0.0423	0.0869

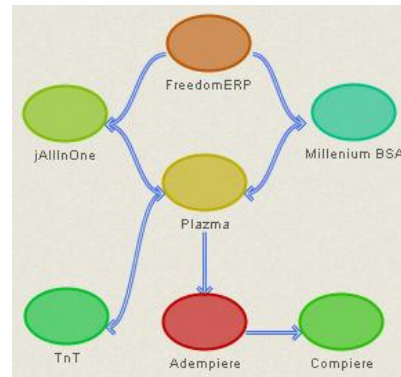


Figure 4. Partial ranking of ERP packages

Table 2. Complete Rangking of Java Libraries

Rank	Alternative	Net Flow ( )
1	java.util	0.0911
2	java.io	0.0311
3	java.math	0.0189
4	java.text	-0.0262
5	java.net	-0.0446
6	java.lang	-0.0703

Table 3. Partial Rangking of ERP Packages

	Strength ( )	Weakness ( )
FreedomERP	0.1423	0.0427
jAllInOne	0.1275	0.0709
Millenium BSA	0.1158	0.0475
Plazma	0.0948	0.0565
Adempiere	0.0857	0.184
TnT	0.0817	0.1711
Compiere	0.0603	0.1353

Table 4. Complete Rangking of ERP Packages

Rank	Alternative	Net Flow ( )
1	FreedomERP	0.0996
2	Millenium BSA	0.0683
3	jAllInOne	0.0566
4	Plazma	0.0383
5	Compiere	-0.075
6	TnT	-0.0895
7	Adempiere	-0.0984

The combination of AHP and Promethee results in rank of relative qualities shown in Table V. When the values are plotted into graph (Figure 5), consistence relative qualities are surprisingly maintained over the whole packages (java libraries and ERP packages).

Table 5. Relative Quality Rank of Java Libraries and ERP Packages

Rank	Alternative	Promethee Net Flow ( )	AHP Result
1	java.util	0.1192	0.444
2	java.io	0.0678	0.409
3	FreedomERP	0.0623	0.402
4	java.math	0.0545	0.400
5	Millenium BSA	0.0298	0.383
6	jAllInOne	0.0202	0.376
7	java.text	0.0164	0.375
8	Plazma	0.0024	0.365
9	java.net	-0.0021	0.363
10	java.lang	-0.03	0.354
11	Compiere	-0.1019	0.297

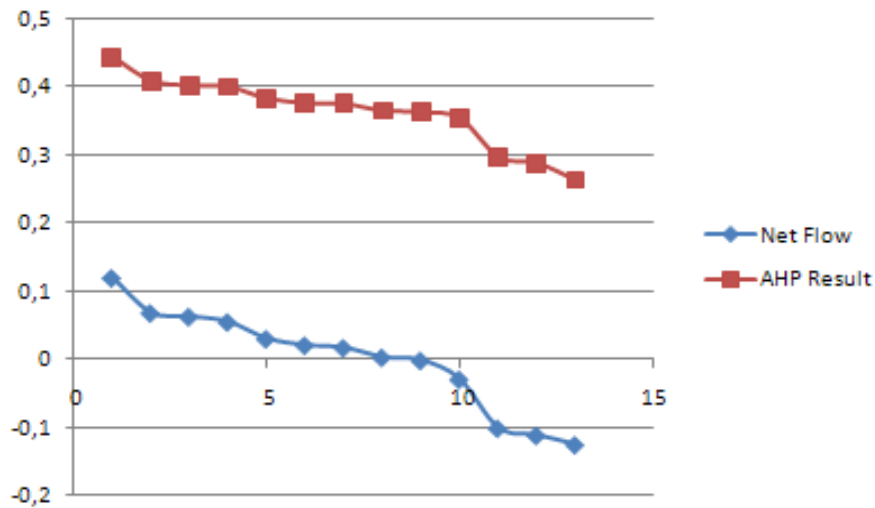


Figure 5. Comparison graph for AHP and Promethee final results

## 5. Conclusion and Future Research

The experimental results tell us that AHP and Promethee can be combined to define ranks based on MOOD2 measurement over several java packages. The obtained ranks are considered stable because all alternatives maintain their relative position of the ranks within the same package type. This method can be utilized to select java package having the best object oriented design quality. Moreover, by using Promethee I, it is possible to sort java programs by its strength and/or weakness and know which java programs are incomparable to one another. The results also support a view that AHP's pairwise comparison can act as a vital supplement for the lack of weighting mechanism in Promethee.

Based on benefits and weakness of both methods, future work can be directed to explore more on the hybrid method for evaluating OO software based on the combination of AHP and Promethee. It is also worth to develop measurement tools that has ability to evaluate OO design based on conceptual model of the application. This can expose weakness as well as improve the design before investing too much effort on code implementation.

## REFERENCES

- [1] Henry S., Long C. (2002) Measurement Technique for Object Oriented Systems. Available on <http://www.ssgrr.it/en/ssgrr2002s/papers/58.pdf>.
- [2] Basili V.R., Briand L. and Melo W.L. (1995) A Validation of Object-Oriented Design As Quality Indicators. Dept of Computer Science – Univ of Maryland. Available on <http://www.cs.umd.edu/projects/SoftEng/ESEG/papers/CS-TR-3443.pdf>.
- [3] Latanzzi M., Henry S. (1994) Object Oriented Metrics: Generation and Application. Available on <http://eprints.cs.vt.edu/archive/00000400/01/TR-94-20.ps>.
- [4] Abreu F.B., Melo W.L. (1996) Evaluating the Impact of Object-Oriented Design on Software Quality, 3rd International Software Metrics Symposium, Germany. Available on <http://www-ctp.di.fct.unl.pt/QUASAR/Resources/Papers/Metrics/Metrics96.pdf>.
- [5] Charitariny (2004) Metrics Calculator for Object Oriented System, Final Project, Faculty of Computer Science, Universitas Indonesia.
- [6] Nurmaya (2007) The Measurement of Software Design using Metrics for Object Oriented Design (MOOD), Final Project, Faculty of Computer Science, Universitas Indonesia.
- [7] Abrue F.B., Esteves R., Goulao M. (1996) The Design of Eiffel Programs: Quantitative Evaluation Using the MOOD Metrics, *Proceeding of TOOLS*, USA.
- [8] E. Hermawan (2007) Improving software design based on quality properties and MOOD using Analytic Hierarchy Process, Master Tesis, Faculty of Computer Science, Universitas Indonesia.
- [9] Dulianto D. (2008) The Application of AHP and Software Metrics in Defining Object Oriented Software Quality, Master Thesis, Faculty of Computer Science, Universitas Indonesia.
- [10] Rahman, M.Z. (2009). "The Application of Metrics For Object Oriented Design Version 2 in Measuring Design Quality of Object Oriented Software," Universitas Indonesia Fakultas Ilmu Komputer Program Studi Magister Teknologi Informasi.
- [11] Saaty, T.L. (2000) Fundamentals of Decision making and Priority Theory, RWS Publications, 1<sup>st</sup> ed.
- [12] J.P. Brans, Ph. Vincke (1985). "A Preference Ranking Organisation Method: (The PROMETHEE Method for Multiple Criteria Decision-Making)," *Management Science* Vol. 31, No. 6.
- [13] Wolfslehner, B. (2006). "Potentials And Limitations Of Multi-criteria Analysis Methods In Assessing Sustainable Forest Management," Department of Forest and Soil Sciences, Institute of Silviculture, University of Natural Resources and Applied Life Sciences, Peter-Jordanstr.
- [14] C. Macharis, J. Springael, K.D. Brucker, A. Verbeke (2003). "PROMETHEE and AHP: The Design of Operational Synergies on Multicriteria Analysis. Strengthening PROMETHEE with ideas of AHP.," *European Journal of Operational Research* 153 (2004) 307 - 317.

- [15] Z.K. Lin, J.J. Wang, Y.Y. Qin (2007). "A Decision model for Selecting an Offshore Outsourcing Location: Using a Multicriteria Method".
- [16] J.P. Brans, B. Mareschal (2005), "Multiple Criteria Decision Analysis: State of the Art Surveys - Chapter 5 Promethee Methods," Springer Science and Business Media, Inc.
- [17] M. Hostmann, "Decision Support for River Rehabilitation (2005). " Dissertation at Swiss Federal Institute of Technology Zurich.
- [18] A. Niknafs, N.M. Charkari, A.A. Niknafs (2008). "A PROMETHEE-based Recommender System for Multi-sort Recommendations in On-line Stores," Tarbiat Modares University.
- [19] K.P. Anagnostopoulos, C. Petalas, V. Pisinaras (2005). "Water Resources Planning Using The Ahp And Promethee Multicriteria Methods: The Case Of Nestos River - Greece," The 7th Balkan Conference on Operational Research, Romania.