

# The Method for Comparative Evaluation of Software Architecture with Accounting of Trade-offs

Alexander Kharchenko<sup>1</sup>, Ihor Bodnarchuk<sup>2\*</sup>, Vasyl Yatcyshyn<sup>3</sup>

<sup>1</sup>Department of Computer Informational Technologies, National Aviation University, Kyiv, Ukraine

<sup>2</sup>Department of Computer Science, Ternopil Ivan Pul'uj National Technical University, Ternopil, Ukraine

<sup>3</sup>Department of Computer Engineering, Ternopil Ivan Pul'uj National Technical University, Ternopil, Ukraine

\*Corresponding author: bodnarchuk.io@gmail.com

Received September 30, 2013; Revised January 17, 2014; Accepted January 22, 2014

**Abstract** Since growing complexity of software systems it is more difficult to meet demands of quality during the process of their design. To solve this problem with minimal loss this process is transferred onto more early stages of design, particularly onto the design of architecture. The architecture is treated as the set of components that encapsulates the logic of calculations, and connections that ensure the interaction between components and create their configuration. Since the architecture of software system is a high-level abstract model for representation of system structure and key properties, its selection grounds the insurance of quality for software system. In the paper the questions of evaluation of architecture quality on the set of quality attributes with Analytic Hierarchic Process (AHP) with applying of optimization algorithm for estimating of weights of alternatives are discussed. The conflicts between quality indices and trade-offs between them are analyzed.

**Keywords:** *Software Architecture, Multivalued Optimization, Quality of Software Architecture, Software Architecture Evaluation*

**Cite This Article:** Alexander Kharchenko, Ihor Bodnarchuk, and Vasyl Yatcyshyn, "The Method for Comparative Evaluation of Software Architecture with Accounting of Trade-offs." *American Journal of Information Systems*, vol. 2, no. 1 (2014): 20-25. doi: 10.12691/ajis-2-1-5.

## 1. Introduction

The quality of architecture decision is evaluated on the totality of criteria, and its selection always is a trade-off, because for the given set of functional requirements and requirements for quality there is no single best solution, and improving of some criteria leads to deterioration of others and vice versa. So why when selecting the architecture decision for software system it is necessary to apply the methods of multicriteria evaluation with accounting of conflicts between indices of quality and to make trade-offs.

There are early and late evaluation of architectures. Early one is based on the experience of designers and logical reasoning because there are not any arte facts that allow to simulate the performance of the software system. Methods that realize early evaluation are based of scenarios. Following methods are belong to this group: SAAM and ATAM [1]. On the base of stakeholder priorities defined quality criteria. A scenario is working out to check the satisfaction for each quality attribute, and assessing the level of satisfaction of the attribute by given architecture alternative is carried out.

ATAM method is similar to SAAM, but here for selected architectures the risks of satisfaction of quality attributes are evaluated on the base of analysis of scenarios. Evaluation of risks the group of experts

performs, and this group as well ranges alternatives according to the level of risk and defines so called points of sensitivity on the components or connections of architecture, trade-offs between indices of quality are analyzed too.

For reasonable selection of decision in the method SAAM/ATAM selected architectures are analyzed on the subject of costs and benefits with applying of method CBAM [2]. This method ensures economic analysis of software system, which are based on the selected on previous methods variants of architecture and scenarios of simulation. Experts assign scores for quality indices from 1 to 100 and range architectures according to scores what these architectural decisions ensure for certain attribute. Evaluation of each architecture alternative is calculated according to the following expression:

$$B(A_i) = \sum_{j=1, K} (Cont_{i,j} \cdot Q_j) \quad i = \overline{1, n} \quad (1)$$

Here –  $Cont_{ij}$  is the weight of  $i^{\text{th}}$  architecture corresponding to  $j^{\text{th}}$  attribute;

$Q_j$  – the priority of  $j^{\text{th}}$  attribute.

This method allows to evaluate cost for realization of each alternative and gives possibility to calculate the index of desirability as relation of benefit to costs. On the base of obtained data the best solution is selected.

Often appears the problem to create software system on the base of existing one via redesign to meet new requirements for quality. To solve problems like this the method SSAR for reengineering of software system architecture on the base of scenarios was created [3], that is the aggregate of four methods of architectures evaluation relatively to quality attributes:

- evaluation on the base of scenarios;
- simulation;
- math simulation;
- evaluation of the base of practical experience.

When using SSAR the one method among other is selected, but the main is the method on the base of scenarios. This method is similar to that one realized in SAAM.

Detailed analysis of scenario-oriented methods for evaluation of architectures such as ATAM, SAAM, SAEM and others is made in the work [4]. Shown the schemas of scenarios for different methods of evaluation. The conditions when applying of certain method will be most effective are described.

The common disadvantage for all discussed above methods is consequent evaluation of the architecture relatively to one parameter, what require to elaborate new scenario and to carry out expert evaluation of risks each time, so it makes the process of selection laborious and nonformalized. Here as well impossible to obtain comparative estimations for the set of alternatives.

Further researches in this field showed the effectiveness the method on the base of Analytical Hierarchic Process (AHP) for solving of such problems [5,6]. One of the first publications on applying of AHP for evaluation of software system architectures on the totality of quality indices is the work of M. Svahnberg [6]. It contains the step-by-step algorithm of solution for the problem of evaluation of the set of architectures on the totality of quality indices and choosing of the best architecture of software system. The method's utilization is illustrated on the concrete example.

To overcome the disadvantages of AHP connected with inconsistency of the matrix of pair wise comparisons for big quantity on alternatives ( $n \geq 7$ ) it is proposed to correct the elements of the matrix. But it leads to partial use and disfiguration of expert data what decrease the truthfulness of obtained results. In the works [6,8,9] for correct applying of AHP in case of big quantity of alternatives ( $n \geq 9$ ) the modified AHP was used.

In the modified AHP the comparative evaluation of architectures are used relatively to their realization of quality attributes. It allows to find relative weights of alternatives for each quality attribute and to range them. According to assigned by stakeholders priorities of quality attributes their averaged value is calculated and the weights of alternatives are estimated regarding to the totality of quality attributes.

Obtained relative assessments of alternatives can be used for analysis of conflicts between quality attributes and for identifying of compromise decision.

The advantages of modified AHP method are the evaluation of alternatives for all quality attributes, optimization of decisions, and enough high level of formalization, what allows to automate the process.

Offered in this paper approach is grounded on the applying of AHP with optimization algorithm for finding of weights of alternatives. This allows to expand the usage

of algorithm on greater quantity of compared alternatives ( $n > 9$ ). The sensitivity of solution to the change of priorities for quality criteria is examined, and conflicts and trade-off between criteria are analyzed.

## 2. The Evaluation of Architectures with Analytical Hierarchic Process

The process of comparative evaluation of architectures with applying of AHP is shown on the Figure 1.

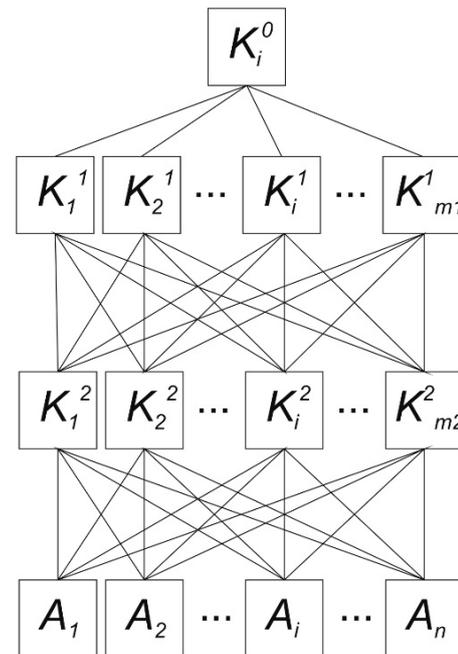


Figure 1. Hierarchic representation of the problem of architecture evaluation

The selection of architecture must be performed so that built on its base software system will meet quality requirements. Hence here we see two levels of interconnected quality attributes:

$K_i^1, i = \overline{1, m1}$  – quality attributes of software system according to the standard ISO/IEC 25010;

$K_i^2, i = \overline{1, m2}$  – quality attributes of architecture;

$A_i, i = \overline{1, n}$  – alternative architectural decisions.

The set of quality criteria of software system  $\{K_i^1\}$  and constraints for them are defined on the stage of gathering of requirements to the software system. The set of quality indices for the architecture  $\{K_i^2\}$  can be defined with applying of QFD (Quality Function Deployment) method or the method of pair wise comparisons [10].

It is necessary to select among available alternatives  $\{A_i\}$  such one, that will ensure the quality of software system. In other words it is necessary to solve the optimization problem regarding to the totality of parameters  $\{K_i^1\}$  and  $\{K_i^2\}$ . This is the problem of multicriteria hierarchical optimization and to solve it the AHP will be applied [11].

When AHP is applying for solution of problems like this the relative weights  $\{w_i^s\}$  of alternatives are found on each level using of matrices of pairwise comparisons  $B^s \{b_{ij}^s\}$ . These matrices are filled up by experts and here  $b_{ij}^s$  means the supremacy of  $i^{\text{th}}$  alternative over  $j^{\text{th}}$  one.

Coefficients of matrices must be consistent, i.e.  $b_{ij}^s = w_i^s / w_j^s \quad \forall b_{ij}^s \in B^s$ . The weights in this case will be found as components of eigenvector for matrix of pairwise comparisons, which correspond to the maximum characteristic number of the matrix. The calculation of eigenvector is enough complicate procedure. So why the approximate value is used very often [11]:

$$w_i^s = \frac{1}{n} \frac{\sum_{j=1}^n b_{ij}^s}{\sum_{j=1}^n b_{ij}^s} \quad (2)$$

But for considerable quantity of alternatives, since influence on experts of different factors, the matrix  $B^s \{b_{ij}^s\}$  is inconsistent and its rank will differ from 1, i.e. the matrix will have some different eigenvalues. A. Pavlov in his paper [12] proposed for solution of this problem models and method for calculation of weights for alternatives on the base of minimization of inconsistency of the matrix  $B^s \{b_{ij}^s\}$ .

And in [6,8,9] this method was applied for the problem of architecture selection with accounting of quality indices. For this problem the measure of inconsistency was used like following.

$$\left| \frac{w_i^s}{w_j^s} - b_{ij}^s \right| \leq \delta_t \cdot b_{ij}^s, \quad \delta_t \geq 0, \quad (3)$$

where  $\delta_t$  – is given threshold value.

Then, when the condition of independency of error

form  $\frac{w_i^s}{w_j^s} - 1$  in estimation of  $b_{ij}^s$  by experts is kept,

weights  $w_i^s$ , which minimize (3) can be found as solution of linear programming problem [13]:

$$\begin{aligned} \min_{\{w_i\}} & \sum_{i=1}^n \sum_{j=1}^n (y_{ij}^+ - y_{ij}^-) \\ w_i^s & \geq a_i, \quad i = \overline{1, n}, \\ w_i^s - b_{ij}^s w_j^s & = y_{ij}^+ - y_{ij}^-; \quad b_{ij}^s \geq 1; \end{aligned} \quad (4)$$

$$\begin{aligned} -\delta_t \cdot b_{ij}^s \cdot w_j^s \leq w_i^s - b_{ij}^s \cdot w_j^s \leq \delta_t \cdot b_{ij}^s \cdot w_j^s, \\ y_{ij}^+, y_{ij}^- \geq 0; \quad i, j = \overline{1, n}. \end{aligned} \quad (5)$$

Thus, estimation of weights for alternative architectures  $w_i$  is reduced to the solution of problem (4), (5). During solving of this task the incompatibility of constraints (5) can appear, so the  $\delta_t$  must be increased. Application of

this algorithm gives possibility to obtain the weights of alternatives both for realization of each quality criteria and for their totality. In the work [7] the advantages of used modified AHP were illustrated.

But when making decision on selection of architecture on the found estimations of  $w_i^s$  for taking into account stakeholders' constraints and priorities it is need to analyze conflicts between quality criteria and to find compromise decisions.

### 3. Investigation of the Sensitivity when Ranging the Alternatives and Analysis of Trade-offs

After finding according to (4), (5) the solution  $w_i^s$  for each criterion the architectures can be ranged according to the values of weights.

Such research were carried out regarding the sensitivity of architectures' ranging to the changes of priorities for quality criteria. For this matter the weights of alternatives were calculated regarding for each following criteria: modifiability, scalability, performance, ease of install, cost, development effort, and portability.

And calculations for the totality of criteria were executed too.

Below most typical results of calculations are represented for four alternatives as most representative for their field of usage:

1. Three-level on the base of Java Enterprise Edition (TJEE) as the base for development uses the platform Java 2 Enterprise Edition (J2EE) what is at same time the industry-accepted standard that enables solutions for developing and deploying of distributed multitier enterprise applications.

2. Three-level on the base of .NET (TNET) is analogical to the TJEE but for its realization the tools and frameworks for applications development from Microsoft® are used.

3. Two-level (TWOL) provides the realization of all business logic on the client side.

4. Platform with support of distributed agent (DAS). Using of distributed agent (DAS) for building of the application provides higher flexibility. As a technology CORBA, DCOM or SOAP can be used.

Figure 2 shows weights of alternative architectures regarding for the quality criteria.

To define integral index of architecture quality we need to define the priorities of quality criteria. Since in the process of design of software systems some groups of stakeholders are involved, which have different priorities for each quality attribute, it is necessary firstly to define the priorities for each group via composing by them matrices of pairwise comparisons with applying of modified AHP to these matrices. Then priorities of criteria  $\{P_i^s\}$ ,  $i = \overline{1, k2}$  is calculated ( $s$  – is the number of group of experts).

Compromise decision can be found as a geometric mean  $P_{ij}^* = \sqrt[n]{P_{ij}^1 \cdot P_{ij}^2 \cdot \dots \cdot P_{ij}^n}$  or as mean with accounting of index of competency for each expert group

$$P_{ij}^* = P_{ij}^{\alpha_1} \cdot P_{ij}^{\alpha_2} \cdot \dots \cdot P_{ij}^{\alpha_n} \quad (\alpha_1, \alpha_2, \dots, \alpha_n - \text{are indices of competency}).$$

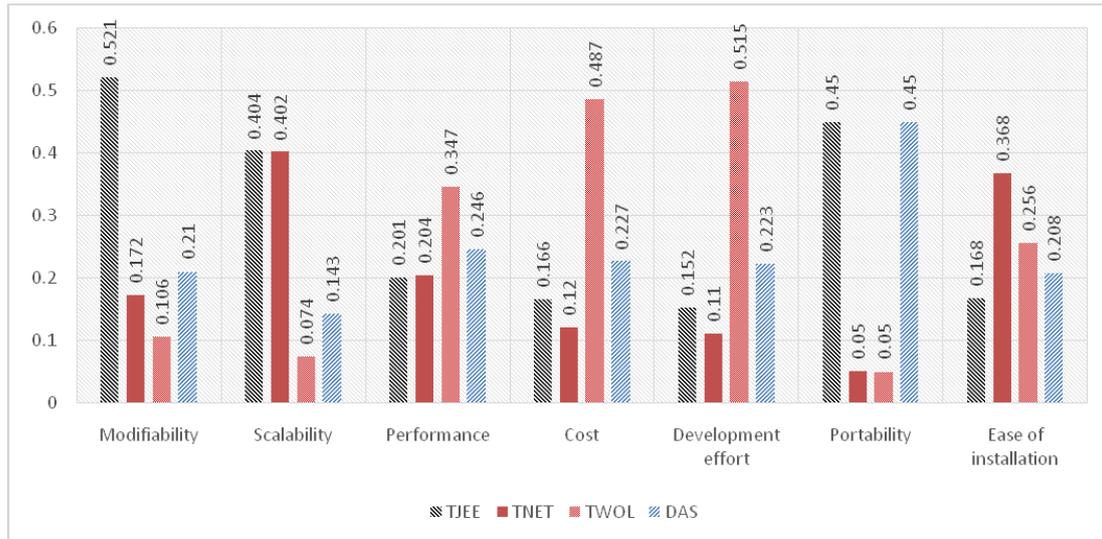


Figure 2. Weights of architectural alternatives regarding to quality criteria

The assessments of priorities for quality criteria from different expert groups obtained by method of pairwise comparisons are shown on the Figure 3.

As we can see from the Figure 3, modifiability is most important attribute with weight 0.28, then follows ease to

install. In the final order of ranged alternatives the sample TJEE considerably exceeds all the rest alternatives. DAS and TWOL are very closed with weights 0.237 and 0.248 correspondently. TNET is the worst variant.

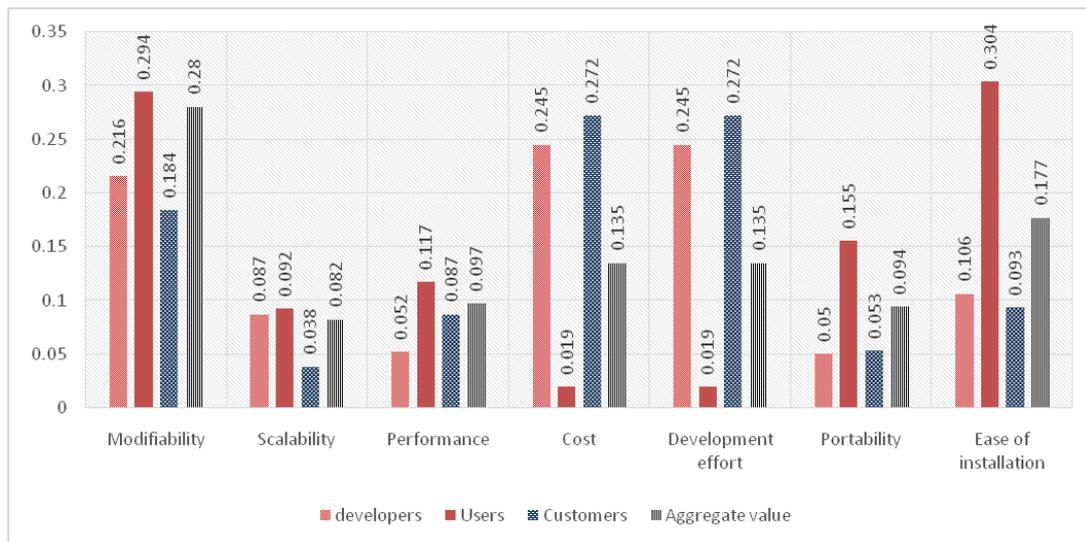


Figure 3. Priorities of quality criteria

It is not convenient to analyze correlations between weights of alternatives when they are represented as numbers. So more acceptable is graphical representation of these results.

Graphical representation is good as well for analyzing of trade-offs between quality attributes of architecture.

### 3.1. Trade-offs Analysis for Quality Attributes

The diagram of trade-offs between cost and modifiability for examined alternatives is shown of the Figure 4.

Such representation gives possibility to visualize possible trade-offs, their relative values and relations between alternative designs from the point of view of compromises.

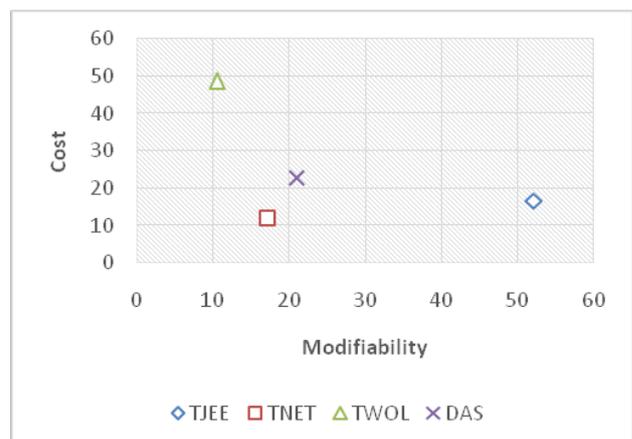


Figure 4. Trade-off between quality attributes

When conflicting attributes have same priority then graphically the trade-off between them will be shown on the diagonal of first quadrant. When architectural decision must be selected with accounting of the priority of one quality attribute so, that Pareto-optimal decision will be selected what is highest relatively to the diagonal of first quadrant when the attribute with priority is on the axis of ordinate. When the attribute with priority is on the x-axis, so the Pareto-optimal decision will be the lowest relatively to the diagonal of first quadrant in the rightest position.

For the decision TWOL (left upper cornet on the Figure 4 the architecture is Pareto-optimal regarding to the

critierion cost and when this criterion has higher priority than modifiability this architecture will be selected. In same time, the alternative TJEE is Pareto-optimal for the index of modifiability, and this is represented as point that is located under the diagonal in the most right position.

So the size of a trade-off is represented by the location of the point in the left-and-up or right-and-down direction relatively the diagonal of first quadrant. When the alternative is located in the third quadrant it means that both attributes have negative influence on the architecture decision.

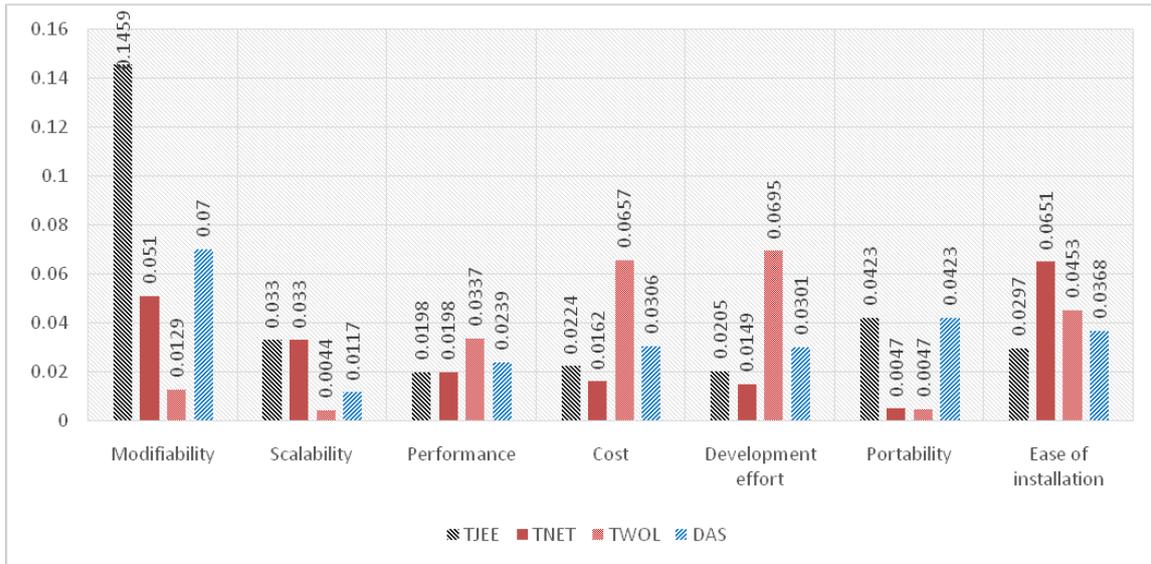


Figure 5. Ranged architectural alternatives with accounting of priorities of quality attributes

When to take into account the weights of attributes for analysis of trade-offs then the results can change. On the Figure 5 the architectural alternatives are shown, ranged according to the weights of attributes. Then the trade-offs diagram from the Figure 4 will be reduced to the diagram on the Figure 6.

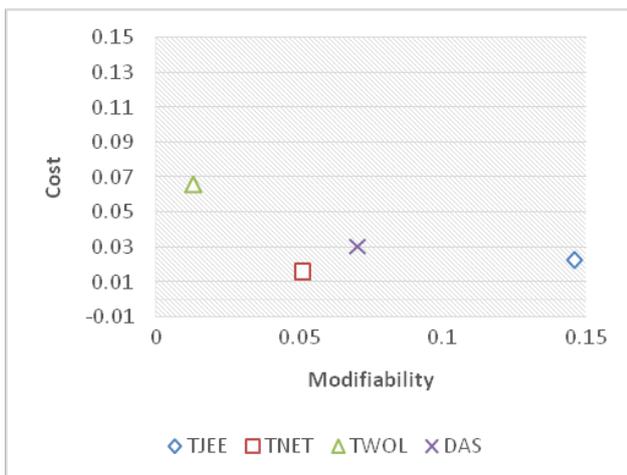


Figure 6. The trade-offs diagram between cost and modifiability (with accounting of weights of quality attributes)

The alternative TJEE has considerable superiority in modifiability regarding to other alternatives. So for the TJEE trade-offs between modifiability and any other quality attributes will strengthen on the diagram of trade-offs. Thus, these diagrams visualize trade-offs and their values, which are not visible when AHP is used.

### 3.2. Analysis of Sensitivity

Now let us examine the sensitivity of decisions obtained with AHP, to the change of priorities of quality criteria.

Let we have ranged architectural alternatives  $\{A_i\}$ , obtained with Analytical Hierarchic Process. The formula below is for definition of minimal change of absolute value of weight for quality attribute such, that the order of adjacent architectures  $A_i$  and  $A_j$  will be flipped on reverse as shown in the Eq. (6) below.

$$D'_{s,i,j} = \frac{|J_i - J_j|}{|w_i^s - w_j^s|} \cdot \frac{100}{P_s} \tag{6}$$

Here  $D'_{s,i,j} (s = \overline{1, m2}; i, j = \overline{1, n}, i \neq j)$  is minimal change of the value of the priority  $P_s$  for the quality criterion  $K_s$ , which changes the order of adjacent alternatives  $A_i$  and  $A_j$  in ranged list on opposite. The minimal value of  $D'_{s,i,j}$  shows that the priority  $P_s$  for the quality criterion  $K_s$  is critical for the changing of assessments in pairwise comparisons. This expression can be used as well for the case of change of requirements to the software system during the design process, what can lead to the change of priorities for quality criteria.

For each quality attribute more than one value of  $D'_{s,i,j}$  is possible, which can cause the change in the order of adjacent alternatives.

## 4. Conclusions

The above analysis of methods for tevaluation of architectures has shown, that the most acceptable is the Saaty's Analytical Hierarchic Process, which allows to get relative assessments of alternative architectures both for one quality criteria and for their totality. To expand the correctness of AHP the optimizational method of calculation (definition) for weights of alternatives is used, what is based on the model of inconsistency minimization for the matrix of pair wise comparisons. Obtained in AHP solutions are convenient for the analysis of the sensitivity of ranged alternatives to the change of priorities of quality criteria, and for definition of trade-offs when making decision. For the fixed example of alternatives and criteria the effectiveness of proposed method is shown.

Further researches are dedicated to creation of adaptive to changes of quality requirements method for evaluation of alternative architectures of software system on the set of quality indices. Such method will allow to make operative corrections in the evaluation of alternatives without repeat solution of the problem. One more direction of researches is the modification AHP with taking to account of degree of alternatives' incomparability and its applying for the problem of architectures selection on the set of quality indices. In fact the standard AHP cannot be applied when supremacy of some alternative over other one is more than 9 [11].

## References

- [1] Kazman, R. *ATAM: Method for Architecture Evaluation* / Rick Kazman, Mark Klein, Paul Clements. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, August 2000. CMU/SEI-2000-TR-004, ADA377385. 83 p.
- [2] Kazman, R. Quantifying the costs and benefits of architectural decision/ Kazman, R., Asundi, J., and Klein // *Proceedings of the 23rd International Conference on Software Engineering (ICSE)*, 2001. pp. 297-306
- [3] Bengtsson, Perlof Architecture-level modifiability analysis (ALMA) / Perlof Bengtsson, Nico H. Lassing, Jan Bosch, Hans van Vliet // *Journal of Systems and Software*. 2004. Vol. 69, No. 1-2. pp. 129-147.
- [4] L. Dobrica and E. Niemela. A Survey on Software Architecture Analysis Methods. *IEEE Transactions on Software Engineering*, vol. 28, no. 7, pp. 638-653, July 2002.
- [5] T. Al-Naeem, I. Gorton, M.A. Babar, F. Rabhi, and B. Benatallah, "A quality driven systematic approach for architecting distributed software applications", *Proceedings of the 27th International Conference on Software Engineering (ICSE)*, St. Louis, USA, 2005. pp. 244-253.
- [6] M. Svahnberg, C. Wholin, and L. Lundberg. A Quality-Driven Decision-Support Method for Identifying Software Architecture Candidates. // *Int. Journal of Software Engineering and Knowledge Engineering*, 2003. 13(5): pp. 547-573.
- [7] Harchenko Alexandr, Bodnarchuk Ihor, Halay Iryna. Stability of the Solutions of the Optimization Problem of Software Systems Architecture // *Proceeding of VIIth International Scientific and Technical Conference CSIT 2012*. pp. 47-48, Lviv, 2012.
- [8] AlexandrHarchenko. The Tool for Design of Software Systems Architecture // AlexandrHarchenko, Ihor Bodnarchuk, Iryna Halay, Vasylyatcyshyn // *Proceeding of XIIth International Conference CADSM' 2013*. pp. 47-48, Lviv.
- [9] AlexandrHarchenko. DecisionSupportSystemofSoftwareArchitect // AlexandrHarchenko, Ihor Bodnarchuk, IrynaHalay // *Proceeding of the 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS). Volume 1*, pp. 265-269, Berlin.
- [10] Kharchenko O.H. Instrumentaljnij zasib rozrobky ta komunikaciji vymogh jakosti do prohramnykh system / O.G. Kharchenko, V.V. Yatcyshyn, I.E. Rajchev // *Inzhenerija prohramnogo zabezpechennja*. Kyiv, – 2010. # 2. pp. 29-34.
- [11] Saaty T. *Decision Making with the Analytic Network Process.*/ Saaty T. Vargas L.// – N.Y.: Springer, 2006. 278 p.
- [12] Pavlov A.A. Matematicheskie modeli optimizacii dlanahozhdenijavesov obyektov v metode parnyh sravnenij. Pavlov A.A, Lishchuk E.I., Kut V.I.// *Sy`stemni doslidzhennya ta informacijni tehnologiyi* – Kyiv.: IASA (Institute for Applied System Analysis), 2007. #2, pp. 13-21.
- [13] Zgurovskij M.Z., Pavlov A.A., Shtankevich A.S. Modificirovannyj metod analiza ierarhij // *Sy`stemni doslidzhennya ta informacijni tehnologiyi* – Kyiv.: IASA (Institute for Applied System Analysis),–2010. #1, pp. 7-25.