Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 178 (2017) 233 – 242

Procedia Engineering

www.elsevier.com/locate/procedia

16th Conference on Reliability and Statistics in Transportation and Communication, RelStat'2016, 19-22 October, 2016, Riga, Latvia

Hierarchical Criterion Approach for Architecture Structure Selection of Transportation Software

Sergey Orlov, Andrei Vishnyakov*

Transport and Telecommunication Institute, Lomonosova str.1, Riga, LV-1019, Latvia

Abstract

Architectural structural design of modern transportation software has a huge impact on a quality and cost of the development process, hence it's required to pay reasonable attention while making architecture design decisions. The technique proposed in this paper allows selecting the optimal software architecture among several alternatives. This selection technique is reduced to the criteria importance theory for decision-making problems with a hierarchical criterion structure. For applying it, we need to pick up a set of metrics that assess the characteristics of the software architecture. Next, we need to determine metrics scale and create the hierarchical criterion structure with all the relations between software metric groups.

Peer-review under responsibility of the scientific committee of the International Conference on Reliability and Statistics in Transportation and Communication

Keywords: multicriteria decision analysis, hierarchical criterion, criteria importance theory, software architecture, architectural patterns

1. Introduction and related work

The formation of architecture is the first and fundamental step in the software design process and provides the framework of a software system that can perform the full range of detailed requirements (Orlov, 2016; Bass *et al.,* 2013).

* Corresponding author.
E-mail address: Andrei.Vishnyakov@gmail.com

Peer-review under responsibility of the scientific committee of the International Conference on Reliability and Statistics in Transportation and Communication
doi:10.1016/j.proeng.2017.01.104

Most of the existing techniques for constructing software architecture are not well formalized and are usually not based on any mathematical theory (Bass *et al.,* 2013). Therefore, the problem of software architecture selection and analysis based on quantitative evaluation is very important. The analysis of architecture enables early prediction of a system's qualities. In other words, it would be desirable to have a formalized technique that is based on mathematical theory, and which allows the user to analyse and make decisions when choosing software architecture or its components.

Several techniques have been proposed to assist software architects in making architecture decisions (Aleti *et al.*, 2013; Falessi *et al.*, 2011). There are several groups of such techniques, where some of them focused on architecture trade-off analysis, quality evaluation model analysis, performance optimization and some others well-known techniques (Aleti *et al.*, 2013; Falessi *et al.*, 2011).

Some other studies propose the usage of the criteria of efficiency and the architecture efficiency metrics for quantitative evaluation of a software architecture structure (Orlov and Vishnyakov, 2010; Orlov and Vishnyakov, 2014). The disadvantage of this method is that the components of the architecture efficiency metrics are explicitly defined, and we cannot easily extend them to reflect the required software architecture features.

In this paper, we propose a technique that allows us to make architectural decisions when creating structure of a software architecture using set of architectural patterns. In other words, this technique allows us to choose the best structural organization of the software architecture that is build using the architectural patterns.

The proposed technique is based on the so-called criteria importance theory for decision making problems with a hierarchical criterion structure (Podinovski, 1994; Podinovski and Podinovskaya, 2014). It allows decisions to be made when choosing a software architecture system from among several alternatives and lacks the disadvantages that exist with other methods.

2. Model definition for software architecture selection

To create the technique for selectingthe optimal software architecture, we build a model based on criteria importance theory for decision-making problems with a hierarchical criterion structure (Podinovski and Podinovskaya, 2014). To applying it, we need to pick a set of metrics that assess the characteristics of the software architecture. These metrics are then used to build a hierarchical criterion structure. Next, we need to determine the metrics scale and information about their importance.

The mathematical model of individual decision making for multiple criteria includes the following components:

- set of alternative software architectures X;
- vector criterion$f$;
- preference and indifference relations of the decision maker (DM), which are denoted as P (preference) and I (indifference).

Each alternative x from the set of alternatives X is characterized by a number of criteria f_i,i = 1, ...,m, which are called *particular criteria*. The ordered set of such criteria forms a *vector criterion* $f = (f_1, \ldots, f_m)$.The *criterion* f_i is the function defined on X and taking its values from z_i which is called a common scale (or number of assessments of such criterion). According to the standardsoftware engineering terminology, we callparticular criteria metrics.

Assume that we have a number different of software architectureoptions. For example, the architecture of some transport system can be implemented based on Service-oriented pattern; alternative architecture could be based on a Multitier pattern; another with the use of a Microservices architectural pattern and so on. We define a set of such alternative software architectures as $X = \{X_i \mid i = 1, \ldots, n\}$. Let us assume that all alternative metrics are homogeneous, i.e. they are all measured using the same scale and have the same range defined as Z_0. Suppose that the number of scale gradationsis finite, then: $Z_0 = \{1, \ldots, q\}$, where $q > 1$.

In other words, each metric f_i from the set of alternative software architectures X can take the values from the set of scale gradations Z_0. Assume that all estimates are expressed in numerical form and higher values are preferableto smaller ones. Thus, each software architecture alternative X_i characterized by values $f_i(X_i)$ of every metric and forms its *vector estimate* $y = f(X_i) = (f_1(X_i), \ldots, f_m(X_i))$. Alternative software architectures are compared by comparing their vector estimates. The set of all possible vector estimates is defined as$Z = Z_0^m$.