BPEL-based Usage Profile Construction

Chao FENG¹*, Hong ZHANG¹, Cong PAN¹ and Ya-ru GUO²

¹Beihang University, 37 XueYuan Road, HaiDian District, Beijing, China
²Lanzhou University, 222 TianShui Road, ChengGuan District, LanZhou, China

*Corresponding author

Keywords: BPEL, Usage profile, Similarity, Probability distribution.

Abstract. With the increased use of Software-Oriented Architecture in financial, military and other high-reliability fields, the reliability problems of SOA software systems need to be solved. Software reliability testing is an effective way to ensure the quality of software systems, and the usage profile or operational profile is the basis for software reliability testing and even software reliability engineering. This paper presents an overall framework and a method of constructing usage profile for SOA software systems, so as to guide the reliability testing and assessment of SOA-based software systems. We use this method to extract the process of operations from the BPEL to form the main structure of usage profile that we call the usage structure for further use. Then, a method of probability distribution for usage structure is presented to form the whole usage profile. This method can be used to model not only the SOA but other distributed systems.

Introduction

With the software part of various systems gradually becoming the dominant part and the complexity and scale of software systems continuing to grow in recent years, the Software-Intensive System (SIS) is becoming an important system form. Thanks to the characteristic of flexibility, Service-Oriented Architecture (SOA) [1, 2, 3], as a kind of SIS, shows obvious advantages in response to complexity and heterogeneity, and is rapidly changing the present and future of software engineering.

At present, in terms of the system’s testing for the research and implementation of SOA systems, we focus more on unit testing, service combination testing and functional testing. For reliability, most articles are based on the establishment of model to predict the reliability of SOA system. The unique functionality of the BPEL [4, 5, 6] generates new challenges for testing. Hou S S [7] presents a testing method of BPEL by generating a series of valid message sequences. The author models the BPEL process to a message sequence graph (MSG), and generates message sequences based on MSG to find fault. In order to capture the BPEL control process, Liu C H proposes a structural testing method based on the BPEL and models the Web service composition for testing [8]. Specifically, the model contains synchronization and concurrency characteristics of the BPEL, and describes possible execution flows by using the BPMN.

However, up to now, there is no article to propose solutions to get the reliability index of the whole system through the SOA system’s overall reliability testing with the help of usage profile or operational profile. The overall idea of our method is: firstly, to construct the SOA system’s usage profile [9, 10, 11], and then use the profile to generate test cases for the SOA system’s reliability testing work, so that we can not only find the fault but implement reliability assessment [12, 13] of the SOA system, and ultimately get the reliability of the entire system. The focus of this paper is to construct usage profile of the SOA system, and in order to get more accurate usage data of the business process, especially to focus on solving the problem of similarity between the previous version and latest version to be tested of SOA system.
In this paper, we divide BPEL document into a series of modules and extract structure relationships [14, 15, 16] between operations, accompanied with the extraction of the interface, variables and other information that the reliability testing needs to form profile elements. The next step is to combine these profile elements into the whole structural model of usage. The last step is to make probability distribution for usage structure to form the whole usage profile. The method is intended to be part of a more general framework for the reliability testing and assessment.

Overall Framework

Figure 1 shows an overall framework of the method to develop a usage profile for the SOA from BPEL. The rectangles represent processes and rounded rectangles indicate inputs/outputs.

In Figure 1, the construction process of usage profile can be divided into four steps: (1) decomposition of BPEL documents; (2) transformation for profile elements; (3) combination of profile elements; (4) distribution of probability. The former three steps present construction method of usage structure. The second step is divided into the transformation of basic and structural activities from blocks of the BPEL into profile elements.

Specifically, our contribution is to extract the operations and relationships [17, 18, 19] (sequence, concurrency, loops, etc.) between operations from the BPEL of SOA softwares, and then construct usage structure. As for the probability distribution of usage profile, we can obtain it from log information of the SOA software, and predict future usage of the software to get expected probability distribution. The two methods can be combined together to get the software probability distribution further close to actual usage of the conversion method.

Method of Usage Structure Construction

Since the BPEL document has modular program blocks and clear structure, we can naturally extract operations from BPEL on the basis of modules. The introduction of method of usage structure construction is divided into four parts as follows: (1) decomposition of BPEL documents; (2) transformation of basic activities; (3) transformation of structural activities; (4) combination of profile elements.

- Decomposition of BPEL documents
  Each business process has a major activity, and a BPEL activity can be any of the following:
When decomposing and mapping the elements of the BPEL into the elements of usage profile, we will identify these activities one by one as well as all information involved in these activities, such as the operations, ports of operations, relationships between operations, variables related to operations and so on.

- **Transformation of basic activities**
  The first step is to convert basic activities, such as receive, reply, invoke, assign and so on, to derive the profile elements. We are concerned with some information about the operations of basic activities, and translate it into elements of usage structure.

- **Transformation of structural activities**
  Generally, a structural activity contains not only the basic activities but other structural activities. The structural activities, such as sequence, flow, while, repeatUntil, if and so on, can be classified into subsequence, subsequent, branching, circulating structure.

- **Combination of profile elements**
  In fact, we have already completed the main work of the transformation task up to now. But we only get a series of separate blocks of profile elements, after which we need to combine these independent profiles together as a whole structure of the profile based on the relationships between profile elements. In general, the process of combining profile elements to the usage structure of the profile is carried out according to the chronological order of which the BPEL process is executed. As for the call relationships between services, we will reflect it in the description of the operations to guide the subsequent reliability testing.

### Distribution of Probability

How to calculate the transfer probability accurately between operations to reflect the actual use of the software has always been a problem that is worthy of in-depth discussion. At present, the more popular probability distribution methods are "statistical method", "experience method", "expert scoring method" and Musa hierarchical division method. From the point of view of engineering application, the more feasible method is to calculate the transfer probability by analyzing the actual running data of the software because the actual log information of operations reflects the real use of the system. Specific steps of distribution of probability are as follows:

- **Log collection**
  The collected log information is in a timely manner and should reflect the average usage of the SOA system. The contents and formats of different log may differ, but the content should be detailed enough to extract the operational information. For convenience of analysis, all kinds of logs should be converted into a unified format.

- **Log preprocessing**
  The purpose of log preprocessing is to select the normalized data from the original log file for use in sequence pattern mining. The main task of data preprocessing is to remove incomplete paths, noise, and other data that is not related to the profile structure, so as to achieve the purpose of simplifying the log and improving the efficiency of the log analysis process.

  The log contains some pictures, audio, video and other information that is useless for construction of usage profile. The log information, which contains the suffix jpeg, jpg, wav, gif, js and other log records, should be deleted in the preprocessing. In addition, the log records with error states code which is not between 200 and 299 should be deleted. The last step of log preprocessing is log reordering based on the user’s IP.
• Operation sequence mining
An operation sequence mined from the log corresponds to an operation flow in BPEL. The purpose of operation sequence mining is to get all operation sequences and the number of each operation sequence. Then, calculate the probability of occurrence of the operation sequence. If the number of each operation sequence is \( n_1, n_2, n_3, \ldots, n_n \), and the sum of all the operation sequences is \( N \), the probability of occurrence of each operation sequence is calculated according to the formula \( P_i = \frac{n_i}{N} \).

• Calculation of transfer probability
If the set \( M \) represents the operation flow and probability of the operation flow:
\[
M = ((\text{opp}_1, p_1), (\text{opp}_2, p_2), \ldots, (\text{opp}_n, p_n))
\]
Assume that the probability transfer matrix of the set of operations \((\text{op}_1, \text{op}_2, \ldots, \text{op}_{n-1}, \text{op}_n)\) is:
\[
T = \begin{bmatrix}
    p_{11} & \cdots & p_{1n} \\
    \vdots & \ddots & \vdots \\
    p_{n1} & \cdots & p_{nn}
\end{bmatrix}
\]

The transfer probability \( p_{ij} \) between the operation \( \text{op}_i \) and \( \text{op}_j \) can be calculated according to the sequence, selection, concurrency, and circulation structure between the operation \( \text{op}_i \) and \( \text{op}_j \):

- **Sequence structure**
  \( \text{op}_i \rightarrow \text{op}_j \rightarrow \text{op}_k \)
  The sequence transfer probability \( p_{ij} = p_{jk} = 1 \).

- **Selection structure**
  \( \text{op}_i \rightarrow \text{op}_j \) or \( \text{op}_i \rightarrow \text{op}_k \)
  Assuming the transfer probability is \( p_{ij} \) and \( p_{ik} \), and the data obtained by log statistics are \( n_{ij} \) and \( n_{ik} \) respectively, then

  \[
p_{ij} = \frac{n_{ij}}{n_{ij} + n_{ik}}, \quad p_{ik} = \frac{n_{ik}}{n_{ij} + n_{ik}} = 1 - p_{ij}
\]

- **Concurrency structure**
  \( \text{op}_i \rightarrow \text{op}_j \rightarrow \text{op}_m \parallel \text{op}_i \rightarrow \text{op}_k \rightarrow \text{op}_m \)
  The operations \( \text{op}_j \) and \( \text{op}_k \) are executed concurrently. Then, the transfer probability \( p_{ij}, p_{jm}, p_{ik}, p_{km} \) related with the operations \( \text{op}_j \) and \( \text{op}_k \) are 1.

- **Circulation structure**
  \( \text{op}_i \rightarrow^* \text{op}_j \)
  Assuming that the self-loop probability of operation \( \text{op}_i \) is \( p_{i^*} \),

  \[
  \begin{align*}
  \text{op}_i \rightarrow \text{op}_j & \quad n_1 \\
  \text{op}_i \rightarrow \text{op}_i \rightarrow \text{op}_j & \quad n_2 \\
  \text{op}_i \rightarrow \text{op}_i \rightarrow \text{op}_i \rightarrow \text{op}_j & \quad n_3 \\
  \quad \vdots \\
  \text{op}_i \rightarrow \text{op}_i \rightarrow \cdots \text{op}_i \rightarrow \text{op}_j & \quad n_n
  \end{align*}
  \]

  Ideally, \( 1/p_{i^*} = n_1/n_2 = n_2/n_3 = \ldots = n_{n-1}/n_n \).

  The probability values \( p_{i^*1}, p_{i^*2}, p_{i^*3}, \ldots, p_{i^*m} \) are calculated according to the formula \( 1/p_{i^*} = (n_{m-1})/n_m \), and then these probability values are plotted in scatter plot. The horizontal line \( L \) can be regarded as the value of the probability \( p_{i^*} \) when the points are evenly distributed on both sides of line \( L \).
Summary

We have presented a systematic method for developing the usage structure of usage profiles for the SOA using BPEL. Furthermore, a method of probability distribution has been presented.

Our method can be used to model not only the SOA software systems but other distributed systems. Furthermore, the usage data derived from previous version of SOA system can be applied to the system to be tested with similarity method.

The future work is to derive test cases based on the usage profile to test the SOA system and calculate the reliability value of the system.

Acknowledgement

I would like to express my gratitude to all those who helped me during the writing of this paper. A special acknowledgement should be shown to Professor Min-yan Lu, from whose fund project I benefited greatly. Thanks are also due to Prof. Zhang for assistance and Mr. Pan for valuable discussion.

References


