Research on the Implementation Technology of Generic Codec for Web Application Testing

Yongpo Liu  
School of Computer Science and Engineering,  
BUAA  
Beijing, China  
e-mail: liuypo@sei.buaa.edu.cn

Shuangmei Liu  
College of Computer Science  
Qingdao Agricultural University  
Qingdao, Shandong, China

Ping Li  
Information Technology Center  
Daqing Petrochemical Corporation  
Daqing, Heilongjiang, China

Ji Wu  
School of Computer Science and Engineering,  
BUAA  
Beijing, China

Abstract—This paper presents the design of generic codec for testing Web application. Codec is a necessary part of TTCN-3 test system which transfers from TTCN-3 template data and SUT (the system under test) input data, and from SUT output data to TTCN-3 template data. The generic codec needs to transform TTCN-3 template into URL request string, analyzing the Web page, and then extract the Web page. Because of conscious of the Web pages with large volume of data and noise data, the multi-level parsing and composition strategy has been presented in this paper. Experiments show that the strategy works well and could save much codec development effort.

Keywords—TTCN-3; test system; Web application testing; codec; multi-level parsing

I. INTRODUCTION

TTCN-3(Testing and Test Control Notation 3rd) is a test specification and test standard developed and promoted by ETSI (European Telecommunications Standard Institute). As a test description specification with rich ability, it is based on black-box [1]. And it can be applied to various forms of distributed systems. As the test technology based on TTCN-3 develops, it has been increasingly applied to all kinds of testing fields. And it has made great achievements in the telecommunications, automotive, networking and other areas. By using it, a series of test systems has been developed. The development of test system based on TTCN-3 has the similar features as software development [2]. But as the test system grows in size and the testers change, for the huge test systems, the management and maintenance have become more and more difficult. Therefore the reverse engineering based on TTCN-3 test systems can help testers grasp the system design from higher levels, and can test the consistence between test design and test implementation, which is of great significance and important value for test system maintenance, expansion and evaluation.

Because of its wealth of data description methods, TTCN-3 is suitable for testing the software based on data exchange. On the basis of the application of TTCN-3 in many fields, the author has carried out the research on the testing object to be measured to Web application and desktop application software. By the load testing, function testing, API testing on the Web application, and the GUI testing, function testing and API testing on desktop software, a series of experimental data has been obtained on the test development test and test code size [3]. After analyzing the data, the applicability of TTCN-3 was analyzed and evaluated in three aspects, namely the language specification, the system under test and test developers. The research shows that both the TTCN-3 and its support tools are more suitable for testing the Web application whose core is the data exchange that is the theoretical principle of this paper. But when building the testing system for Web application, there is a large of codec development workload. The encoding and decoding rules of different Web pages are very different. Therefore it has important practical value and prospective in studying the design and implementation of the generic codec for Web application testing system.

II. TTCN-3 TESTING SYSTEM

TTCN-3 is a system that can complete the specified test behaviors by interacting with the software under test. It includes six entities and two interface standards, shown as Fig. 1. Every entity corresponds to the function of the testing system.

Figure 1. the implementation diagram of TTCN-3 entities

- TTCN-3 Executable (TE), it can execute test codes.
The implementation process is as follows.
1. When the testing system executes the Send of TE, it firstly calls the Encode of CD in order to code the parameters from Send.
2. The incoming data (raw excitation data) will be coded by CD. The coded data will be back to TE.
3. The testing system sends the coded excitation data to the system under test by calling triSend of TA [6].
4. TA sends the excitation data to the system under test by using HTTP.
5. After the system under test deals with the excitation data, it transmits the response data back to TA by using HTTP. Because TA can receive the response data, it hasn’t the method named Received.
6. TA transmits the raw response data back to TE.
7. The testing system decodes the raw response data to the format understood by it by calling the Decode of CD.
8. CD will send the decoded data back to TE aiming at being compared.

IV. THE IMPLEMENTATION SCHEME OF CODEC
The research shows that TTCN-3 is suitable for Web application testing, but there are also problems. Because all the pages of Web application are not same, it is necessary to design a codec scheme corresponding to each page. Therefore, in the development of TTCN-3 testing system for Web application, writing CD codes becomes the most complex and difficult work. However, in spite of different page contents, the structure characteristic of the page script is the same. So parsing the page script is general, which makes CD having domain generality. On this basis, providing a generic codec for Web application can improve the efficiency of the testing an development work, and can also enhance the applicability of TTCN-3 in the Web application testing.

As mentioned above, the SUT response that TA receives is a complete Web page to the TTCN-3 testing system. Its data format is HTML code, which implies the data that the
testers concern. CD parses the Web page according to the data definition of TE, and extracts the data, then converts its format. Different from testing the Web application with traditional C/S structure, the objects concerned by testers not only include data elements (that is the response results from the application to input data), but also include the page elements (that is the form of Web pages, such as the form, the text box etc.) in the course of testing the Web application with B/S. Therefore, the data which TE expects to receive is defined as a hierarchical structure, which can effectively describe the target data. The relevant research had been carried out by Pulei Xiong [7], shown as Fig. 4.

Because Frame, Form, Select, Input and Hyperlink are the common elements of Web pages, they are selected as WebElement, and other special or unique elements are classified as OtherElement. DataElement describes the specific data. It can improve the generality to define the expected receiving data by combining with WebElement, OtherElement and DataElement. The following code gives a specific definition.

```
module ResponseType {
  type set DataElement {
    charstring DataField,
    charstring DataValue
  } type set of DataElement DataElements;
  type set Frame {
    charstring FrameName,
    charstring FrameSrc
  } type set of Frame Frames;
  type set Form {
    charstring FormName,
    charstring FormAction,
    charstring FormMethod
  } type set of Form Forms;
  type set Input {
    charstring InputName,
    charstring InputType
  } type set of Input Inputs;
  type set Select {
    charstring SelectName,
    charstring SelectId
    optional,
    charstring SelectClass
  } type set of Select Selects;
  type set Hyperlink {
    charstring Href,
    charstring Target optional
  } type set of Hyperlink Hyperlinks;
  type set WebElements {
    Frames frames,
    Forms forms,
    Inputs inputs,
    Selects selects,
    Hyperlinks hyperlinks
  } type set OtherElement {
    charstring OEField,
    charstring OEValue
  } type set of OtherElement OtherElements;
  type set Response {
    DataElements dataElements,
    WebElements webElements,
    OtherElements otherElements optional
  }
}
```

In the above code, `module` is the top element of TTCN-3 code. Test data, configuration and behavior are defined in it. `Set` is an out-of-order structured type, which is similar to the `Struct` type of C. `Set of` is a collective description whose elements are `Set`. In the code, DataElement, Frame, Form, Input, Select, Hyperlink and OtherElement define seven basic elements of end nodes in Fig. 4. They contain the name/value pairs from respective attributes. DataElements, Frames, Forms, Inputs, Selects, Hyperlinks and OtherElements are the set of all the corresponding basic elements. WebElement is a set of Web elements. `Response` contains three sets of data elements, Web elements and other elements, which is corresponding to the entire page. The `Optional` of the code indicates that the item is optional, and may not exist in real data.

V. COVERSION OF TARGET DATA

This paper focuses on the generality which implies a certain degree of abstraction. CD is a complete unit of work which has specific input and processing corresponding to the input and output. Among these, the input is determined by the system under test, the corresponding output is different because the system under test is varying. From another perspective, although the input content of CD (that is original data) is complicated, it still has some similarities as a Web page, which determines the common codec should be reflected in the data processing method. Because the parsed Web page has complex structure, a large amount of information and interference information, the efficiency is very low if the received data is directly extracted from raw response data. According to the above characteristics, this paper adopts two-layer analytical strategy to decompose the problem. The target data will be obtained through two times
of continuous analysis, data format conversion will be completed by the assembly, shown as Fig. 5.

![Diagram](Image)

Figure 4. two-layer analytical strategy

The corresponding parsing rules are customized according to the different system under test, which can deal with the differently raw data. XML is a standard markup language, and has a standard method to describe the structured data. So the parsing rules are described by using XML, and the description will be input into each layer of parsing function module in the form of a document. The data flow of CD is represented as Fig. 6.

![Diagram](Image)

Figure 5. the flow chart of CD

Aiming at different systems under test, the analytical rule description file is only modified to parse the different targets by using the two-layer analyzing. The two-layer analyzing and assembling both implements the decode method of CD. The role of the first layer analysis is to filter irrelevant data, parse the raw data according to the data elements mentioned above, and form the data object set corresponding to data element, Web page element and other elements. The first layer analysis is achieved by using the open source or library from the third party, such as HTML Parser, JTidy. The second layer analysis aims at the specific data elements contained in the target data from every data object set. Assembling aims at making the elements from the second layer analysis into the target data with the same data format as the desired receiving data, which can be done TCI interface that has function of the data conversion and data packaging. The expected receiving data defined in the above code is as an example. Seven data sets can be obtained after the first layer analysis, namely Data, Frame, Form, Input, Select, Hyperlink and other elements. The second layer extracts the specific elements in the desired receiving data which provides the material for assembly. The target data can be formed after assembly.

Of course, the design also exist some problems. Paring a Web page requires to input two analytical rule description files. If the system under test is a large scale of Web site, there will be a large number of the analytical rule description files. Therefore, it is necessary to abstract the parsing rules in a certain degree, which can form the predefined parsing rules with model features. The predefined parsing rules are universal, which reduce the number of parsing rule description files and enhance the universality of encoder.

VI. SUMMARY

As a test standard, TTCN-3 is platform-independent and universal, which has been used in more and more fields, and is studied and extended both at home and abroad. From the TTCN-3 specification itself, both its core language and interface are the good foundations in every test application field. On the other hand, the TTCN-3 support tools provide the strong support for the spread of TTCN-3, such as TTworkbench based on Java and Exhaustif/TTCN based on C++. At present it is deficient in the relatively mature development methods of TTCN-3 testing system, this paper proposed a development method of testing Codec for Web application. Because of the large quantity of data and interference information of Web application, the raw data is processed by using the multilevel parsing, and then the target data will be acquired.

The purpose of the study is to propose a universal scheme for Web application software testing system based on TTCN-3, which helps to enhance the applicability of TTCN-3 specification in Web application software testing field. Therefore, the design of this method has been completed. The next step is to achieve the specific functions. In addition, this method is now focused on the data which is right or wrong, and the location distribution of Web page elements has not been considered, future research will focus on it.

ACKNOWLEDGMENT

This paper is contributed by National Key Laboratory of Software Development Environment Fund (SKLSDE-2010ZX-15) and High Technology Research and Development Program of China 863 Fund (2009AA01Z145).

REFERENCES


