

XML-based E2E Test Report Management

Ray Paul

Department of Defense, Washington, DC 20016

Wei-Tek Tsai, Bing Li, Xiaoying Bai

Department of Computer Science and Engineering
Arizona State University, Tempe, AZ 85287

Abstract. This paper presents an XML-based test report management system for end-to-end (E2E) integration testing. This system allows test engineers at different sites using different platforms to collaborate and exchange testing effort by using XML-based communication. The system also supports report creation, storage, search, version management, and project management.

1 Introduction

Testing especially integration testing has been a serious challenge for system development. To address this problem, the U.S. Department of Defense (DoD) recently proposed an End-to-End (E2E) testing process to test an integrated system [DoD 2001, Tsai 2001a, Tsai 2001b]. This E2E testing and addresses test specifications, test case generation, risk analysis, regression testing [Tsai 2001b], ripple effect analysis, test schedules, test planning, and test result analysis. We developed a web-based system to support the E2E testing [Bai 2001]. This paper focuses on test report management in the testing tool using XML. Test data include test specifications, design, results, and reports.

The class diagram of test assets in the E2E tool can be interchanged through XML. Using XML, engineers can share and exchange test assets independent of any machine platforms. Tools and repositories that need to exchange metadata do not need to implement a new API, which can be expensive, or to re-host to a new repository, which can be costly too. They need to provide only an XML export and import facility.

2 XML-based Reporting Subsystem

Figure 1 shows the architecture of the test report subsystem. On the client side, Report Format Customizer and Report Query Customizer accept users' requests, and Report Specification Generator generates specifications of the requests. On the server side, Report Generator, in Application Server, accepts requests from clients, and generates reports based on data stored in the test asset database, Web Server publishes reports

and enables users to remote access reports through web-browsers; Report Builder periodically generating reports to ensure that reports are current, Report Analyzer compares different reports and generates analysis reports.

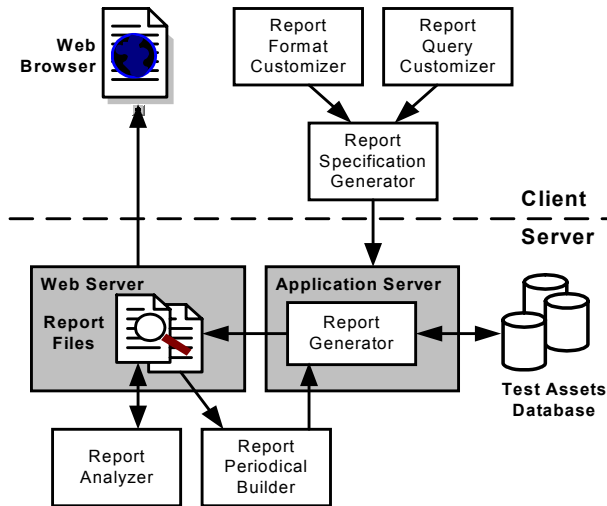


Figure 1 Structure of Test Report Subsystem

The main issue in test report management is that report definitions are highly dynamic and change often. Users may define different report structures by selecting various combinations of assets and asset attributes. They may also define different report representations. Traditionally, each test report must be coded individually. In this way, one must repeat work such as designing database structures and storage. This is costly. Our test report subsystem avoids this problem. When one needs a new test report, the only thing to do is to describe the new report's model in DTDs and submit the DTDs into the subsystem. No additional programming is needed, and all the relevant design work such as creation, searching and storage management can be reused.

References

- [Bai 2001] X. Bai, W. T. Tsai, Ray Paul, Ke Feng, and Lian Yu, "Distributed End-To-End Test Management", to appear in Proc. of IEEE EDOC, 2001.
- [DOD 2001] DoD OASD C3I Investment and Acquisition, "End-to-End Integration Testing Guidebook", 2001.
- [Tsai 2001a] W.T. Tsai, X. Bai, R. Paul, W. Shao, V. Agarwal, T. Sheng, and B. Li, "End-to-End Integration Testing Design", to appear in Proc. of IEEE COMPSAC, 2001.
- [Tsai 2001b] W.T. Tsai, X. Bai, R. Paul, and L. Yu, "Scenario-Based Functional Regression Testing", to appear in Proc. of IEEE COMPSAC, 2001.
- [Tsai 2000] W.T. Tsai, X. Bai, B. Huang, G. Devaraj, and R. Paul, "Automatic Test Case Generation for GUI Navigation", in The Thirteenth International Software & Internet Quality Week, 2000.