Computer Science

Opponent(s):

Margaret Mburu and Robert Josefson

Respondent(s):

Peter Hjärtquist and Erik Möller

Robustness Test on CPPemu

Opposition Report, C/D-level

2005:xx

1 A General Evaluation of the Project

Generally this report was well structured and explains a complicated topic in a simplified manner for a recipient with no or little prior knowledge of the subject to understand. This project was to examine whether the CPPemu could be used for testing software instead of running tests using physical hardware. The CPPemu is a tool developed by Ericsson to emulate the hardware used in the network nodes of the phone system CPP.

The experiment involved generating a fail-over. In this case a fail-over is defined as a *failure and automatic replacement of part of a system such that the user does not notice the failure and is not affected by it. The part which has failed is replaced by a backup part.*

The main tools which were used in this project were: LOCO to generate traffic over the network, Cppemu which is the emulator used to emulate a physical CPP node and the Mo-Shell which is a command line interface used for running commands and administrating the Managed objects in OSE. These tools were used together to run traffic over a network for testing of fail-over.

The experiment was performed with six different measurements for three different test cases. The results of the test cases are then presented as graphs. In order to conclude if the fail-over worked or not, three parameters which were used by the test cases were examined.

The project was implemented by running three different test cases. The results from the test cases were presented in graphs. The expectation of the result was that since fail-over works on the CPP node in a physical environment it should also work in an emulated environment. The result of the experiment was according to the expectations.

Most of the time was spent on learning the systems used for the experiment and configuring the system and the environment. The project has shown that a fail-over works on a broad band connection. Future work may be based on examination of whether a fail-over can occur using the same configurations and test cases on other connections such as a High Speed Link (HSL).

2 Comments on the Project in Relation to the Dissertation

The implementation of the project has been very well documented in the report. It has defined the problem, process, implementation and evaluation and described them in a manner which makes it easy for the reader to understand with little or no knowledge about the topic. However it is unfortunate that some of the results could not be disclosed due to confidentiality reasons.

2.1 Title

The title of the project conforms to the dissertation. The word robustness suits the project well since robustness when used with regard to computer software, refers to an operating system or other program that performs well not only under ordinary conditions but also under unusual conditions that stress its designers' assumptions. However the title could be altered to reflect the context of testing.

2.2 Dissertation Layout

The layout is good and well structured.

Chapter 1, the introduction to the project which describes the tools being used and the goals set for the project.

Chapter 2 describes the background of the project, purposes and tasks that where to be performed and finally detailed descriptions about the tools used.

Chapter 3 describes approach of the experiment. Outlining how it was performed and a detailed description of the implementation.

Chapter 4 presents the results of the project. Six measurements were performed on three different test cases. The results were presented in diagrams however the values of the x-axes and y-axes were not disclosed for confidentiality reasons. Therefore it is hard to interpret the exact results of the project. Further, chapter 4 presents the evaluation of the results and conclusions drawn from the results.

Chapter 5 presents the conclusions for the entire project. It also presents the reason why the test was performed, the project evaluation and future work.

2.3 Scientific Method

The experiment was to examine the hypothesis on whether the CPPemu could be used for testing software instead of hardware. The scientific method used was to set up a redundant network between two emulated nodes, writing test scripts for the network, generating and sending traffic through the network and testing the fail-over by ejecting a board which is currently sending and receiving traffic and inspecting the log files to evaluate if the fail-over works as intended

2.4 Argumentation and Conclusions

The report clearly states why the experiment was performed and how it was implemented. The reason for the experiment was that TietoEnator would like to examine whether the emulator may be used for testing software instead of running tests using physical hardware.

2.5 The Abstract

The abstract has been kept short and underlines the most important issues related to the project. It describes in short terms the background of the project and the goals of the project. Further the abstract describes in short terms how the project is implemented.

2.6 Language Aspects

The language is simple and understandable.

2.7 References and Sources

The terms used in the document are well referenced. However some of the references are not accessible such as reference [4] which directs us to a page stating the article we where looking for does not exist. The same applies to reference [6]. When we try to access link an error is displayed “**error 404 Page not found”**. The same problem applies to internal reference [14] which contains a hyperlink to a page which cannot be displayed. Our concern is to know whether the internal references such as [9] CPP emulator User Guide R12C are available or classified.

2.8 General Comments on the Project

Generally we feel the project is well documented. However regarding the test scripts it is not clear whether they were pre written or created during the project. The abstract indicates that writing test scripts was part of the task, but the description of the implementation of such scripts could not be found in the report. However, the report contains a number of technical terms, some of which are not well referenced. For example the internal references are not accessible. This makes it hard for us as readers who have no prior knowledge to understand certain areas and tools involved. Further in some areas we are a little uncertain about what has been done such as the case with the test scripts. The question is whether they were made particularly for this project or they were provided by TietoEnator.

3 Chapter by Chapter Evaluation of the Dissertation

3.1 Chapter 1

Chapter 1 is a brief introduction of the project. The project was to examine whether the CPPemu could be used for testing software instead of physical hardware. The CPPemu is an emulator developed by Ericsson for the CPP phone system. The experiment was about testing a fail-over procedure which was to be tested and verified in the CPPemulator. To accomplish the task a number of subtasks was to be performed:

- Set up a network between two nodes inside separate emulators.
- Set up the logical network between the nodes.
- Send data between signaling point 1 and signaling point 4.
- Model a failure
- Evaluate the results

The experiment was divided into two subtasks:

- Examination of what happens when a link between two connected signaling points fails.
- Automation of the ability to eject and insert boards in the emulator. (This was subject to the availability of time at the end of the project).

Finally the chapter outlines structure of the report.

3.2 Chapter 2

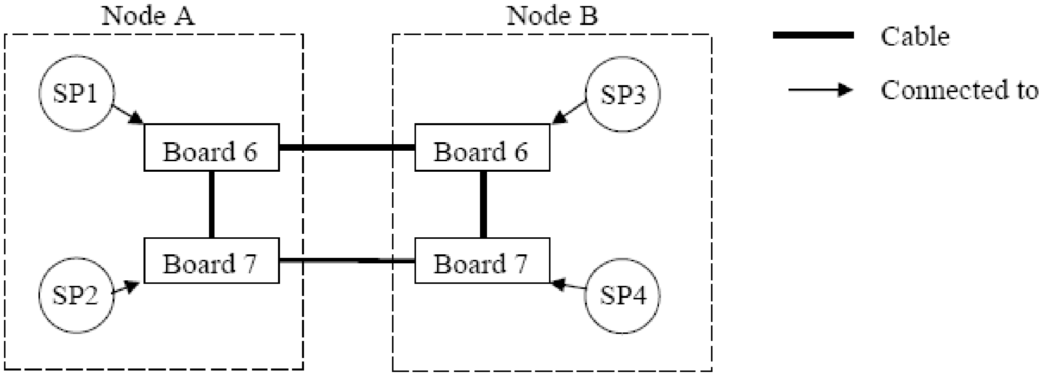
Chapter 2 presents a details description of the background to the project. TietoEnator wants to examine whether the CPPemu could be used for running tests on software instead of hardware. This could lead to several advantages for TietoEnator such as cost savings in terms of maintenance of software instead of buying new hardware.

The chapter describes all the tools involved and the main tools are:

- LOCO - A network traffic generation tool used for generating a specific load of traffic to a network during tests.
- CPPemu - Connectivity Packet Platform Emulator used to emulate a physical CPP node.
- Mo-Shell - Command line interface used for running commands and administrating the managed objects in the OSE (Operating System Embedded)

3.3 Chapter 3

Chapter 3 is the detailed description of how the experiment was implemented. The experiment is about generating a fail-over. The network consists of 2 nodes each containing 2 boards. Signalling point one is sending data to signalling point 4 through signalling point 2. The failure is simulated by the ejection of board 7, this forces signalling point one to automatically send data to signalling point 4 through signalling point 3 instead. This is illustrated in the diagram below.



The experiment was performed through the following steps:

- Two emulators are used and have to be connected to each other through at least two connections
- The desired network has to be configured and loaded onto the emulators.
- Traffic was run over the network and the failure was simulated by ejecting one of the boards and thus removing a connection.

3.4 Chapter 4

The experiment involved three test cases. The results and measurements of the experiment are presented as graphs in this chapter. However since some of the results are confidential the values of the x-axis and y-axis were not disclosed. In order to determine whether the fail-over worked or not, the effects of three parameters were examined. The parameters used were: packet size, packet intensity and number of packets. The expectation was that a fail-over should work in an emulated CPP node as it also works in a physical CPP node. The results from the three test cases matched their expectations.

3.5 Chapter 5

Chapter 5 contains a detailed discussion about what happens when a fail-over occurs. This chapter also contains the project evaluation and future work.

The main task of the project was to configure the network and get it up and running. However most of the time was spent learning the systems used. The project has shown that a fail-over works on a broad band connection. One future task is to examine whether a fail-over can occur using the same configuration and test cases on other connections such as a High Speed Link (HSL).