

Improved Quality and Reliability in Telecommunication Network Testing by means of a Remote Online Test System

<http://dx.doi.org/10.3991/ijoe.v10i4.3830>

C. Davutoglu¹, Numan M. Durakbasa², G. Bas^{1,2}, E. Guclu²

¹ IN-AUT Intelligent Automation Solutions GmbH, Vienna, Austria

² Vienna University of Technology, Vienna, Austria

Abstract—Increasing demand for high quality and reliable operation of the telecommunication network services and its applications have been the main challenges of the telecommunication network providers in the last decade. In this paper, a remote online testing facility (Intact – Intelligent Test Automation and Configuration Tool) is introduced that proposes a solution for telecommunication network providers to meet demands and to ensure business services as well as network related services (circuit switched and packet switched) operate efficiently.

This paper presents a particular solution of remote testing based on the automated functional “End to End” testing approach. “End to End” starts by configuring service properties, subscriber record and service logic on one hand. On the other hand several hundred call scenarios can be executed fully automatically. Moreover the call detail records written by the network elements are verified automatically too. Hence a test description and web automation language was developed which is very easy to learn. This testing approach improves test speed and quality leading to reduced risk and investment on test procedure. As a result, the proposed solution of a remote automated online test system aims to overcome higher cost, longer time to market periods and increased effort in testing.

Index Terms—Quality management, telecommunication network, remote automated test, end to end testing.

I. INTRODUCTION

The telecommunication network operators face challenges of providing the customers with reliable, faster and more cost-effective services that depend on developing and maintaining a consistent set of quality system requirements and measurement techniques [1, 2]. Development of the test systems has been a strategic approach in the sector in order to overcome the challenges and meet the requirements.

Integration and implementation of test systems comprise of a combination of various computing systems, telecommunication equipment and test engineers [3]. Existing test systems available for large scale telecommunication networks demonstrate disadvantages of being inefficient and high cost in operation.

The scope of this paper is to present a novel testing facility “Intact – Intelligent Test Automation and Configuration Tool” that proposes a remote online solution for the telecommunication network providers. The tool works

based on a particular solution of “End to End” test approach that starts by configuring service properties, subscriber record and service logic fulfilling automated test calls and verification of the call detail records.

II. THE TEST METHODOLOGY

The test methodology was established based on automation approach. The Intact was designed to overcome intensive and expensive testing. Hence, the targets were defined as:

- Test automation should improve quality.
- Test automation should be easy to run.
- Test automation should reduce risk and costs.
- Automated tests should be easy to write and maintain.
- Automated tests should require minimal maintenance even “System Under Test” (SUT) changes continuously.
- Test automation should provide Key Performance Indicators (KPIs) to determine status in an early phase of the project.

The tests are designed for both fixed and mobile network services that require huge amount of test complex test cases. The telecommunication services are defined in the Intact tool are mainly:

- Intelligent Network Services
- Service Node applications
- Supplementary services
- Private Branch Exchange services
- and any kind of telecommunication services which can be initiated by a smartphone.

During the test development phase, the problems that occur in the test automation are reported as:

- The used tools rarely provided an “End to End” view. In complex set-ups with various options and services automation tools cover individual functionality. There is no tool which covers all services or provides an “End to End” view.
- Many tools were expensive and they do not offer an interface to integrate the tools in a smarter environment.
- Integration of simulators and other tools into the SUT environment was always very complex and expen-

sive. Interfaces have to be simulated and it is always hard to prove if the simulation is erroneous or the service under test.

- Acceptance of the test and automation environment was nearly as cost effective as test of the SUT itself.
- Automation test suites are developer or tester focused but not focused on the user experience.
- Automation test suites where test cases have continuously to be updated due to lack of compatibility, especially for “Graphic User Interface” (GUI) related activities, e.g. record and play back.
- Test tools generally have their own language, syntax, graphical user interface, parameter definition and configuration. This makes it very expensive to create an “End to End” view for all use cases. A single automation engineer needs to know all of these languages in order to create “End to End” use cases.

Therefore, the automation test methodology of the Intact is established based on integration to the existing development cycle with the following approaches:

- Test automation starts in the requirements definition phase of a project. Please keep in mind “Use cases” are exactly the same as “Test cases”. When you define your “Requirements”, you can derive them from “Use cases”. And “Test cases” are “Use cases” with some detailed value settings.
- Test automation has a scope and target. There are many stakeholders in the company and project. It must be clear to every stakeholder which goals the company and/or project follows.

- Test automation needs good quality of documentation, especially in terms of good “Use cases”.
- Test automation needs “Proof of Concept”. If a test suite for an “End to End” test cannot be established/integrated within a short timeframe then there is something wrong!
- Be aware of next steps: Test automation must be embedded into the whole development process. Also future versions of the product have to be supported and changes to existing test cases have to be reduced to a minimum.
- Don’t try to automate 100 % of your test cases. 80% is good enough. 20% manual testing will keep the know-how of your test team up to date.

III. THE AUTOMATED REMOTE TESTING: INTACT

Basic philosophy of the Intact is to make life easier for all testers of telecommunication services as stated in detail Intact Handbook [4]. Intact is an “End to End” testing tool for people who are sitting in test labs and making one phone call after the other in order to test a telecommunication service. The idea was to create software which performs all actions a human being would perform during test of a telecommunication service beginning with provisioning and configuration, execution of the test case and taking necessary notes and reports following an “End to End” view. All this shall be performed automatically. Following graphical representation gives a high level overview of the remote online configuration of the test tool.

Intact is designed for “End to End” black box testing. Everything a telecommunication user can do via his mobile or a graphical user interface are automated (Fig.1).

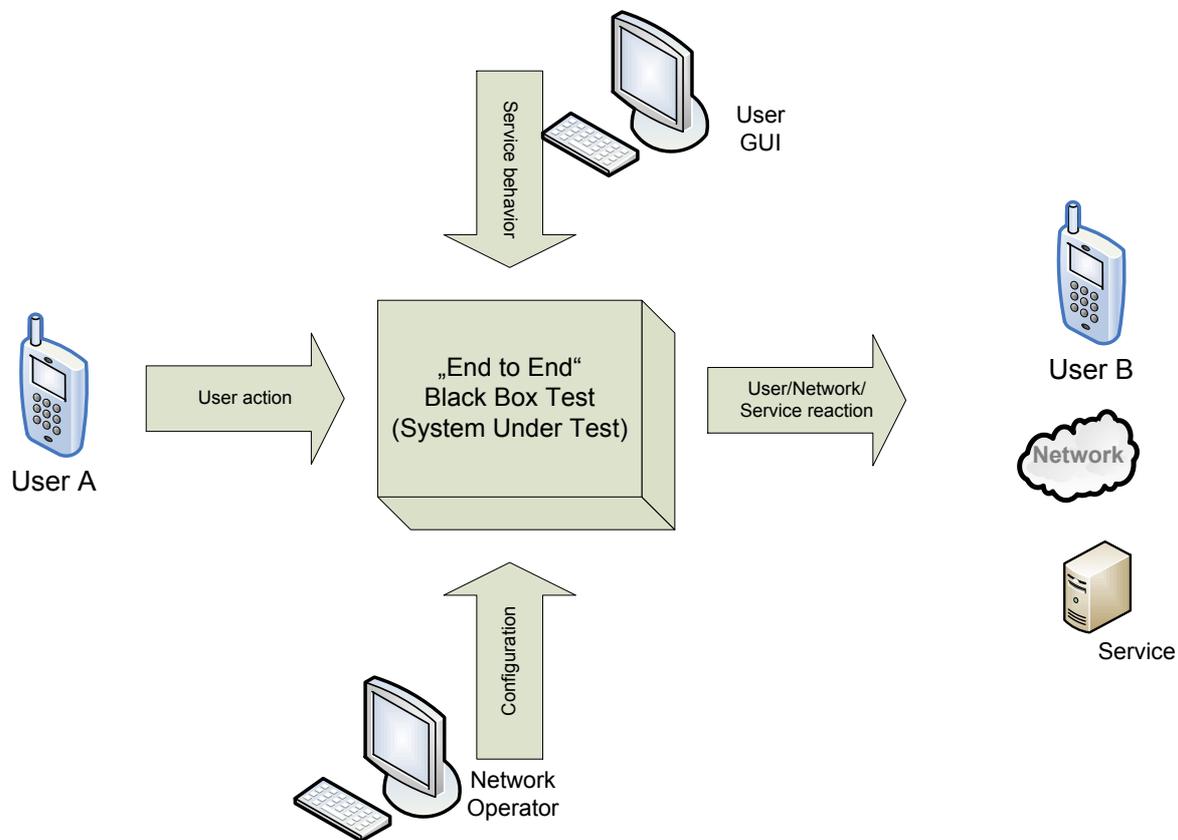


Figure 1. The “End to End” Black Box Test system

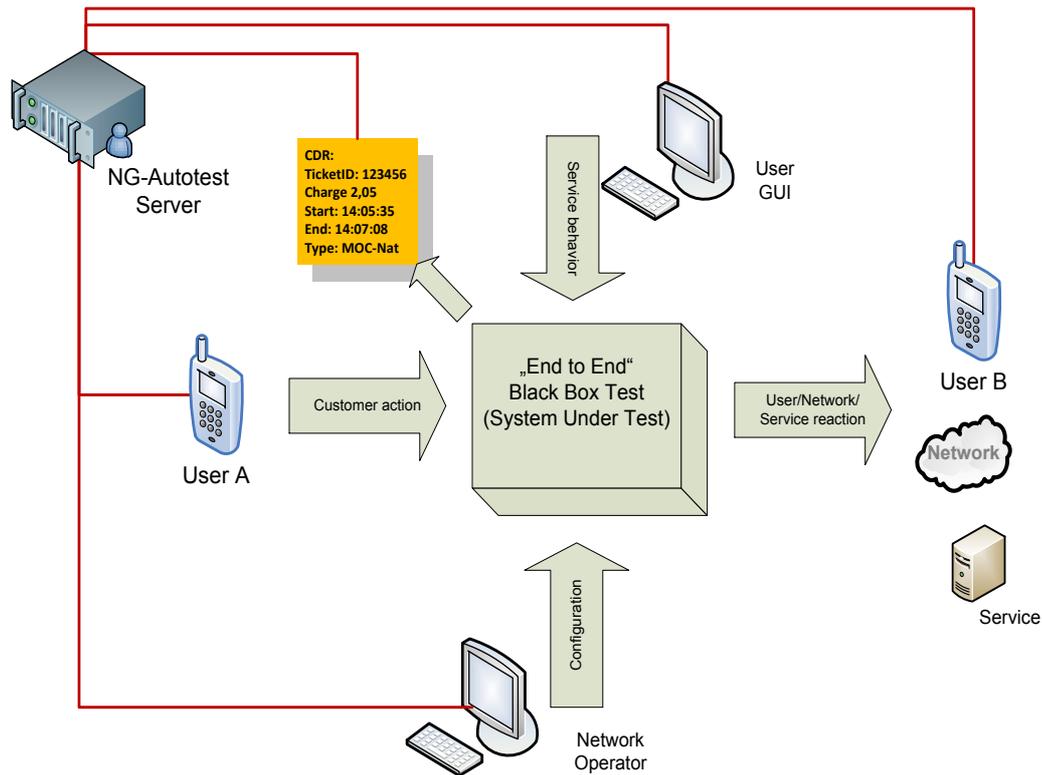


Figure 2. The overview of integration of the Intact

The basic philosophy is: “Reduce boring manual test activity to a minimum”. On the other side, a language and syntax are created in order to be able to test all telecommunication use cases, voice, SMS, USSD, MMS, IP-session, IP-event and provisioning activities via web browser or clients.

The second topic is: “Create a common environment, language and syntax for all use cases”.

The third aimed topic is “Keep the language simple to understand, readable and learnable for everybody”.

The Intact provides a wide range of functionality in order to overcome test automation problems of the past:

- All tools and frameworks integrated in the developed tool are license free.
- The tools were chosen where the coding
 - is modern and clear,
 - follows a main philosophy,
 - is object oriented
 - is tested by automated test tools itself
- It is concentrated that the effort address to a certain niche of telecommunication services and provide a very clear “End to End” view.
- A programming language which is very similar to English was developed which is easy to understand and easy to learn.
- Table driven input were provided via variables to replicate and execute examples (test cases) easily.
- Customizable reports and logs are provided.
- A simple alarming is integrated to support active probing respectively continuous testing.
- The set-up is very simple and does not require complex and expensive integration effort.

The main approach was to keep “System Under Test” untouched. Applications and adapted tools were developed in order to control user, customer and administration equipment and devices. The following figure represents a good high level overview of Intact’s integration (Fig.2).

Fig.2 presents clearly that Intact does not interact with system under test. It only interacts with the triggering equipment e.g. mobile and fixed network phones (customer premises equipment); graphical user interfaces; CDRs/EDRs.

Each test case / test script has a specific amount of phones in use. The simplest test case is A calls B. In order to configure it we have to define two phones: E.g. +436641234567 and +436641234568.

A major concept is the unpredictable selection of used phones. This means when there are two phones assigned, which of the phones will be A or B is not known. This is intentional. We want to keep it unpredictable so we can detect additional faults which one would never find by same behavior. In the first executed test A can be +436641234567 and B +436641234568 and in the next test execution it can be vice versa. Moreover, it is also possible to do a dedicated assignment.

IV. CONCLUSION

The basic principles of a telecommunication service are generally the same:

- It starts with a configuration action at customer or administrator GUI and ends:
 - In a system/platform configuration
 - In a change of subscriber data respectively customer profile

- It starts with an input (Voice, Video-Call/SMS/MMS/USSD/Browsing) at customer premises equipment (CPE) e.g. smartphone and ends:
 - In a call (SMS/MMS/USSD) to another party (calls with/without CF. CT, conference, etc.)
 - In a subsystem, e.g. mailbox or announcement machine
 - In another fault (USSD-Info, Top-up, Office-Net, VAS, Browsing, 3rd Party Service, etc.)
 - In a network fault (network info e.g.: ton or announcement)
- It starts with fetch of a ticket/record/log and ends:
 - In a verification of values

The Intact follows these basic principles. In principle we can define the Intact as a software program with an execution environment where soft bots (agents) are triggered by a test plan written in a common syntax (based on telecommunication terms in English). The soft bots themselves trigger telephones. Thus there are soft bots to control web GUIs. All actions triggered by the test plan are recorded and reported by a separate reporting and formatting module.

The test achievements during the test automation and testing are provided in the Intact to evaluate the quality at each step of the test. The useful metrics and key performance indicators that are used in the tests are:

- Level of Automation
- Automation Progress
- Test Progress
- Test Progress per time
- Testing Coverage
- Defect Trend Analysis
- Defect Removal Efficiency
- Automated Test Case Execution Efficiency

where:

Level of Automation % = (Number of automated test cases *100)/Number of total test cases

Automation Progress = Number of actual automated test cases/Number of test cases automatable

Test Progress % = (Successful completed test cases*100)/Number of automated test cases

Test Progress per time = Successful completed test cases/Working days or weeks or month(time)

Test Coverage = Test cases/Total amount of requirements

Defect Trend Analysis = Number of Defects/Test Cases Executed

Defect Removal Efficiency = Number of Defects Corrected/New Defects Found

Automated Test Case Execution Efficiency = Number of Executed Automatic Test Case/Time

With the quality management system integrated in the organization and the test automation system as stated in the ISO 9001(2008) and TL 9000 Quality Management System Requirements Handbook 5.5 (2013), continuous improvement is the basics of the whole process. The automation process is evaluated, documented and always aimed to improve quality to achieve the desired key per-

formance indicators. The international standard ISO/IEC 15504(2004) [5-9] under the general title Information technology-Process assessment is considered for this work as the framework with compliance to the rules stated in the ISO/IEC directives.

The metrics and key performance indicators of the system provide consistency, efficiency and quality while reducing redundancy and improve customer satisfaction. As a conclusion, it enables increased competitiveness together with improved quality in operation.

REFERENCES

- [1] Quality Excellence for Suppliers of Telecommunications (QuEST) Forum, *TL 9000, Quality Management System Requirements Handbook*, Release 5.5, 2013.
- [2] International Standards Office, *ISO 9001 Quality management systems – Requirements*. 2008, Geneva: ISO.
- [3] A. Schuster,R. Sterritt,K. Adamson, E.P. Curran, C.M. Shapcoti, “Towards a Decision Support System for Automated Testing of Complex Telecommunication Networks”, <http://dx.doi.org/10.1109/ICSMC.2000.886408>
- [4] Zehender M., Davutoglu C., *INTACT(Intelligent Test Automation and Configuration Tool) Handbook*, 2012, Austria
- [5] International Standards Office/International Electrotechnical Commission, 2004. *ISO/IEC 15504 - 1 Information Technology – Process assessment – Part 1: Concepts and vocabulary*. Geneva.
- [6] International Standards Office/International Electrotechnical Commission, 2004. *ISO/IEC 15504 - 2 Information Technology – Process assessment – Part 2: Performing an assessment (ISO/IEC 15504-2:2003+Cor. 1:2004)*. Geneva.
- [7] International Standards Office/International Electrotechnical Commission, 2004. *ISO/IEC 15504 - 3 Information Technology – Process assessment – Part 3: Guidance on performing an assessment*. Geneva.
- [8] International Standards Office/International Electrotechnical Commission, 2004. *ISO/IEC 15504 - 4 Information Technology – Process assessment – Part 4: Guidance on use for process improvement and process capability determination*. Geneva.
- [9] International Standards Office/International Electrotechnical Commission, 2006. *ISO/IEC 15504 - 5 Information Technology – Process assessment – Part 6: An exemplar Process Assessment Model*. Geneva.

AUTHORS

C. Davutoglu is with the IN-AUT Intelligent Automation Solutions GmbH, Vienna, Austria (can.davutoglu@in-aut.at).

N.M. Durakbasa is Head of the Department of Interchangeable Manufacturing and Industrial Metrology at the Institute for Production Engineering and Laser Technology, Vienna University of Technology, Vienna, Austria (durakbasa@ift.tuwien.ac.at).

G. Bas is with the Department of Interchangeable Manufacturing and Industrial Metrology, Institute for Production Engineering and Laser Technology, Vienna University of Technology, Vienna, Austria and with IN-AUT Intelligent Automation Solutions GmbH, Vienna, Austria (goekcen.bas@ift.tuwien.ac.at).

E. Guclu is with the Department of Interchangeable Manufacturing and Industrial Metrology, Institute for Production Engineering and Laser Technology, Vienna University of Technology, Vienna, Austria (e.guclu@ift.tuwien.ac.at).

Submitted 30 April 2014. Published as re-submitted by the authors 08 June 2014.