Jukka Tuominen

DEVELOPMENT OF TEST CASE LIBRARY FOR SERVICE AWARE CHARGING AND CONTROL

Faculty of Technology and Maritime Management Pori
Information Technology
2009

Satakunnan ammattikorkeakoulu
Satakunta University of Applied Sciences

DEVELOPMENT OF TEST CASE LIBRARY FOR SERVICE AWARE CHARGING AND CONTROL

Tuominen, Jukka
Satakunta University of Applied Sciences
Faculty of Technology and Maritime Management Pori
Information Technology
Sponsored by Oy L M Ericsson Ab
Instructor: M.Sc. Jari Temonen
Tutor: M.Sc. Kirsi Koskenheimo
August 2009
Supervisor: Lecturer, M.Sc. Ilkka Kulmala
UDC: 621.395
Number of Pages: 41

Keywords: GPRS, SACC, testing, library systems, telecommunications networks

---

The purpose of this thesis was to improve the thesis worker's competence on mobile phone network technology and develop a library system for documents generated during testing of Service Aware Charging and Control solution.

Testing of mobile phone networks is performed when modifications or upgrades have been implemented to networks. Documentation for testing process is made for functions and features to be tested. Test documentary creation was planned to make easier by developing a library system within this thesis. Existing and new documents can be stored to this library. By using this documentary, stored in the library, makes test document creation easier and faster.

Theory part of this thesis includes description of packet core network and description of service aware charging and control solution. It also introduces components occupied by this solution. Practical part of this thesis considers the developing process and testing of the library system.

The work was successfully finished and targets been set were reached. Library system is now usable and almost all existing test documentary is stored into the library.

DEVELOPMENT OF TEST CASE LIBRARY FOR SERVICE AWARE CHARGING AND CONTROL

Tuominen, Jukka
Satakunnan ammattikorkeakoulu
Tekniikka ja merenkulku Pori
Tietotekniikan koulutusohjelma
Yritys: Oy L M Ericsson Ab
Ohjaaja: DI Jari Temonen
Tutor: DI Kirsi Koskenheimo
Elokuu 2009
Valvoja: Lehtori, DI Ilkka Kulmala
UDK: 621.395
Sivumäärä: 41

Asiasanat: GPRS, SACC, testaus, kirjastojärjestelmät, tietoliikenneverkot

---

Tämän opinnäytetyön aiheena oli syventää opinnäytetyön tekijän osaamista matkapuhelinverkkoteknologioihin liittyen, sekä kehittää Service Aware Charging and Control –järjestelmän testauksessa syntyviä dokumentteja varten kirjastojärjestelmä annettujen vaatimusten mukaan.

Matkapuhelinverkkojen testausta suoritetaan aina, kun muutoksia tai päivityksiä verkkoihin tehdään. Näitä testauksia varten laaditaan dokumentit testattavista asioista ja toiminnoista. Tämän lopputyön puitteissa pyrittiin helpottamaan dokumenttien luomista kehittämällä kirjastojärjestelmä, johon jo olemassa olevat sekä uudet testidokumentit tallennetaan. Näitä valmiita testidokumentteja hyväksi käyttäen uusien testidokumenttien luominen helpottuu ja nopeutuu.

Tämän opinnäytetyön teoriaosuus kattaa kuvauksen pakettipohjaisesta runko-dataverkosta ja kuvauksen palvelutietoisen laskutus- ja hallintajärjestelmän toiminnasta sekä siihen liittyvistä komponenteista. Käytännön työn sisältäneessä osuudessa käsitellään itse kirjastojärjestelmän luomisprosessi ja sen testaus.

Työ onnistui odotusten mukaan ja sille asetetut tavoitteet saavutettiin. Kirjastojärjestelmä on saatu toimivaksi tietokannaksi ja siihen on syötetty lähes kaikki tällä hetkellä olemassa olevat testitapaukset.

FOREWORD

The work was done for Oy L M Ericsson Ab in Finland. Ericsson offers mobile communication technology solutions worldwide, and is best known for its strength in developing solutions for mobile communication.

This thesis was written in English for the following reasons; 1), Ericsson is world-wide company and the thesis must be able to read by anyone. 2), Terminology has, at least for most of parts, never been translated in to Finnish. Every part of material and books were written in English.

The work done for this thesis looks already very promising, but the exact feedback of the usefulness of library system can not be gathered until the library have been used for a longer time. Authors own competence in networking area has grown, making the good base for future working tasks. I would like to thank my line manager Jari Temonen and tutor Kirsi Koskenheimo for the support and recommendations given regarding my thesis. I would also like to thank all other people for assisting me during this entire thesis project.

Kirkkonummi, 21th August 2009

Jukka Tuominen

ABBREVIATIONS

3GPP	Third Generation Partnership Project
APN	Access Point Name
CDR	Charging Data Record
CS	Circuit Switched
CSR	Customer Service Request
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GSM	Global System for Mobile Telecommunications
IMS	IP Multimedia Subsystem
IOT	Inter Operability Testing
IP	Internet Protocol
LAN	Local Area Network
MMS	Multimedia Messaging Service
PC	Personal Computer
MS	Mobile Station
PCC	Policy and Charging Control
PCRF	Policy Control and Rating Function
PDN	Packet Data Network
PDP	Packet Data Protocol
PISC	Packet Inspection and Service Classification
PS	Packet Switched
PSTN	Public Switched Telephone Network
RADIUS	Remote Authentication Dial-In User Service
RG	Rating Group

RS	Requirement Specification
RTF	Rich Text Format
SACC	Service Aware Charging and Control
SAPC	Service Aware Policy Controller
SASN	Service-Aware Support Node
SGSN	Serving GPRS Support Node
SQL	Structured Query Language
SI	Service Identifier
STP	System Test Plant
TE	Terminal Equipment
TD	Test Description
TOL	Test Object List
TR	Test Record
TR	Test Report
TS	Test Specification
USB	Universal Serial Bus
VB	Visual Basic
VBA	Visual Basic for Applications
WCDMA	Wideband Code Division Multiple Access

CONTENTS

1 INTRODUCTION

During the past few years cellular networks have improved massively in terms of reliability, coverage, and efficiency. Fast development of smart phones has enabled the use of the wireless high speed data services such as Internet, Mobile TV and Multimedia Messaging Service (MMS). This kind of development has brought out need of alternative charging methods for counting data usage, and controlling data flows depending on service and customer rating.

The purpose of this thesis was to develop a library system for documents used in testing of Service Aware Charging and Control (SACC) Solution. Testing of new technology produces great number of documents and it was necessary to figure out a way how to store this documentary into one specific system so that test cases are easy to find and sort afterwards. Another purpose was to improve authors own knowledge on the Packet Core Network and network components within this system for future work tasks.

This thesis consists of three main sections. First section includes a short description about Packet Core Network system and more specific technical overview about Service Aware Charging and Control solution. Second section consists of a description about SACC testing and testing processes in practice. Last main section describes development and implementation processes of test case library system.

In the Summary and Conclusion chapter I have written findings about usefulness, advantages, and problems occurred by starting use the new library system.

2 PACKET CORE NETWORKS

2.1 Introduction

This Chapter is based on reference [1].

Traditional GSM (Global System for Mobile Telecommunications) is Circuit Switched (CS) system and it was originally developed for voice traffic between mobile phones and the Public Switched Telephone Network (PTSN). However, it can also transfer packet data between Mobile Stations (MS) and Packet Data Networks (PDN), for example Internet, corporate Local Area Network (LAN) or operator's service network.

Wideband Code Division Multiple Access (WCDMA) System is a Third Generation (3G) mobile system. It supports both Circuit Switched (CS) and Packet Switched (PS) communication. The speed of the transmission of data packets is higher than in GSM systems, and radio network resources are more efficiently used.

General Packet Radio Service (GPRS) provides data packet services to both GSM and WCDMA systems. It enables end-users to get fast and effective access to the PDNs, for example Internet, a corporate LAN or operator's service network through the PS network. It uses scarce radio resources efficiently, and setup and access times are much faster. Packet data function does not have an impact on circuit switched, for example traditional telephony, services which are used by GSM.

In packet data system, a message consists of large amounts of data and it is divided into several packets. The packets are reassembled to the form of the original message, after those are delivered to their addressed destination.

GPRS system provides a solution for Internet Protocol (IP) communication between MSs and PDNs. This, how the IP connection is established through mobile network, is shown basically in Figure 1.

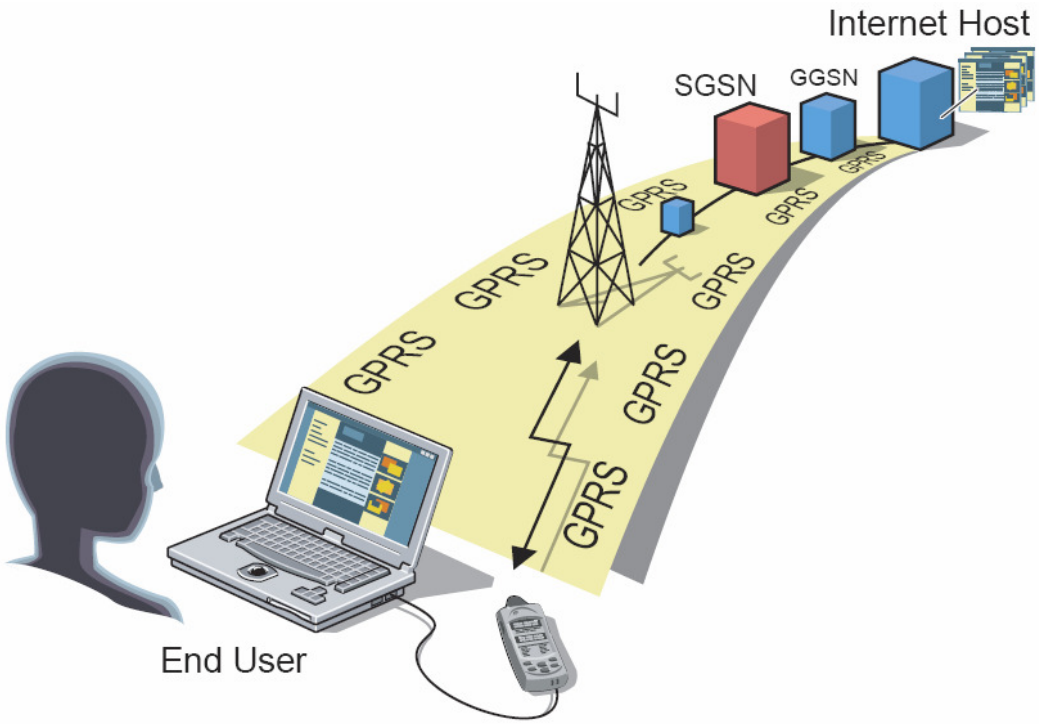


Figure 1. IP Connectivity between MSs and Internet Hosts in the GPRS Network [1]

Most important support nodes in the GPRS network are Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). The basic setup of the IP connection in the GPRS system is shown in Figure 1.

SGSN handles communication with MSs via base stations and establishment of the connection between MSs and the Packet Data Network (PDN). It forwards IP packets between all GPRS-attached MSs, within that SGSN service area, and GGSN. Connections between SGSN, MS and GGSN are handled through session management. Session management means the activation, modification, and deactivation of Packet Data Protocol (PDP) of contexts. GGSN forwards IP packets between the SGSN and

the PDN. It also collects information about data usage for charging, and handles session management.

PDP Context is a term, which indicates a logical association between MS and PDN running through a packet core network. It defines aspects, such as, routing, Quality of Service (QoS) and charging. PDP context is created when end user opens packet data connection, for example, to the Internet.

2.2 Service Aware Charging and Control (SACC)

Fast evolution of mobile phones has enabled additional functions like music players, visual radios, video cameras and satellite navigation systems, to be integrated in one user device. It has created a base for many next generation mobile services. These kinds of new services need packet data connections for different purposes and most of them need work simultaneously. Traditional fixed price per transferred byte for all packet data traffic transferred is not anymore the only charging method in the all time developing services world. Operators want to be aware of the services being used, and also to be able to control the network usage. They must also develop sensible rating system for different services, and ensure that services can easily be accessed. SACC solution presents service awareness in the packet core network. It merges the service and connectivity world.

2.2.1 Introduction

SACC is a network solution which allows flexible charging and control of service data flows at the packet core network bearer level. SACC solution consists of several network nodes and a set of optional features, such as, authorization and policy control, traffic redirection, and charging for individual services based on inspection of

packet headers and payload. It also makes possible to offer multiple services on one Access Point Name (APN), and this enables end users to use services easily without re-configuring phones depending on services used. With QoS control operators are able to manage bandwidth on separated services and user groups or individual users. QoS control enforces “fair usage” policies, in order to best utilize network resources, and granting bandwidth to specified services. SACC solution is available in different customer's configurations. For example SACC functionality can be integrated in GGSN or it can be ran in separate node called SASN (Service Aware Support Node). [2], [4], [9], [10]

SACC 2.0 uses legacy interfaces for online control, those are, Service Charging Application Protocol (SCAP), Service Rating Application Protocol (SRAP), and Subscriber Update Request Protocol (SURP). It also supports a limited number of optional features. SACC 2.0 provides service classification based on IP header inspection, application layer protocol inspection, or basic heuristic inspection of encrypted or proprietary protocols. It also provides charging based on used volume or active time, online credit control and charging of services, service authorization and policy control, and automatic traffic redirection of unauthorized services. SACC 2.0 makes possible to manage bandwidth based on used service, and it fully supports offline charging through Charging Data Records (CDR). [2], [3]

SACC 3.0 solution introduces the Third Generation Partnership Project (3GPP) standardized interfaces Gx and Gy with additions to also support optional features, such as, event-based charging. More specific description about interfaces, introduced in Figure 2, follows later in this report. SACC 3.0 has more advanced packet inspection compared to older SACC versions. It also introduces a more granular QoS control and GGSN support for Event and Content charging using network centric service awareness. Advice of Charge function allows a user to get information about the price of the certain service before accepting a charge. Full support for offline charging through CDRs is also supported. [2], [4]

In the rest of this document, the terms SACC and the SACC solution refers to SACC 3.0 release if not mentioned otherwise.

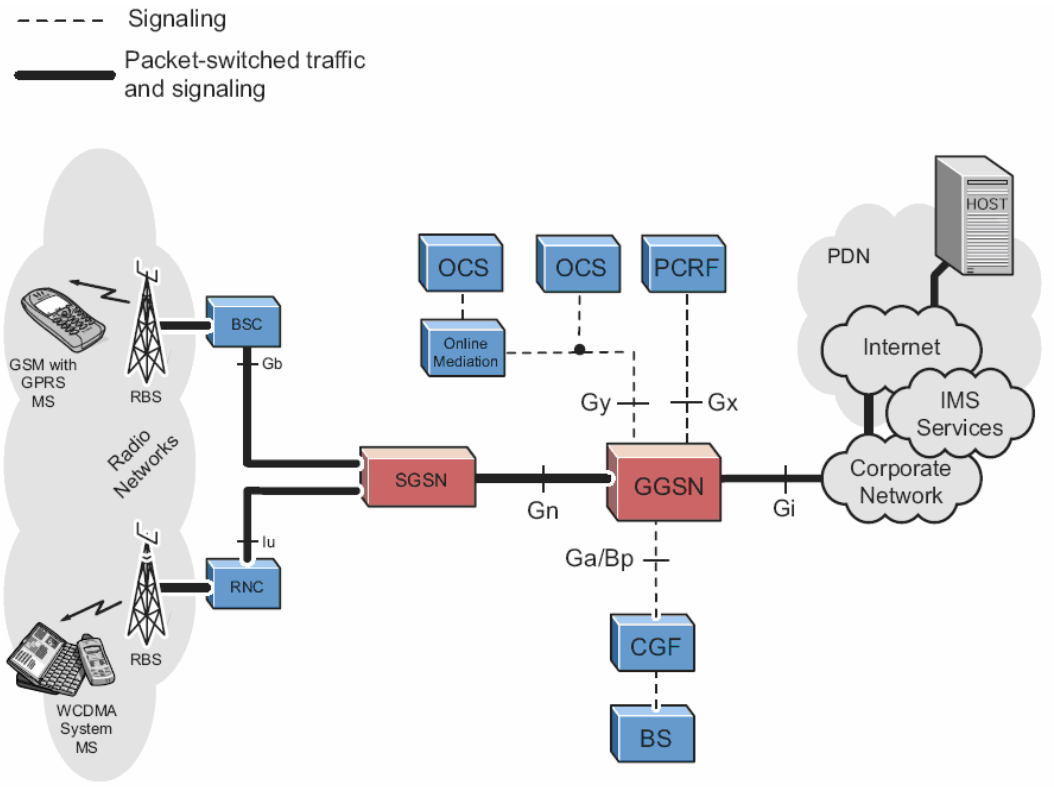


Figure 2. SACC interfaces and nodes in the packet core network in case of "GGSN integrated" SACC solution

Nodes are hardware units, where different services and functions are performed. Nodes involved in GGSN integrated SACC, and how those are interconnected in the packet core network system, are shown in Figure 2. Basic functions and services of these nodes are described in the next chapter. Standalone SACC Solution with Service-Aware Support Node (SASN) is described in chapter 2.2.3. In the rest of this document terms SACC and SACC solution refers to the GGSN integrated SACC solution if not mentioned otherwise.

2.2.2 Nodes in GGSN integrated SACC Solution

Gateway GPRS Support Node (GGSN)

GGSN is a main component of packet core network, and it is the gateway node between operator's internal network and external packet data network, such as, the Internet. GGSN's tasks are to check if the MS is GPRS-attached and forward data coming from external network to the specific MS, or if the MS is inactive, data from external network is discarded. GGSN also converts data packets coming from SGSN into the appropriate PDP format, for example IP, and forwards them to the corresponding PDN. When packets are coming from external network to internal packet core network, GGSN converts destination PDP addresses to MS addresses of the destination user. In the other hand GGSN is responsible for IP address assignment for the connected device. It also handles session management, which includes dynamic IP address allocation and QoS negotiation. Another important GGSN function is to perform charging and authentication task, and handle the PDP context's QoS management. With SACC it is possible to enable the service aware charging on the GGSN. [1], [5]

Serving GPRS Support Node (SGSN)

SGSN is responsible for the communication between MSs and establishment of connection between MS and the PDN through the GGSN. If SGSN pool functionality is not used, an SGSN serves all GSM or WCDMA system users, which are physically located in the geographical SGSN service area. SGSN (WG) can serve GSM and WCDMA users simultaneously. SGSN forwards IP packets between all GPRS-attached MSs and GGSN within current SGSN service area. Connections are handled by session management. SGSN also registers information about MS location. Subscriber data management enables differentiated services, for example, different QoS. SGSN also gathers information about data packet traffic and can send it to the charging system. [1]

Charging System (CS)

CS allows real-time credit control and charging of user traffic. This node applies service rates to usage reported by GGSN. If online charging is performed by a proprietary charging system, the connection to the CS system can be mediated by an Online Mediation (O.M.) system. The Charging System is Ericsson specific name for the 3GPP specified node Online Charging System (OCS). [2]

Online Mediation (O.M.)

Online Mediation is Ericsson's own optional mediator, which is used when the online charging is performed by a charging system which is not compliant with the 3GPP Gy interface. Online Mediation helps operators in managing charging information. [4], [6]

Service Aware Policy Controller (SAPC)

Service Aware Policy Controller, formerly known as Ericsson Policy Controller (EPC), allows GGSN to receive dynamic authorization and policy control information. SAPC is based on 3GPP Release 6 standard of Policy Control and Rating Function (PCRF), and it also includes parts of next Release 7 standardization of a Policy and Charging Control (PCC) architecture. [7]

Billing System (BS)

Billing System handles CDRs which have been generated by traffic through GGSN. [7]

File & Event Mediation (F&E)

File & Event Mediation is an optional mediator for CDR between BS and GGSN. The F&E can translate signaling, if the external node does not recognize protocol

used by GGSN. The name File & Event Mediation is Ericsson specific name for 3GPP specific Charging Gateway Function (CGF). [2], [7]

2.2.3 Service-Aware Support Node (SASN)

SASN is the stand-alone service aware product in the Ericsson SACC solution. It includes all the same functions like GGSN integrated SACC solution, but the difference is that SACC functions like Packet Inspection are done in a separate SASN node instead of GGSN node. In some cases it is useful to separate SACC functionality outside GGSN to SASN node because of flexibility or performance reasons. [11]

2.2.4 Interfaces

The nodes are connected to each other by interfaces. All interfaces are compliant with different standards and protocols. Interfaces between nodes in the SACC solution are shown in Figure 2.

Gx+

Gx+ is Ericsson specific interface, based on 3GPP Release 9 functionality for Flow Based Charging (Gx). It includes all the functions specified in 3GPP specification, but it also includes Ericsson specific protocol extensions. This interface is connecting GGSN to SAPC. [7]

Gy+

Gy+ interface is an Ericsson specific interface, which has all the functions of the 3GPP Gy interface which is a subset of the 3GPP Ro interface. It also has Ericsson specific extensions for standard Gy interface. This interface connects the GGSN and

CS. The connection to the CS may be also mediated by an Online Mediation system. [7]

Ga

Ga interface, also called Bp, connects GGSN to Billing System. This connection may also be mediated through a CGF. This interface supports standardized 3GPP Release 6 Gz reference point. [7]

2.2.5 Charging and Control Methods

SACC provides three main charging methods. Those are charging by volume, time and event. Both online and offline charging are supported. Content charging is based on traffic inspection and classification and it's a part of all above-mentioned method.

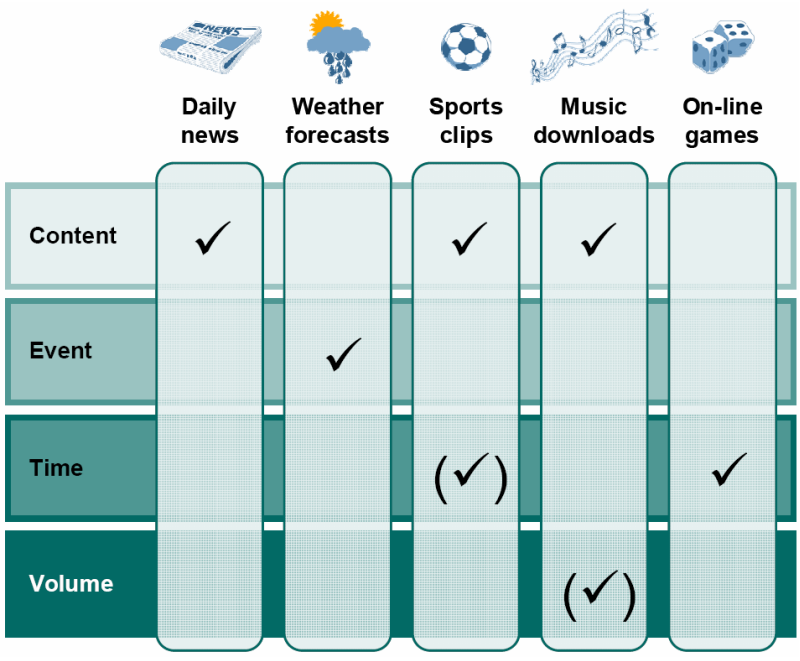


Figure 3. Flexible Charging on the same bearer [12]

Volume based charging measures only the amount of transferred bytes. Traffic can be counted by calculating all forwarded bytes or only by payload of the application used. Time based charging measures only used time, and there are four different methods available on how to measure the active time. Event based charging charges only when event, for example file download, has completed successfully. Example of the flexible charging is shown in the Figure 3. GGSN tracks protocol events and these are used as the basis for charging. Commonly, the volume based charging is used by default. Time- and event-based methods are optional. [7]

Service-Aware Control allows operator to control service delivery and ensures right service level on a bearer. It implements advanced network control functionality for Quality of Service (QoS). This is needed to ensure a good user services that utilizes high or varying bandwidth. QoS control makes it possible to differentiate services and manage bandwidths of many services in one bearer (Figure 4). It enables operators to control and achieve a predictable service delivery to ensure optimal network utilization and grant enough bandwidth for a certain services.

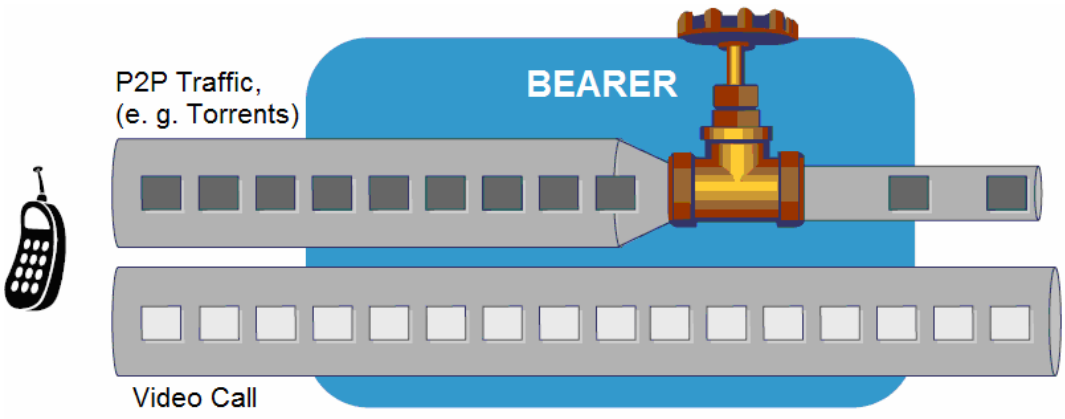


Figure 4. QoS Control

2.2.6 Packet Inspection and Service Classification (PISC)

All the functionality in SACC solution is based on Packet Inspection and Service Classification function in the GGSN. PISC identifies different types of traffic with the Service Identifiers (SIs). There are two levels of packet inspection offered by GGSN (shown in Figure 5); header inspection and deep packet inspection. [14]

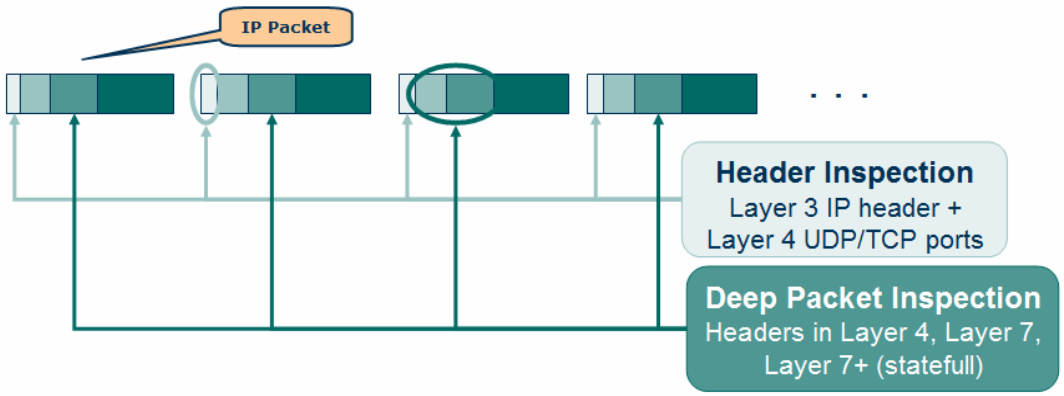


Figure 5. Packet Inspection

The purpose of PISC is to support authorization and charging of services delivered to the user. It makes it possible to filter out different types of traffic and to identify the volume, time and event for certain services. Every packet which arrives to PISC enabled GGSN, is inspected and classification is made in to SIs and Rating Groups (RG) (Figure 6). RG is a configurable collection of SIs. [4]

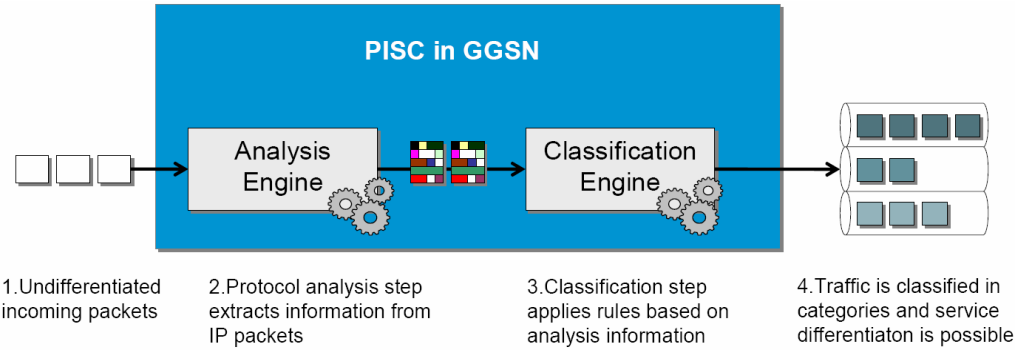


Figure 6. Packet Inspection with packet analysis and classification [8]

2.2.7 Online Charging

This Chapter is based on reference [8]

When a user starts to use packet data services, for example opens the web page, Service session is being created. GGSN classifies the packets and opens a credit session with online charging system. Quota is being requested for the service, and if the quota is granted, the user is allowed to use the service. If the user has enough credits, the quota is granted for the service, and if not, service may be blocked or subscriber may be redirected to other address, for example to info page which tells that user is running out of credits.

SACC uses multi bucket system which means that each rating group will have its own credit bucket (see Figure 7). Basically it means that the user can use services with different tariffs and charging models simultaneously.

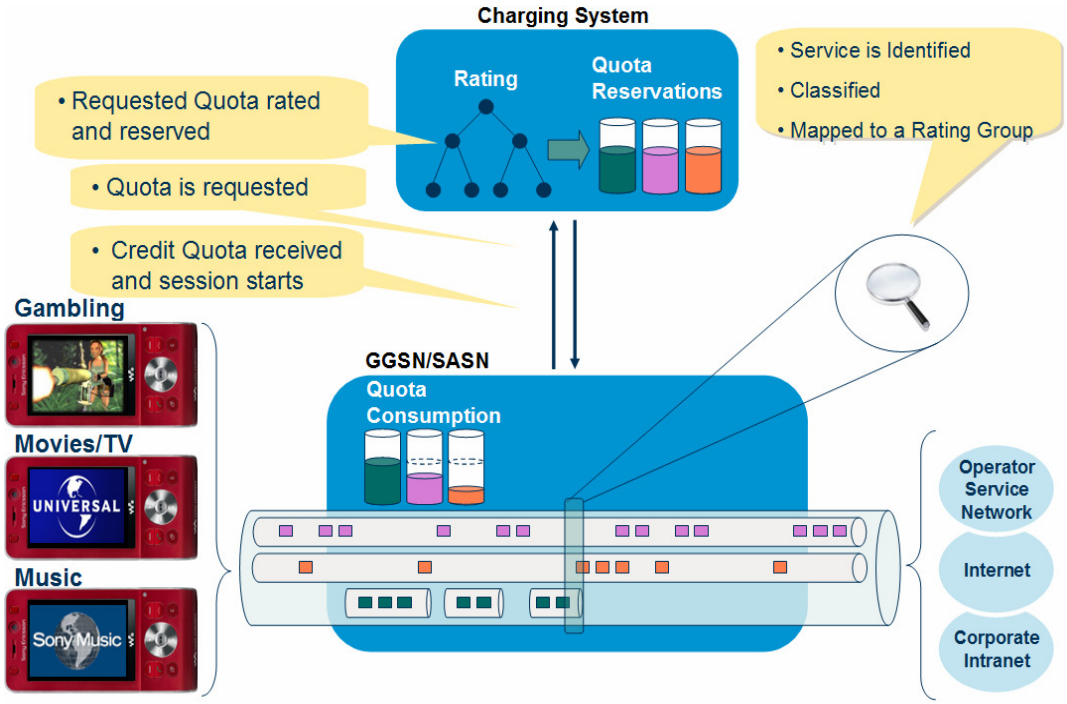


Figure 7. Online Charging [13]

GGSN does not make any rating, in the other words, it does not handle any money. It only reports detailed service usage and handles the data flow according to policies and received quotas.

While using services, more quotas is requested and received automatically. Depending on configuration or used service, new quota can be requested when old has been consumed, or there is threshold in quota when the new quota is requested. After the user has stopped the use of service, remaining quota is returned to users account.

It is also possible to change tariff in the middle of the service session. For example the switch can happen before or after office hours, when the tariff is higher. SACC 3.0 will split the reporting in usage before and after the switch, so accurate rating is possible and signaling storms are avoided.

2.2.8 Authorization and Policy Control

Authorization and Policy Control consists of three optional features, which enable authorization for IP packets in PDP context. These features are Gx+ Static Access Control, Gx+ Dynamic Services Control, and Remote Authentication Dial-In User Service (RADIUS) Access Control. GGSN receives charging rules and authorization policies over Gx+ interface and handles traffic based on this information. GGSN can outsource authorization decisions to PCRF or RADIUS. [8]

2.2.9 Benefits

SACC functionality is useful for the operators to present and enforce an understandable price model to end users. It includes deep packet inspection which improves operator's existing infrastructure with service and subscriber awareness. This makes it

possible to charge and control different subscribers individually. With QoS control, operator can improve service quality by controlling the usage of the network. SACC provides opportunity to control all the data traffic in network. It also simplifies charging of 3rd party value add services. No additional logins by users is needed when using service authorization and access control. In other words, user identification happens automatically by inspecting traffic. [8], [9]

From end user's point of view SACC offers high technology services through one portal, and secure payment system through operator account. There is no need to send other information, for example credit card info, while using premium services. Overall SACC offers easy access and usage off services, and clear pricing and payment systems. [9]

SACC also allows the operator to obtain valuable information about users' behavior and statistics about the spread of used services by analyzing traffic. This kind of information is very valuable when operator, for example, is creating new services or charging models, and measuring the capacity of network. Based on this information, operator can, for example set lower QoS for different services in peak hours to ensure network's functionality.

3 TESTING OF SACC SOLUTION

There is no existing literature or documents commonly available about SACC solution testing process, therefore all the information I have for this report, is gathered by interviewing my manager Jari Temonen, tutor Kirsi Koskenheimo, solution architect Juha Palenius, and other people involved in SACC testing process.

There are lots of different kind of testing performed while developing and generating new software and hardware, but I will mainly concentrate on testing, what happens

after the solution is sold to the customer, because that phase of testing is performed by Ericsson Service Delivery Organization.

3.1 General information about testing

The general purpose of testing is to find out if the solution delivered is working properly and if it includes all the functions which customer is requiring. Testing also works as verification for the Customer. For Ericsson testing is a proof, that solution did work properly when it was delivered. Sometimes test cases are used for Customer Demonstration purposes, when it is needed to show some features or deliver a trial system for customer.

Inter Operability Testing (IOT) is a part of developing a new product. Tests are made to ensure that Ericsson products are working with other vendor products, which makes it possible to sell Ericsson products to networks, where other vendor products are used. IOT testing was performed here in Ericsson Finland few years ago, but nowadays it is performed in Germany.

Test process starts with making Requirement Specification (RS) and the Solution Description documents, which describe the needs and requirements of configuration. The configuration and Test Case List are made based on these documents. Test cases to perform are chosen, too. Typically, the same person makes the configuration and the test case documents mentioned above. This test documentation has to be approved by the Customer before the configuration can be installed on the Customer's system and testing process can be started. The Customer is involved in the testing as early as possible, because the time is used then more efficiently and the solution can be mobilized earlier.

3.1.1 Ericsson Internal Testing

This phase of testing is performed in the Ericsson's System Test Plant (STP). The new configuration is tested on Ericsson's own equipment before it is tested in the Customer's system. Simulators are used to generate traffic for testing purposes. For example when testing GGSN configuration in STP, the simulator called Outline simulates signaling between GGSN and charging nodes and the Trembler simulates traffic between SGSN and GGSN. Data flow is being monitored with protocol analyzer like Wireshark or Ethereal.

The purpose of this phase is to reach a level where configuration is working substantially. In addition, this phase helps a person who has generated the configuration to learn how it works in the practice. Normally the same person is generating the configuration and performing testing of the configuration.

3.1.2 Integration Testing

Integration testing is performed with the Customer's equipment in the Customer's network. New configuration for the solution is loaded, and Ericsson experts perform their own tests to make sure that the configuration is working the same way as it worked in the STP. In this phase the customer does not intervene which tests is performed. Testing is performed in co-ordination with experts of all nodes used in the solution. Optimal situation for efficient and fast testing is that all experts of certain nodes are physically located in the same site at the same time, but at all times this is not possible. If all the persons needed are not able to come in testing place, the remote connections are used.

Integration testing is performed to ensure the solution is working like it has been designed to work and the configuration is compatible also with Customer's hardware and network. Integration testing is a prerequisite for acceptance testing.

3.1.3 Acceptance testing

After integration tests are finished, the Customer is taken into the testing process. Every one of the functions ordered by the Customer are tested and demonstrated to the Customer. Acceptance testing is actually end-to-end testing, which means that this testing is performed with real Terminal Equipment (TE), for example mobile phone, or computer with a GPRS card. Tests can also be executed in the real working mobile network. Some of the Customers might have a special network for testing purposes, but most of the tests are still run in the live network. If testing is performed in the live network, certain test APN is usually generated for the testing purposes. Normal end-users, operators' customers, are not allowed to access these test APNs.

Number of test cases is tried to be kept slightly small to speed up the mobilization of new solution. That's why only the ordered functions are tested. After a test case is successfully executed, the Customer signs its acceptance for that specific test case. All the test cases chosen to Test Object List (TOL) are tested at this phase.

If an error is found during the testing phase, it can be checked from Ericsson knowledge database whether corresponding problem has already been found and fixed in a later software version. If the problem is already fixed in the new software version, the software is updated to that version and testing will continue with the new software version. If an error is found from Ericsson knowledge database, but it is not fixed yet, it is possible to refer to the already found problem in the Test Report (TR). When the fix date and/or software are known, it will be mentioned in the TR as well. In case that the error has never been found before, the Trouble Report is written and the continuation of testing is being consulted with the Customer. The continuation of testing

process depends of the type of the error; does it prevent the using of the whole solution or is it only a minor function, so that use of other functions could be possible.

3.1.4 User Acceptance Testing

There is also a possibility that Customer wants to perform some Customer specific test cases. This normally happens after the acceptance testing is done. The Customer might get assistance from Ericsson for doing this testing.

3.2 Different Test Case categories

Test cases can be categorized by Interface or Feature and Standard. By Interface or Feature means that test cases can be grouped depending on which interface (e.g. Gy or Gx) or SACC feature (e.g. PISC or CDR) they are bound to. Also test cases can be valid only for specific standard (e.g. 3GPP, Ericsson, Vodafone) or it can be used commonly. Some of test cases, for example PISC test cases, do not depend of any standard.

3.3 Documents

Testing process produces a great number of documents and the main idea for this thesis was to create a library system for these documents. From this library, the information can easily be found and re-used in the further projects. Typical documents, generated during testing process, are introduced below.

Test Object List (TOL)

Test Object List is a general list of test cases being performed. It includes at least a name, interface or feature and a very basic description of the subject being tested in the specific test case. It can include information about success of performed test cases, as well.

Test Specification (TS) / Test Description (TD)

This document includes all the same information of test cases as TOL, but there is also more detailed description of test process, such as actions done in the test process and results which the test should return.

Test Report (TR) / Test Record (TR)

Test Report or Test Record contains more detailed information, like malfunctions or faults found. Other noticeable information about the performed test case can be mentioned in this document as well. It can also contain information about written Trouble Tickets, if there were malfunctions in the tested configuration. Testing progress information, for example planned schedule, is sometimes written in this document. This document is not mandatory to write, because the information of test success can also be written in TS or TOL, if no more detailed information of test case success is needed.

4 TEST CASE LIBRARY

The purpose of this Test Case Library is to decrease the amount of time and work, which might be used to search through multiple hundred pages documents to find a correct test case. The optional situation is that all existing and new test cases are stored into the library system.

4.1 Requirement specification

The base for requirement specification was created in weekly thesis follow-up meetings with my line manager Jari Temonen and tutor Kirsi Koskenheimo. The specification became clearer during the development and programming phase of the library system. In the weekly meetings there was discussion about features and functions of the library system and how those should work. Some ideas regarding the library were gathered in discussions with engineers in the coffee corner. The final requirement specification developed to following:

This library should categorize different types of test cases and represents them in a comprehensible form. The library should make possible to sort and search test cases easily from large amount of data, select the test cases needed, and export those to external program, for example Microsoft Word or Microsoft Excel. Also the storing and modification of data (test cases) should be done quite easily. This library system must be able to run in every PC (Microsoft Windows workstation) without installing any additional software.

4.2 Platform selection

Originally the library system was planned to run on Microsoft Excel, but the Excels capability to handle large amounts of data and limited features to sort and export data forced to plan alternative platforms for library system.

For platform was chosen Microsoft Access instead of Microsoft Excel, because of Access advanced database functionality, and cooperation with other Microsoft Office products. Also important reason to select Microsoft Access was that the library system should be able to run on every computer without installing any additional software. It was logical to build database on Microsoft Access databases and the func-

tionality in the library program was written with Microsoft Visual Basic for Applications (VBA), which is an integrated part of Microsoft Office. The Graphical User Interface (GUI), as well, was possible to make with Microsoft Access.

The programs Microsoft Access, Microsoft Excel and Microsoft Word, used in this thesis, are parts of Microsoft Office suite, version 2003 with Service Pack 3. The Microsoft Visual Basic for Applications version was 6.3.

4.2.1 Taking constraints into consideration on Access 2003

Tables of this chapter are based on reference [15].

Microsoft Access 2003 provides a very useful platform for making a large (Table 1) multifunctional database, but still some of limitations must be taken into account while designing functionality and features for this library system.

Attribute	Limit
Size of complete database	2 Gigabytes
Number of objects in a database	32 768
Modules	1 000

Table 1. Microsoft Access Database - General

Most noticeable thing is the limit of characters in a table record (Table 2). Test cases contains long descriptions and normal table record can only have 4 000 characters. This limitation does not relate characters in a Memo field, so that's why memo fields have been used for test case descriptions. Otherwise all the database functions can't be used for memo fields, so identification information (like Standard, Feature information or ID number) of test cases must be written to normal table fields.

Attribute	Limit
Number of fields in a table	255
Size of table	2 Gigabyte
Number of characters in a record (excluding Memo fields)	4 000
Number of characters in a Memo Field	65 535 when entering data through the user interface and 1 gigabyte of character storage when entering data programmatically
Number of characters in a field property setting	255

Table 2. Microsoft Access Table

For the design of sorting and selection -functions in the test library, there were few affecting limits (Table 3). Because there will be hundreds of test cases with different properties, making these functions by using queries may result with too large database handling clauses. That's why the list of selected TCs is created by writing the test case identification information to a temporary table "ApuTable" and using a query for reading TC data from TC database. This temporary table also keeps TCs in right order.

Adding this information to table, instead of query, causes another problem; the ID-numbers in "AutoNumber" field are growing. The "ApuTable" needs to be removed completely after using it and created again with necessary fields before using it again. These temporary tables and queries are being cleared every time before starting new report creating.

Attribute	Limit
Number of enforced relationships	32 per table minus the number of indexes that are on the table for fields or combinations of fields that are not involved in relationships
Number of tables in a query	32
Sort limit	255 characters in one or more fields
Number of characters for a parameter in a parameter query	255
Number of ANDs in a WHERE clause	99
Number of characters in an SQL statement	Approximately 64 000

Table 3. Microsoft Access Query

The limitations of report page width and height (Table 4) causes some unwanted page breaks and line feeds but this problem is partly solved by setting row width to maximum. Some errors are still possible, but those can easily be corrected manually in a word processing program, while adjusting document to fit the document base used. More information about this manual correction can be found from User's Guide – document, which is attached at the end of this thesis. Inputting data to database happens by using text boxes, but individual parts of TCs do not reach the 65 536 characters limit.

Attribute	Limit
Number of characters in a text box	65 535
Form or report width	5 587 mm
Section height	5 587 mm
Number of printed pages in a report	65 536

Table 4. Microsoft Access Forms and Reports

4.2.2 Visual Basic in Access 2003

The functions of Visual Basic for Applications (VBA) are slightly limited of full version of a Visual Basic (VB). On the other hand, VB has some extra features that VBA doesn't include. However VBA is very usable and powerful tool for creating the functionality to the library system. Also VB and VBA are very commonly used programming languages, and help tips for composing with VBA can easily be found on Internet; for example forums and Visual Basic Developer Center (http://msdn.microsoft.com/en-us/vbasic/).

VBA should not be confused with the VB or Visual Basic.NET (VB.NET) programming products. Their syntax is very similar, but they have also some differences. Access 2003 VBA is typically used in building small and simple Access database applications designed for a few users. VB and VB.NET are typically used to build more complex and enterprise level applications which uses external database. VBA uses a built-in database (Access database). [16]

4.3 Functions and User Interface in Test Case Library

The Graphical User Interface is designed as simple as possible, big buttons in logical order and no unnecessary buttons. Colors of forms are set similar like Microsoft Office 2003 basic colors, see Figure 8. The captions of buttons are designed in cooperation with Kirsi Koskenheimo.

Different functions of the library program can be executed from main menu, which opens when the library program is opened. Here is a description about functions of this library program, but more detailed description about using the test case library can be found from Test Case Library's User Guide, which is as attachment at end of this report.

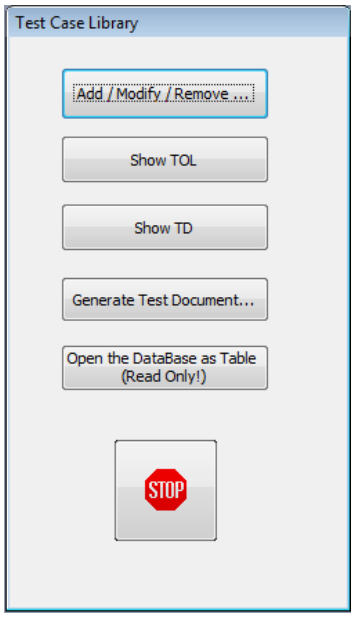


Figure 8. Main Menu of test case library

Test Cases in the library are categorized in the same way as in Chapter 3.2 (Different Test Case Categories). This categorizing is used to select test cases from library with selection form (Figure 9). The library is flexible, which means that users are allowed to add or remove interfaces. This information is stored to own tables in the database. Adding or deleting can be done from Add / Modify / Remove –form, or by modifying the table where the Interface or Standard information is stored.

When list of test cases is opened with Generate Test Document –button, it creates an SQL query, where the information from list boxes appears as sorting attributes. Because the data in the selection list is generated by query, all the data appears in read only mode. All kind of modifications for the test case data must be done from Add / Modify / Remove –form (Figure 11).

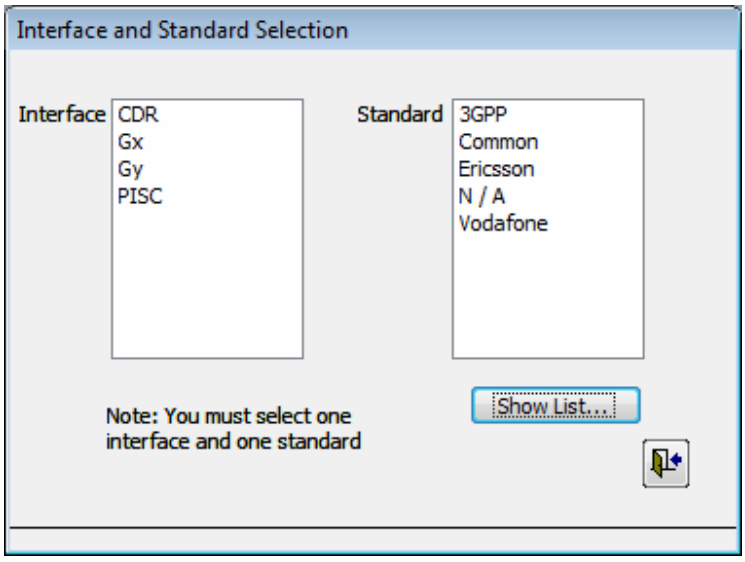


Figure 9. Interface and Standard Selection

In the Test Case Selection –form (Figure 10) the user can pick up one or more test cases from the list. The selection actually writes the selected TCs' ID number to temporary table, called ApuTable and the query gets the TC information from the Test-Cases table.

After the TCs are selected, it is possible to export selected test cases to Excel list for the Test Object List document, or to Rich Text Format (RTF) file for the Test Description. Format of exported documents is simple, the data of documents can easily be copied into template of final document. Possibility for creating Test Report – document does not exist in library program, because this document can easily be created manually using Test Object List. Detailed information about creating these documents with this library system can be found from Users Guide (attachment).

User must select TCs in same order that he/she wants to place them in the TOL and TS documents. Selected TCs can't be re-arranged, but those can be removed one by one. Preview functions were added to the TC selection to make easier to find correct TC. The query, which is used to export documents, is set to order TCs in the same order like them are selected from selection list.

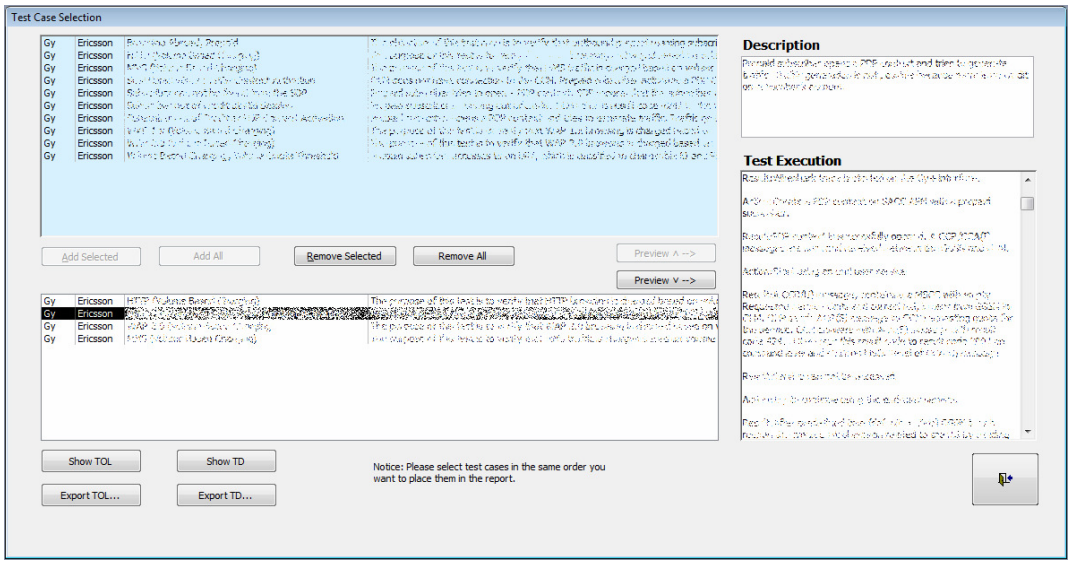


Figure 10. Test Case Selection

All the data manipulation in the library must be done from Add / Modify / Remove – form (Figure 11). This form writes data directly to database to table TestCases. User can browse Test Cases in the library by using the browsing buttons on bottom of this form. Also with a mouse scroll it is possible to browse TCs.

TCs can be modified, deleted or new TC can be added to library. ID-numbering helps to identify TCs and the same ID numbering can be found behind Main Menus' Show TOL –button.

Interface/Feature adding and deletion is also done from this form. This information, like Standard, has also been written to own tables in the database. Standard modification can't be done from this library programs interface, thus it must be done directly to its own table in the database.

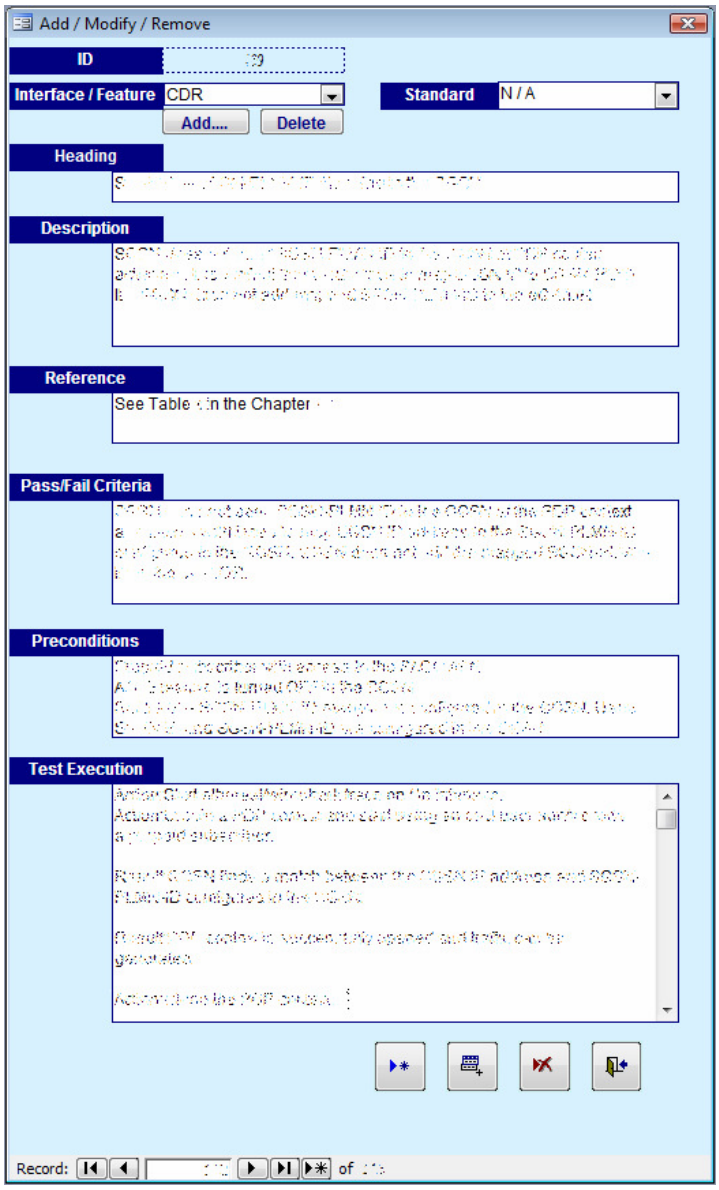


Figure 11. Add / Modify / Remove

4.4 Testing

During the development of library system there has not been a certain testing phase, but testing has been made at every turn during the process. Each function was tested before creating new one. Error correction was made immediately after an error was

found. In weekly thesis follow-up meetings the library program and new features were demonstrated and proposals for improvement were received.

Also other people have helped in testing. My tutor Kirsi Koskenheimo has used the library in her work tasks and she has reported errors and comments about usability of program. My workmate Jafar Aman has also been helping in this project by inputting TC data into the library.

5 SUMMARY AND CONCLUSION

This thesis contains 2 main sections: theoretical part and practical part.

In the theoretical part there is a short description of packet core networks, which helps understanding about description of Service Aware Charging and Control. Main functions of SACC are introduced and some speculation of its benefits is written. This thesis also contains review about testing of SACC solution, different testing phases and documentation generated for testing. In the practical part the library system for SACC test documentation has been created. This thesis describes the developing process of this library.

This library makes test creation of documentation much easier, and it reduces time used with creating test documents. It seems to be a very useful tool. Main functions of library are possibility to add, modify and remove data, and to sort and choose objects, which will be exported out from library. Exported data can be copied to document templates.

In future, developing of this library may continue, if some kind of issues appears after longer time of use. Some features may be needed to add this library or existing fea-

tures may need some updates. Overall this library will need slight maintenance, like updating Standard or Interface information. This can be done by any user.

During this thesis I have gathered a lot of information about mobile network solution testing process and improved my own competence on mobile network technology, which will be usable in my future work tasks. My English language skills are also improved during this thesis process.

REFERENCES

[1] GPRS Overview (67/1551-AXB 250 05 Uen AB), Ericsson AB, 2008

[2] SACC Overview (16/221 02-AXB 250 10/3 Uen J), Ericsson AB, 2008

[3] SACC 2.0 Overview (23/221 02-AXB 250 10/3 Uen F), Ericsson AB, 2008

[4] GSN WORKSHOP – SACC, Slideshow, Olli Salonen, Ericsson AB, March 2009

[5] Wikipedia – GPRS Core Network
http://en.wikipedia.org/wiki/GPRS_core_network, (visited: April 09, 2009)

[6] Multi Mediation 5.0 – Online mediation (1/1551-FAM 901 344 Uen B), Ericsson AB 2008

[7] SACC 3.0 Overview (16/221 02-AXB 250 10/3 Uen A4), Ericsson AB, 2008

[8] SACC 3.0 Solution Overview (1/221 05-FGD 101 30/V3 Uen Rev C), Ericsson AB, 2006

[9] SACC 3.0 – Generic presentation, Slideshow, Ericsson AB, April 2006

[10] Business Guide – Service-Aware Charging and Control, Ericsson AB, April 2008

[11] SASN R5 Technical Product Description Overview (221 02-AXB 250 60/2 Uen), Ericsson AB 2006.

[12] Executive Summary – Service Aware Charging and Control (4/221 01-FGB 101 268 Rev A), Ericsson AB 2006
[13] SACC Training material, Online Charging for SACC, Slideshow, Peter Lindau, Ericsson AB, October 2008

[14] Packet Inspection and Service Classification – Technical Product Description (21/221 02-AXB 250 60 Uen Rev A), Ericsson AB, 2008

[15] Microsoft Access Specifications and Limits | Database Solutions for Microsoft Access | databasedev.co.uk http://www.databasedev.co.uk/access_specifications.html, (visited: July 21, 2009)

[16] D. Gosnell, Beginning Access 2003 VBA – Google books, p 1 http://books.google.fi/books?id=F92Y9Lg_sfcC&pg=PA1&lpg=PA1&dq=visual+basic+access+2003+general&source=bl&ots=iZom3ZD6Oa&sig=HI_E4Um4aKzMz50zR1FFg717wZA&hl=fi&ei=dbplSubXN4fJ-QbMsKFm&sa=X&oi=book_result&ct=result&resnum=2 (visited: July 21, 2009)

Test Case Library

USER'S GUIDE

Jukka Tuominen

August, 2009

CONTENTS

1 INTRODUCTION

This User's Guide is written for Test Case Library (TCL) end users and includes useful information of using library program functions and generating documents with using TCL. Information about development process and more technical information about library can be found in the thesis DEVELOPMENT OF TEST CASE LIBRARY FOR SERVICE AWARE CHARGING AND CONTROL.

2 TEST CASE LIBRARY PROGRAM

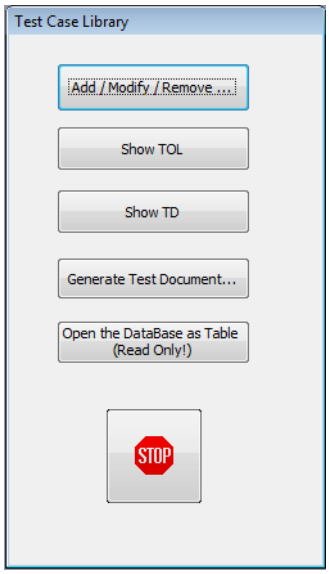
The library program is a Microsoft Access database, which contains Test Case (TC) data. Library can be found from Eridoc, and the folder also includes User's Guide (this document) and templates for Test Object List (TOL), Test Record (TR) and Test Specification (TS) documents.

Contents of folder:

- TCL.mdb - Library Program / Test Case Database
- TCL – User's Guide.pdf - User's Guide
- TOL_template.doc - Template of TOL –document
- TR_template.doc - Template of TR –document
- TS_template.doc - Template of TS –document

2.1 Using the program and Main Menu

Download the library to your computer and start it by double clicking the file TCL.mdb. Microsoft Access opens and following screen should appear.



2.1.1 Add / Modify / Remove

This opens Add / Modify / Remove –form, where new Test Cases can be added into the database. Existing TCs can also be modified or removed here. It is also possible to add or delete interfaces from this form. Please be sure, that any text to be filled into text boxes, DOES NOT contain ANY kind of formatting (for example if text is copied from Microsoft Word).

Functions of buttons in Add / Modify / Remove –form:


Add a new empty Test Case.


Remove current Test Case.


Copy current Test Case to a new Test Case.


Exit form.

2.1.2 Show TOL / Show TD

These buttons open Test Object List or Test Description which includes all TCs inserted in the library. TOL list has also ID number of TCs, which helps to identify them. It is useful if it is necessary to find a correct TC at Add / Modify / Remove form.

2.1.3 Generate Test Document

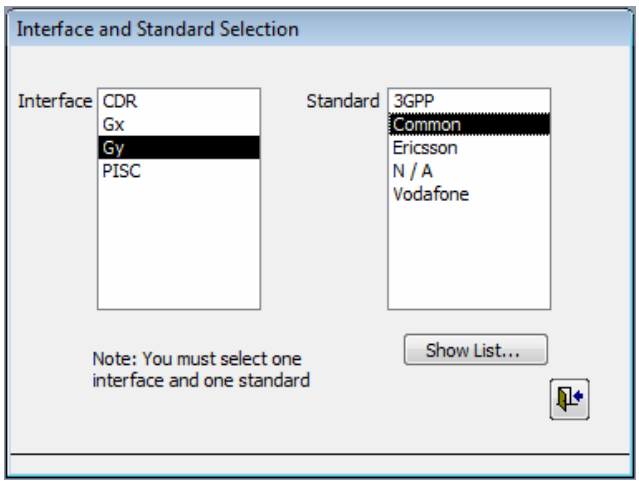
This button starts creation of TOL and TD documents, which can be exported out from library system.

3 CREATING DOCUMENTS WITH USING LIBRARY

This chapter guides you to create and export TD and TOL documents by step by step.

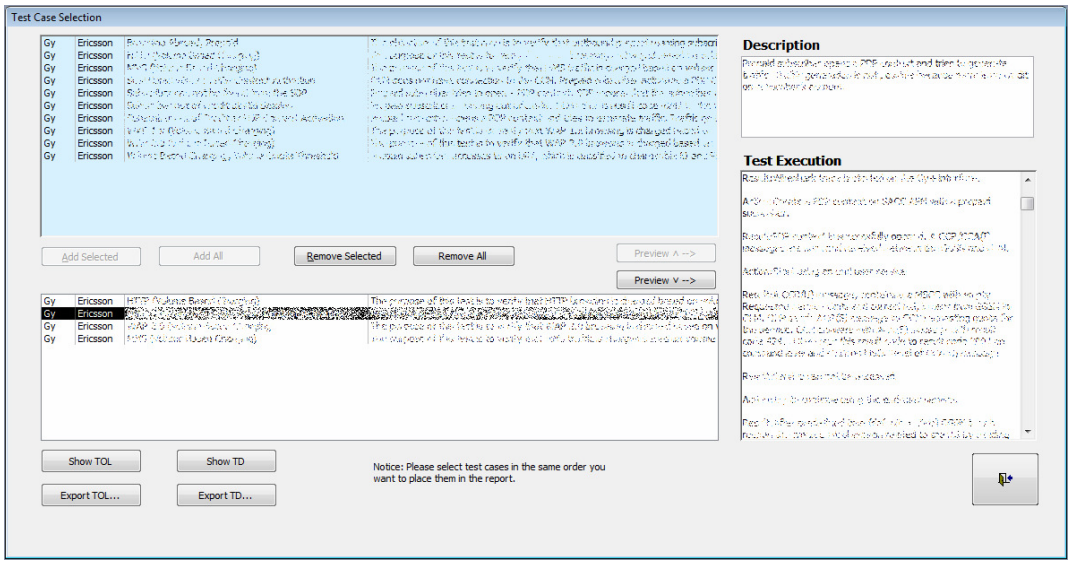
STEP 1: Start the library program and click Generate Test Document… -button.

STEP 2: Interface and Standard Selection



User needs to select one Interface and one Standard. Please note that CDR and PISC are not interfaces, so you must choose N / A – Not Available from the Standard list.

STEP 3: Test Case Selection



User selects necessary Test Cases by choosing it from upper list and clicking Add Selected -button or just double clicking item. User can also Preview selected TC; upper Preview –button works with upper (selection list) and lower with lower (selected list). All test cases can be added by pressing Add All –button, and selection list can be cleared by Remove All –button. NOTE! THE ORDER OF ITEMS IN THE LIST OF SELECTED ITEMS CAN NOT BE CHANGED!!

Show TOL –button shows Test Object List of selected TCs.
Show TD –button shows preview of Test Description document.

STEP 4: Exporting data
Data can be exported by pressing Export TOL or Export TD –button.

Export TOL: Creates Microsoft Excel (.xls) –file of selected items.
Export TD: Creates Microsoft Word (.rtf) –file of selected items.

Exported files do not contain much of formatting; user can format documents as needed. Next chapter contains an example of fitting exported data to templates.

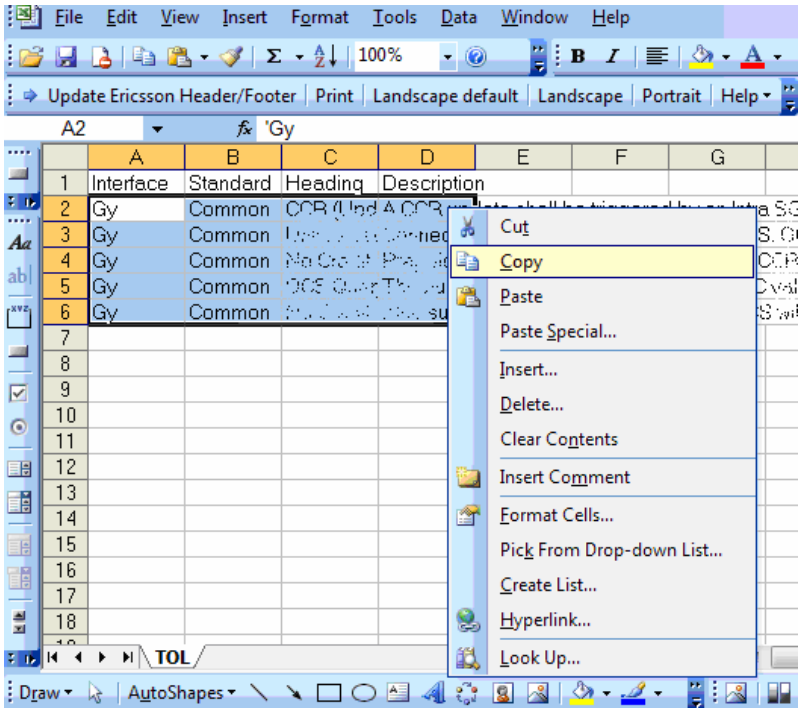
4 FITTING EXPORTED DATA TO TEMPLATES

4.1 Test Object List (TOL)

STEP 1: Download and open file TOL_template.doc

STEP 2: Open exported Excel file.

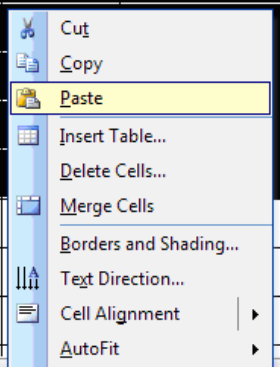
STEP 3: Select the data from Excel sheet (don't select first Heading row) and copy it to clipboard. (Look the figure below)



STEP 4: In Microsoft Word (TOL_template.doc is open) scroll to find following (look at figure on next page) table. Select the area and paste data from clipboard. For example, if there were 5 test cases in Excel sheet, select 5 rows (from TC1 to TC5) and 4 columns (from Interface to TC Description).

Table 1: Acceptance Test Cases

TC nr	Interface	Standard	TC Heading	TC Description	Result	C
TC1						
TC2						
TC3						
TC4						
TC5						
TC6						
TC7						
TC8						

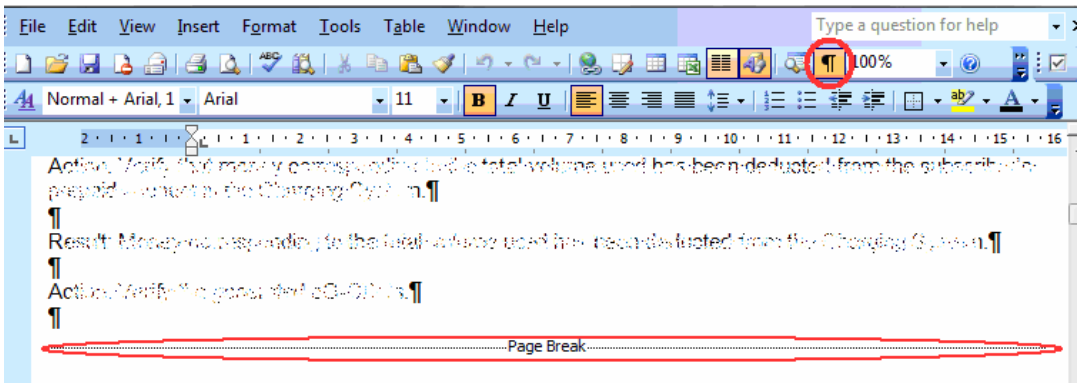


STEP 5: Click File / Save As… to save the document with new filename. (Don't overwrite the template).

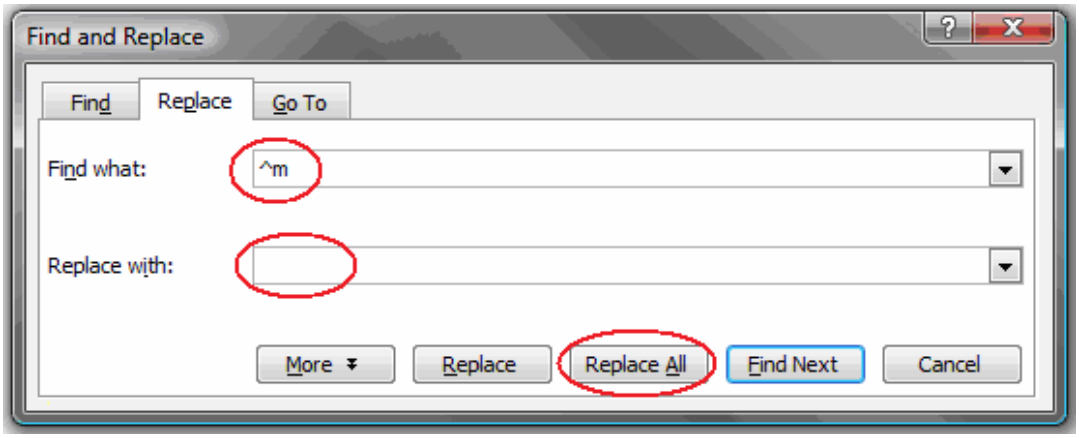
4.2 Test Description

STEP 1: Download and open file TOL_template.doc

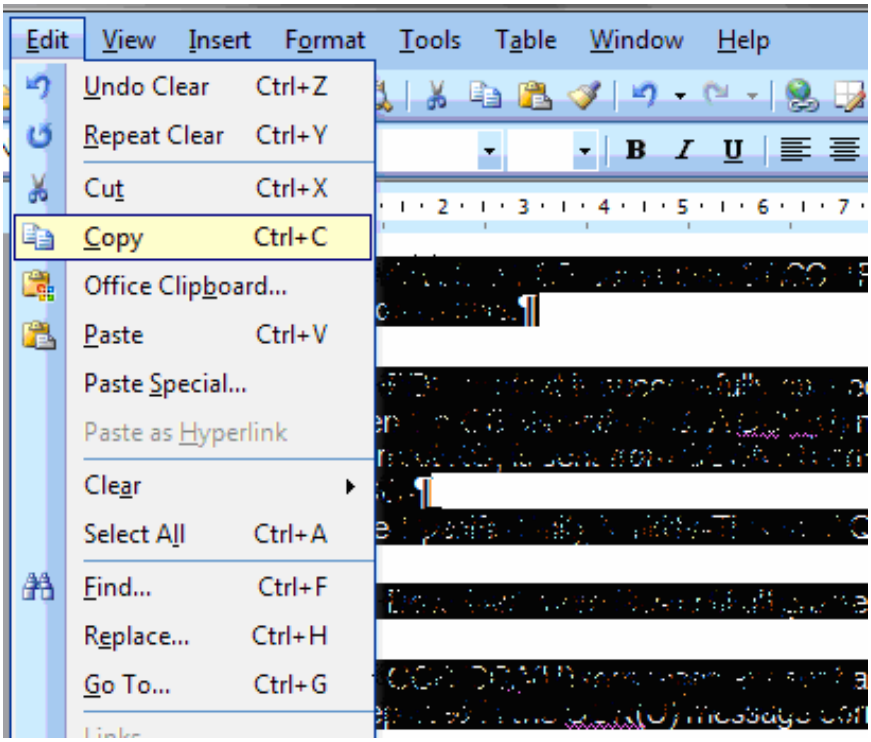
STEP 2: Open exported Word (RTF) file. And set ¶ -marks on.



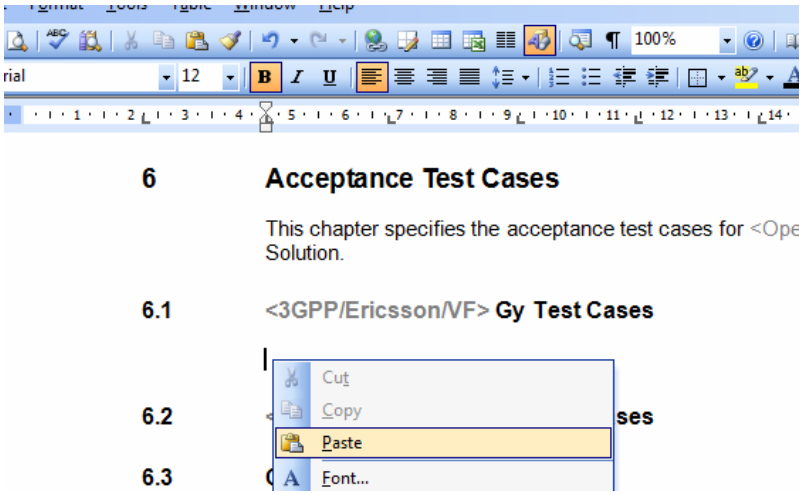
STEP 3: Remove Page Breaks manually or automatically by using **Replace** function from Edit –menu (CTRL+H). Type **^m** to Find what –field and leave **Replace with** –field **empty** and click **Replace All** –button. Look the figure on next page.



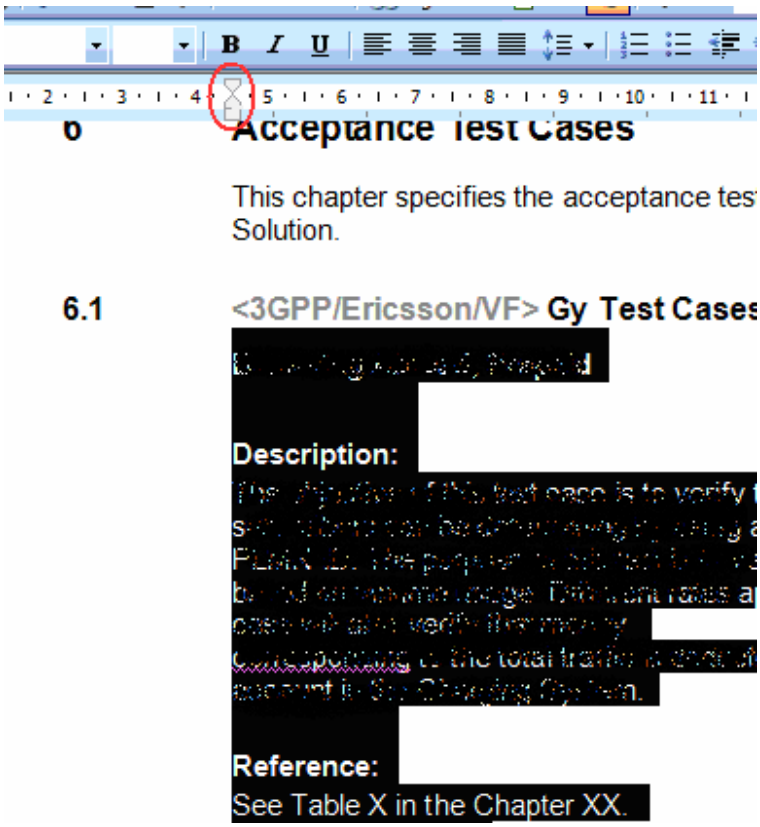
STEP 4: Select text (Test Cases) which you want to copy to the document, or press CTRL+A to select all. Copy selected text to clipboard.



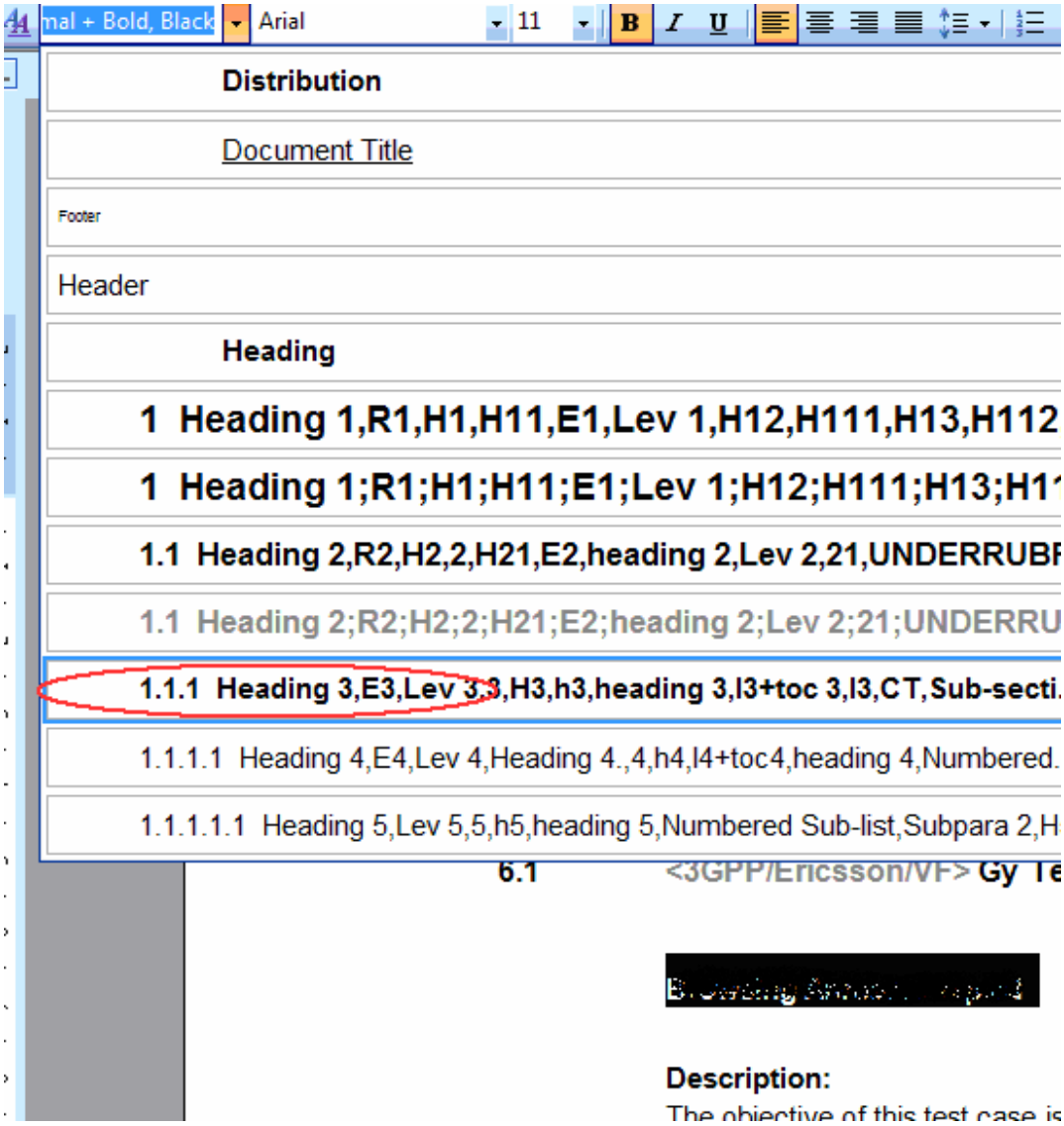
STEP 5: In Microsoft Word (TS_template.doc is open) scroll to find correct header and paste text from clipboard. See the figure on next page.



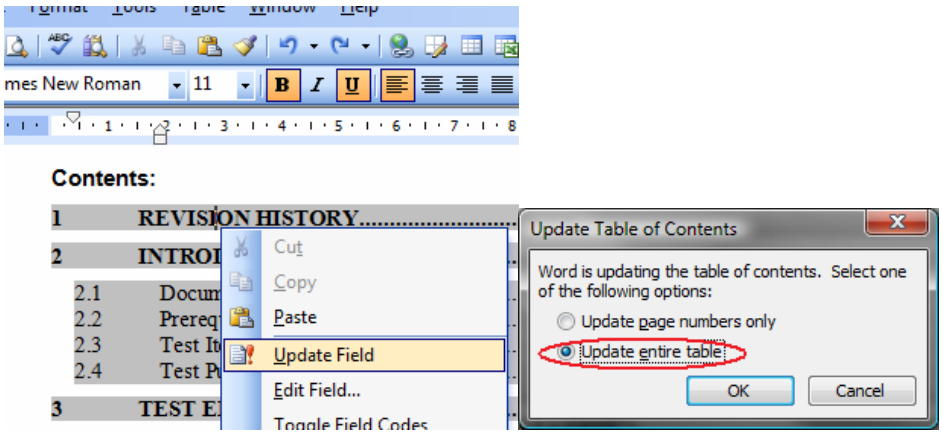
STEP 6: Select all new text you pasted into this document and drag tab indicator on the ruler bar to set text into the right position. See figure.



STEP 7: Select the Heading from the text and apply style **1.1.1 Heading 3** on it. Repeat this step to all Test Case headings.



STEP 8: Go to the Content –page of the document and update index list by clicking right mouse button on the list and choose **Update Field**. Then change selection to **Update entire table** and click **OK**. Look figure below.



ABBREVIATIONS

CDR	Charging Data Record
N / A	Not Available
PISC	Packet Inspection and Service Classification
TC	Test Case
TCL	Test Case Library
TD	Test Description
TOL	Test Object List
TR	Test Report
TS	Test Specification